




**Canadian Journal of Learning and Technology / La revue  
canadienne de l'apprentissage et de la  
technologie, V33(1) Winter / hiver, 2007**

HOME   ABOUT   LOG IN   REGISTER   SEARCH   CURRENT   ARCHIVES

Home > V33(1) Winter / hiver, 2007 > Kay

Font Size:   



**Canadian Journal of Learning and Technology**  
**Volume 33(1) Winter / hiver 2007**  
**A Formative Analysis of Resources Used to Learn Software**

**Robin Kay**

**Authors**

Robin Kay is an assistant profession in the Faculty of Education, at the University of Ontario Institute of Technology, Oshawa, Ontario. Correspondence regarding this article can be addressed by e-mail to: robin.kay@uoit.ca

**Abstract**

**Abstract:** A comprehensive, formal comparison of resources used to learn computer software has yet to be researched. Understanding the relative strengths and weakness of resources would provide useful guidance to teachers and students. The purpose of the current study was to explore the effectiveness of seven key resources: human assistance, the manual, the keyboard, the screen, the software (other than main menu), the software main menu, and software help. Thirty-six adults (18 male, 18 female), representing three computer ability levels (beginner, intermediate, and advanced), volunteered to think out loud while they learned, for a period of 55 minutes, the rudimentary steps (moving the cursor, using a menu, entering data) required to use a spreadsheet software package (Lotus 1-2-3). The main menu, the screen, and the manual were the most effective resources used. Human assistance produced short term gains in learning but was not significantly related to overall task performance. Searching the keyboard was frequently done, but was relatively ineffective for improving learning. Software help was the least effective learning resource. Individual differences in using resources were observed with respect to ability level and gender.

**Résumé :** Aucune comparaison complète et officielle des ressources utilisées dans l'apprentissage des logiciels informatiques n'a encore été effectuée. La compréhension des forces et des faiblesses relatives des ressources pourrait être utile pour les enseignants et les étudiants. La présente étude avait pour objet d'examiner l'efficacité des sept ressources principales : l'aide des pairs, le manuel, le clavier, l'écran, le logiciel (autre que le menu principal), le menu principal du logiciel et l'aide sur le logiciel. Trente-six adultes (dix-huit hommes et dix-huit femmes) de trois niveaux d'habiletés différents avec les ordinateurs (débutant, intermédiaire et avancé) se sont portés volontaires pour faire part de leurs commentaires tandis qu'ils apprenaient, pendant 55 minutes, les étapes rudimentaires (déplacer le curseur, utiliser un menu, saisir des données) nécessaires à l'utilisation d'un progiciel tableur (Lotus 1-2-3). Le menu principal, l'écran et le manuel se sont avérés être les ressources les plus efficaces. L'aide des pairs n'a permis d'effectuer que des gains à court terme en apprentissage mais elle n'était pas vraiment liée à l'exécution globale des tâches. Le clavier a été examiné régulièrement mais a peu contribué à l'amélioration de l'apprentissage. L'aide sur le logiciel a été la ressource d'apprentissage la moins efficace. On a observé des différences dans l'utilisation des ressources selon le niveau d'habileté et le genre.

**Overview**

The role of information and communication technology (ICT) education is becoming increasingly important (Bannert, 2000; Lambrecht, 1999; Oblinger & Maruyama, 1996). According to the 2002 US Economic Census, approximately 15 billion dollars is spent on improving the computer skills of employees. As well, most computer users have to deal with major software changes every 6-18 months (Bellis, 2004; Franzke & Rieman, 1993). The most typical response of institutions to the need for ICT education is to focus on developing effective "stand-and-deliver" training programs (Mahaptra & Lai, 2005; Niederman & Webster, 1998; Olfman, Bostrom, & Sein, 2003), however there is some evidence to suggest that this teaching

approach is not particularly effective (Olfman & Bostrom, 1991; Olfman & Manviwalla, 1995; Shayo & Olfamn, 1993). It has been estimated that more than 50% of the participants in software workshops fail to use the software they were trained on (Olfman & Bostrom, 1991; Olfman & Manviwalla, 1995). Cross (2006) argues that formal training while providing a potentially useful start cannot properly address today's rapidly changing working environments.

Self-regulated, exploratory or informal learning is an alternative and common method that many people use to learn new software (Bartholomé, Stahl, Pieschl, & Bromme, 2006; Cross, 2006). In one report assessing over 25,000 computer users, 96% of respondents reported that they taught themselves to learn software through trial and error (Dryburg, 2002). In order to be successful at exploratory learning, though, effective help seeking behaviour or knowing how to use resources is critical (Bartholomé et al., 2006). The process of learning a new software package can become overwhelming very quickly (Draper, 1999). Ideally, a user would like to get the required help shortly after a specific problem arises (Modesitt, Maxim, & Akingbehin, 1999). Unfortunately many people are confused by the "modern labyrinth" of resources available. (Greif, 1994; Leutner, 2000; Mangold, 1997). According to cognitive load theory (Chandler & Sweller, 1991; Kester, Lehnen, Van Gerven, & Kirschner, 2006; Sweller, 1988, Sweller, van Merriënboer, & Paas, 1998), a user wants to minimize extraneous cognitive load (engaging in processes that are not beneficial to learning) and optimize germane cognitive load (engaging in process that help to solve the problem at hand). In addition, cognitive load will vary according to the experience of the user. In other words, the extraneous cognitive load might increase more rapidly for a novice than an advanced user because the advanced user has a more developed schema to understand and adapt to new situations. It is important, then, to examine how users negotiate resources in order to maximize learning.

To date, there is a relatively small volume of research on how ICT skills are developed (Taylor, 2003). More attention is spent on selecting content (software, version and platform) than on how to teach technology skills (Lambrecht, 1999; McEwen, 1996). Many studies focus on analysing the effectiveness of a single resource or approach (e.g., Bannert, 2000; Bartholomé et al., 2006; Belanger & Van Slyke, 2000; Carroll, 1990; Guzdial, 1999; Rieman, 1996). The purpose of this paper to examine and compare the effectiveness of a wide range of resources used to learn a new software package.

## Literature Review

The paradox inherent in self-guided or exploratory learning is that to learn, one must be able to interact with the software, however, to interact with the software, one must already have some knowledge about how to use it (Lambrecht, 1999). New users typically overcome this paradox by consulting a variety of resources: human assistance, a manual, the software itself (e.g., searching the screen, the menu, or software help), and the keyboard (Rieman, 1996; Reimann & Neubert, 2000). The little evidence that has been gathered comparing the use of resources suggests that people prefer to "try things out" (Carroll, 1990, 1998; Dryburg, 2002; Rieman, 1996), read the manuals (Rieman, 1996), ask for some form of human assistance (Dryburg, 2002; Rieman, 1996), and, in some cases, consult the software help system if they are particularly aggressive explorers (Rieman, 1996). Reimann & Neubert (2000) add that users tend to consult a hybrid of resources while learning instead of relying on a single support tool.

### *Using Software as a Resource*

Even though a majority of users prefer a trial-and-error approach to learning software, limited research has been done looking at how various features on the screen including the software menus are used to advance understanding. Several researchers have examined a "training wheels" approach to learning software where cognitive load is reduced by limiting the number of functions available (Bannert, 2000; Guzdial, 1999; Leutner, 2000). Users who learn with a reduced command set outperform those individuals presented with a full array of options (Leutner, 2000). Other research suggests that the successful use of an interface depends on whether the new software is consistent with previous software learned and how easy it is to guess at commands (Guzdial, 1999). Finally, with respect to using a software menu, users employ a label-following strategy—they select labels or words in the menu that are similar to words and concepts in the tasks they are trying to complete (Polson & Lewis 1990).

### *Using Manuals*

Manuals can provide extensive information in the form of task oriented instructions, indexes, table of contents, pictorial representations, and specialized short-cut guides (Rettig, 1991). In spite of these seemingly well organized aids, most new users, regardless of ability level, begin using new software without reading the manual (Carroll, 1990; Rettig, 1991; Simmons & Wild, 1991; Taylor, 2003). Rettig (1991) refers to computer manuals as the best sellers that no one reads, however Dryburg's (2002) extensive report on over 25,000 users noted that manuals are used 60% of the time at some point in the software learning process.

There is some evidence to suggest that using a manual actually improves learning performance. For example, Rieman (1996) noted that individuals can find out how to do tasks without manuals, but more advanced features remain untouched or unresolved. Bannert (2000) observed that acquiring new software skills with a manual was significantly faster and more productive than tutor guided instruction. However, not all manuals are the same. Manuals that contain ample error information (minimal manuals) help students perform better than manuals with limited error information (Carroll, 1990; Lazonder 1994, Lazonder & Van Der Meij, 1995; Van der Meij & Carroll, 1995). Van der Meij (2000) adds that the most successful format of a manual is a two column layout with instructions and full screen images presented side-by-side.

### ***Human Assistance***

While many users will play with software or use a manual, Simmons and Wild (1991) noted that the majority of individuals end up asking for help from a knowledgeable person. Rieman (1996) also observed that asking another person for help is a natural strategy but there can be several barriers—availability, feeling like you are bothering an experienced user too often, time finding someone, and being too proud to ask for help. E-mailing a person for help is rarely done because most people need a quick answer to their problems (Rieman, 1996). However, instant messaging might prove to be an attractive alternative to email given that response time would no longer be an issue. Several researchers have reported that users would rather ask for help on an “as needed basis” rather than be controlled by a tutor or trainer (Bannert, 2000; Simmons & Wild, 1991).

### ***Software Help System***

Software help features are designed to provide hints, instructions, and immediate feedback to guide new learners (Draper, 1999, Patrick & McGurgan, 1993). However, designing good help systems is not an easy task because it requires one to anticipate the needs and behaviours of a variety of learners (Allwood & Kalen, 1993; Duffy, Mehlenbacher, & Palmer, 1992; Lazonder & Van Der Meij, 1995; Patrick & McGurgan, 1993). Many users appear to spend little time using software help (Aleven & Koedinger, 2000; Bartholomé et al., 2006). Nonetheless, there is some evidence to suggest that properly designed help features can foster learning (Bartholomé et al., 2006; Wood & Wood, 1999), particularly context sensitive help (Bartholomé et al., 2006; Patrick & McGurgan, 1993).

### ***Input Devices***

The effect of hardware on learning has not been examined in much detail. The majority of research on man-machine interaction has not focused directly on hardware issues (Baecker & Buxton, 1987; Baecker, Grudin, Buxton, & Greenburg, 1995; Carroll, 1991; Norman & Draper, 1986). Buxton (1986), though, has looked at the role of input devices on user behaviour and has noted that the choice of an input device (e.g., keyboard, mouse) can have a marked effect on the user's model of how a software package works.

### ***Individual Differences***

**Ability** . The effective use of resources while learning new software is partially dependent on a user's perception of specific challenges that arise (Lazonder & Van Der Meij, 1995). It is reasonable to anticipate that more able users would have a broader perspective on potential barriers to learning software, and therefore would be more efficient at selecting appropriate resources (Bannert, 2000; Lazonder & Van Der Meij, 1995).

Novices appear to be inconsistent in their approach to learning software and using resources (Rieman, 1996). They are inefficient and often aimless when engaging in exploratory learning (Kamouri, Kamouri, & Smith, 1986; Kluwe, Misiak, & Haider, 1990; Polson & Lewis, 1990; Reimann & Neubert, 2000), have difficulty controlling their learning activities (Bannert, 2000) and knowing where to search for answers (Van der Linden, Sonnentag, Frese, & Van Dyck, 2001), and scan or act upon information very quickly (Brandt & Uden, 2003).

As learners grow and develop understanding and expertise, their need for software support and functionality will change as well (Jackson, Krajcik, & Soloway, 1998). More experienced users read manuals in greater depth (Rieman, 1999) and are more proficient in selecting and executing search strategies (Bartholomé et al., 2006; Lazonder, 2000; Wood & Wood, 1999). However, domain specific software expertise appears to be more important than general expertise (Draper, 1999). For example, specific knowledge of spreadsheet software would help an individual learn a new spreadsheet software package more than overall software expertise (Draper, 1999). In addition, the differences between novice and experts begin to disappear when tasks become more complex (Lazonder, 2000).

**Gender** . Gender differences in computer attitudes, use, ability, and behaviour (e.g., Kay, 1992; Kay, in press; Sanders, in press; Whitley, 1997) have consistently reported differences in favour of males. It is reasonable to speculate, then, that differences may occur with respect of the use of resources. To date, little research has been done on gender differences and the use of resources to learn new software. In one large scale study (Dryburg, 2002), men were more likely to use exploratory learning whereas women preferred facilitated methods (e.g., on the job training, help from friends, family, coworkers).

### ***Purpose and Specific Research Questions***

While a comparison of multiple help resources has been examined in previous research (Bannon, 1986; Borenstein, 1985; Carroll, Smith-Kerker, Ford, & Mazur-Rimetz, 1987/88; Dryburg, 2002; Norman & Draper, 1986; O'Malley, 1986; Rieman, 1996), a comprehensive, formal evaluation of the effectiveness of a wide range of resources has yet to be completed. A majority of studies focus on a single resource or approach (e.g., Bannert, 2000; Bartholomé et al., 2006; Belanger & Van Slyke, 2000; Carroll, 1990; Guzdial, 1999; Rieman, 1996).

The purpose of this paper was to examine and compare the effectiveness of seven resources used to learn a new software package: human assistance (the experimenter), the manual, the software itself (other than the menu), the software main menu, the software help system, the screen, and the keyboard. The specific research questions were as follows:

1. Is there a significant difference among resources with respect to frequency of use?
2. Is there a significant difference among resources with respect their impact on learning?
3. Are there significant differences among ability levels and gender with respect to resources used?

## Method

### Sample

The sample consisted of 36 adults (18 male, 18 female): 12 beginners, 12 intermediates, and 12 advanced users, ranging in age from 23 to 49 ( $M = 33.0$  years), living in the greater metropolitan Toronto area. Subjects were selected on the basis of convenience. Equal numbers of males and females participated in each ability group. Sixteen of the subjects had obtained their Bachelor's degree, eighteen their Master's degree, one a Doctoral degree, and one, a community college diploma. Sixty-four percent ( $n = 23$ ) of the sample were professionals; the remaining 36% were students ( $n = 13$ ). All subjects had one or more years experience using computer. Seventy-five percent ( $n = 27$ ) of the subjects had their own computers; 17% ( $n = 6$ ) intended to buy a computer in the future. All subjects voluntarily participated in the study.

### Procedure

**Overview**. Each subject was given an ethical review form, computerized survey, and interview before attempting the main task of learning the spreadsheet software package. Note that the survey and interview data were used to determine computer ability level (see data source below with respect to how computer ability levels were assessed). Once instructed on how to proceed, the subject was asked to think-aloud while learning the spreadsheet software for a period of 55 minutes. All activities were videotaped with the camera focused on the screen. Following the main task, a post-task interview was conducted.

**Software selection**. The spreadsheet software selected for this study was Lotus 1-2-3 (version 5.0). The software was deliberately chosen because it was unfamiliar to all subjects. The more advanced users had a wide range of software they were familiar with, so it was necessary to select a more obscure software package to establish true learning situation. It is realized that this choice of software might affect the generalizability of the results, however, the range of skills attempted during the 55 minute learning session, in general, was quite limited: moving around the screen, using the menu, and entering data. These skills are pretty common to most spreadsheet software packages, regardless of the version, and are typically done in the same way.

**Learning tasks**. The selection and presentation of learning tasks was carefully designed to be as authentic as possible. To do this, spreadsheet software was chosen because most participants had minimal experience using this software. Spreadsheet software is used to create, manipulate, and present rows and columns of data. The mean pre-task score for spreadsheet skills was 13.1 ( $SD = 15.3$ ) out of a total possible score of 44. Ten of the subjects (6 advanced users, 4 intermediates) reported scores of 30 or more. None of the subjects had ever used the specific spreadsheet software package used in this study (Lotus 1-2-3).

Subjects attempted a maximum of five spreadsheet activities arranged in ascending level of difficulty including (1) moving around the spreadsheet (screen), (2) using the command menu, (3) entering data, (4) deleting, copying, and moving data, and (5) editing. They were first asked to learn "in general" how to do activity one, namely moving around the spreadsheet. When they were confident that they had learned this activity, they were then asked to complete a series of specific tasks. This semi-structured exploratory approach to learning software is supported by a number of researchers (Bannert, 2000; Kester et al., 2006; Leutner, 2000; Wiedenbeck, Zavala, & Nawyn, 2000; Wiedenbeck & Zila, 1997). All general and specific activities were done in the order presented in Appendix A.

From an initial pilot study of 10 subjects, it was determined that 50–60 minutes was a reasonable amount of time for subjects with a wide range of abilities to demonstrate their ability to learn the spreadsheet software package. Shorter time periods limited the range of activities that beginners and intermediate subjects could complete.

In the 55 minute time period allotted to learn the software in the current study, a majority of the subjects completed all learning tasks with respect to moving around the screen (100%) and using the command menu (78%). About two thirds of the subjects attempted to enter data (69%), although only one third finished (33%) all the activities in this area. Less than 15% of all subjects completed the final tasks: deleting, copying, moving, and editing data.

### Data Collection

**Think-aloud protocols (TAPs)**. The main focus of this study was to examine the use of resources with respect to learning computer software. The use of think-aloud protocols (TAPs), where subjects verbalize what comes to their mind as they are doing a task, is one promising technique for examining transfer. Essentially, the think-aloud procedure offers a window into the internal talk of a subject while he/she is learning. In a detailed critique of TAPs, Ericsson and Simon (1980) conclude that "verbal reports, elicited with care and interpreted with full understanding of the circumstances under which they were obtained, are a valuable and thoroughly reliable source of information about cognitive processes" (p. 247).

**Learning behaviours**. The analyses used in this study are based on think-aloud data. Specifically, 3,169 learning behaviours involving the use of seven resources were identified and rated according to the degree to which they influenced learning. A learning behaviour was defined as "an action that influenced learning or the completion of an assigned task". Examples of learning behaviours might include pressing a key on the keyboard, searching the keyboard, reading the manual or screen, or searching through a menu

**Presentation of TAPs**. The following steps were carried out in the think-aloud procedure to ensure high quality data:

- Step 1. (Instructions) Subjects were asked to say everything they were thinking while working on the software. Subjects were told not to plan what they were going to say;
- Step 2. (Examples) Examples of thinking aloud were given, but no practice sessions were done;
- Step 3. (Prompt) Subjects were told it was important that they keep talking and that if they were silent for more than 5 seconds, they would be reminded to "Keep talking";
- Step 4. ( Reading). Subjects were permitted to read silently, but they had to indicate what they were reading and summarize when they had finished;
- Step 5. (Giving help). If a subject was really stuck, he/she could ask for help. A minor, medium, or major form of help would be given, depending on how "stuck" a subject was;
- Step 6. (Recording of TAPs). Both thinking aloud and the computer screen were recorded using an 8mm video camera.

### Data Sources

**Resources** . There were seven resources examined: human assistance (the experimenter), the manual (an independent, best selling book), the spreadsheet software other than the menu, the software main menu, the software help system, the screen, and the keyboard. These resources were selected based on their prevalence in the literature review. Note that use of the software was broken down into three separate categories (using software, reading screen, using the menu) to provide more detailed information. Operational definitions of each of the resources are provided in Table 1. A total of 3,169 learning behaviours were categorized according what resource was used.

It is important to note that human assistance, one of the resources a subject could use, was only given if a subject felt he/she was stuck. This instruction was deliberately included in an attempt to mimic a fairly typically software learning experience. It is assumed that most people do not have a personal or readily available expert who is willing to answer numerous questions about a new software package being learned. In addition, some research suggests (e.g., Reiman, 1996), there that the typical user is reluctant to ask for help at first.

**Table 1.** Operational Definitions of Resources Used

Resource	Description
Human Assistance	Subject was instructed to ask the experimenter for help only when they were really stuck.
Manual	Subject searched and read a manual on learning spreadsheet software.
Using Software	Subject used the software (not the main menu of software help) to try to do as task, often done by trial-and-error.
Screen	Subject specifically read or searched the screen while looking for help – Note subjects were not looking at a menu or the help screen
Software Menu	Subject searched and tried commands on the software's main menu.
Software Help	Subjects went to help area of the software package and tried to find solutions to problems.
Keyboard	Subject searched the keyboard for solutions.

**Computer ability** . Three computer ability levels were compared in this study: beginners, intermediates, and advanced users. The criteria used to determined these levels included years of experience, previous collaboration, previous learning, software experience, number of application software packages used, number of programming languages/operating systems known, and application software and programming languages known (78 items, reliability estimates ranged from .79 to .97). A multivariate analysis showed that beginners had significantly lower scores than intermediate and advanced users ( $p < .005$ ), and intermediates users had significantly lower scores than advanced users on all eight measures ( $p < .005$ ).

**Learning** . After each of the 3,169 learning behaviours were categorized according to resource type, they were scored on their immediate influence on learning (a score from -3 to +3 – see Table 2 for rating criteria). Five variables were used to evaluate the relative effectiveness of resources: how often a resource was used (frequency), mean influence the resource had on learning (Table 2), percentage of subjects who used the resource, total resource effect score, and performance score. Conceptually, the first three variables assessed prevalence (how often the behaviour was observed and by how many subjects) and intensity (mean influence of resource). The fourth variable, total resource effect score, is a composite of the first three variables and was calculated by multiplying the frequency in which a resource was used by the mean influence score of the resource by the percentage of subjects who used the resource. For example, the software main menu was used 1080 times, had a mean influence of 0.73, and was used by 83% of the subjects. The total resource effect score, then, was 657.0 (1080 x .73 x .83).

Performance scores were calculated by adding up the number of subgoal scores that each subject attained during the 55 minute time period. For each task, a set of learning subgoals was rated according to difficulty and usefulness. For example, the task of "moving around the screen" had five possible subgoals that could

be attained by a subject: using the cursor key (1 point), using the page keys (1 point), using the tab keys (1 point), using the GOTO key (2 points), and using the End-Home keys (2 points). If a subject met each of these subgoals successfully, a score of 7 would be given. If a subject missed the last subgoal (using the GOTO key) a score of 5 would be assigned. A sample of scoring for the first main goal, moving around the screen, is presented in Appendix B.

Note that nine performance scores were calculated for each subject including five main spreadsheet categories (moving around the screen, using the menu, entering data, editing data, and modifying data) and four resources (understanding how to use software help, the screen, the manual, and the keyboard).

**Reliability of TAPs**. Reliability and validity assessments were derived from the feedback given during the study and a post-task interview. One principle concern was whether the TAPs influenced learning. While, several subjects reported that the think-aloud procedure was “weird”, “frustrating” or “difficult to do”, the vast majority found the process relatively unobtrusive. Almost 70% of the subjects (n=25) felt that thinking aloud had little or no effect on their learning.

**Table 2.** Criteria for Scoring Influence on Learning

Score	Criteria Used	Example
-3	A significant misunderstanding or mistake is evident that is judged to use a significant amount of time	Subject thinks that the software help is the main menu and spends 15 minutes learning to do the wrong task.
-2	A significant misunderstanding or mistake which leads subject away from solving the task at hand	Subject believes all commands are on the screen and does not understand that there are sub menus. This results in some time loss and confusion.
-1	Minor misconception that has little effect on the direct learning of the task at hand	Subject tries HOME key, which takes him back in the wrong direction, but does not cause a big problem in terms of moving to the specified cell.
0	(a) Activity has no apparent effect on progress OR (b) Can't directly determine effect of activity OR (c) Both good and bad effects	(a) Subject tries a key and it does not work (e.g., gets beeping sound); (b) Subject gets upset, but it is hard to know how it affects future actions; (c) Subject moves to cell quickly, but fails to learn a better method. It is good that he completed the task, but bad that he did not learn a more efficient method.
1	Minor task or piece of knowledge learned which is not significantly useful in learning the software	Subject uses cursor keys, which are moderately helpful in terms of moving around the screen.
2	Learns significant piece of knowledge which is useful in terms of learning how to use the software	Subject learns the function of the ENTER key when pressed in the command menu. This is an important key.
3	A significantly useful piece of knowledge is learned which saves the subject considerable time	Subject finds page in the manual which clearly lists all the movement keys. Without this page, most subjects would take at least 10 minutes to learn only a subset of the movement keys.

The accurate rating of the influence of a resource on learning (Table 2) is critical to the reliability and validity of this study. Because of the importance of the learning influence scores, six outside raters were used to assess a 10%, stratified, random sample of the total 3,169 occasions when resources were used. Inter-rater agreement was calculated using Cohen's Kappa (Cohen, 1960), a conservative and robust measure (Bakeman, 2000; Dewey, 1983). The Kappa coefficients for inter rater agreement between the experimenter and six external raters (within one point) were as follows: *Rater 1*: .80, *Rater 2*: .82, *Rater 3*: .95, *Rater 4*: .94, *Rater 5*: .93, *Rater 6*: .93. Coefficients of .90 or greater are nearly always acceptable and .80 or greater are acceptable in most situations, particularly for the more conservative Cohen's Kappa (Lombard, Snyder-Duch, & Bracken, 2004).

## Results

**Table 3.** Frequency of Resources Used

Resource Used	Min	Max	Mean (S.D.)	Total No. Obs.	% of Total
Software Menu	1	62	30.0 (12.5)	1080	34%
Keyboard	4	41	19.1 ( 8.3)	687	22%
Manual	0	50	16.3 (13.0)	581	18%
Software Help	0	28	7.8 ( 8.3)	281	9%
Screen	0	17	7.6 (3.5)	273	9%
Using Software	0	12	4.4 (3.5)	159	5%
Human Assistance	0	16	2.9 (3.0)	108	3%
Total				3169	100%

### ***Frequency of Resources Used***

The frequencies of resources used by subjects in this study are presented in Table 3. Subjects used the main menu most often or about one third of the time. The manual and the keyboard, the next two most frequently used resources, were consulted about 20% of the time each. Software help, the screen, and using the spreadsheet software, were used less than 10% of the time. Finally, human assistance was asked for least often although this was not unexpected since subjects were instructed to consult this resource as a last resort. Over the 55 minute period, each subject, on average, used the software main menu 30 times, the keyboard 20 times, the manual 16 times, software help and the screen 7 times each, the software (trial and error) 4 times, and human assistance 3 times.

### ***Mean Influence Score***

Recall that mean influence score was defined as the average immediate effect that a particular resource had on learning (see Table 2 for the rubric). A one-way ANOVA, comparing mean influence scores for the seven resources was significant ( $F = 14.00, p < .001$ ). Human assistance had the highest mean influence on learning ( $M = 1.06, S.D. = 0.86$ ) and was significantly more effective than using the manual ( $M = 0.38, S.D. = 1.04$ ), software help ( $M = 0.37, S.D. = 1.15$ ), or the keyboard ( $M = 0.57, S.D. = 1.09$ ) (Scheffé post hoc analysis,  $p < .05$ ). Reading the screen was the second most influential resource on learning ( $M = 0.83, S.D. = 0.74$ ) and was significantly more helpful than the manual or software help (Scheffé post hoc analysis,  $p < .05$ ). Using the software menu ( $M = 0.73, S.D. = 1.14$ ) and other software features ( $M = 0.73, S.D. = 1.14$ ) were the next two highest mean influences on learning. Using the software menu was significantly more effective than using the manual or software help (Scheffé post hoc analysis,  $p < .05$ ) and using other software features was significantly more effective than using the manual (Scheffé post hoc analysis,  $p < .05$ ). Clearly the manual and the software help were the least effective resources in terms of influencing learning.

It is interesting to note that those resources that had the most immediate impact on learning were not necessarily used more often. Correlations between mean influence score and frequency for all resources were non significant.

### ***Percentage of Subjects Who Used Resources***

While there was some variation with respect to the percentage of subjects who used resources, all resources were used frequently. The least often used resource was human assistance, but subjects still used it over 75% of the time. Most subjects incorporated multiple resources. Forty-seven percent of all subjects ( $n=17$ ) used all seven resources, 94% of all subjects ( $n=34$ ) used 6 out of seven resources, and 100% of the subjects used at least five resources. Clearly subjects consulted on a full range of resources when trying to solve software problems.

### ***Total Resource Effect Score***

The total resource effect score was determined by multiplying the frequency with which a resource was used by the percentage of subjects who used the resource by the mean influence score (see Table 4). The software menu produced the highest total resource effect score, based on frequent use and a moderate mean influence on learning score. However fewer subjects used this resource. The keyboard had the next highest total resource effect score, even though its mean influence on learning was relatively low. The high score was reported because all subjects used this resource frequently. Reading the screen produced a high total resource effect score, largely because of the high mean influence on learning score. Somewhat paradoxically, this resource was used relatively infrequently. Even though the manual was one of the least effective resources in terms of influencing learning, it ranked fourth in terms of total resource effect score. The higher ranking was primarily as result of frequent use. Trial and error use of the software produced a relative low total resource effect score, not because it was ineffective in terms of influencing learning, but because it was used infrequently. Human assistance was not used often, resulting in the second lowest total resource effect score. However, as stated previously, it had the highest mean influence on learning when it was used. Finally, software help was used infrequently by relatively few subjects with minimal gains in learning. This combination produced a total effect score that was more than seven times lower than the highest score reported. Total resources effect scores for all resources are presented in Table 4.

**Table 4.** Total Resource Effect Score s

Resource	Frequency	% of Subjects Who Used Resource	Mean Influence on Learning (S.D.)	Total Resource Effect Score *
Main Menu	1080	83%	0.73 (1.2)	657.0
Keyboard	687	100%	0.57 (1.1)	391.6
Screen	273	97%	0.83 (0.7)	220.3
Manual	581	83%	0.39 (1.1)	188.8
Using Software	159	97%	0.74 (1.1)	114.4
Human Assistance	108	78%	1.06 (0.9)	89.0
Software Help	281	81%	0.37 (1.2)	83.8

\* Calculated by multiplying Frequency by % of Subjects who use resource by Mean Influence on Learning

### Performance Scores

A series of correlations was run for mean influence scores and performance scores (Table 5). The mean influence of using the menu was significantly and positively correlated with all task performance scores. In other words, subjects performed significantly better if they used the menu effectively. Using the manual effectively was significantly and positively correlated with the first two task performance scores (moving around the screen and using the menu) and negatively correlated with the software help performance score. Reading of the screen effectively was significantly and positively correlated with the performance scores for understanding the screen and using the menu. Using software help was significantly and positively correlated with the software help performance score, but no spreadsheet task performance scores. In other words, subjects who used software help, learned about help features, but this knowledge did not translate into improving performance on assigned tasks. Using the software (trial and error) was significantly correlated with only one performance score—software help. Finally the mean influence score for human assistance and searching the keyboard were not significantly correlated with any of the performance scores.

**Table 5.** Correlation Between Resource Mean Influence Scores and Performance Scores

Resources	Performance Scores										
	All Tasks	Move Task	Menu Task	Enter Task	Edit Task	Mod. Task	All Res.	Sft Hlp Res.	Screen Res.	Man. Res.	Key Res.
Main Menu	0.65 <sup>a</sup>	0.42 <sup>c</sup>	0.59 <sup>a</sup>	0.61 <sup>a</sup>	0.47 <sup>b</sup>	0.39 <sup>c</sup>	0.00	-0.06	0.06	0.01	0.02
Screen	0.24	0.23	0.37 <sup>c</sup>	0.26	0.04	0.09	0.38 <sup>c</sup>	0.20	0.49 <sup>b</sup>	0.05	0.14
Manual	0.13	0.47 <sup>b</sup>	0.42 <sup>c</sup>	-0.01	0.01	-0.07	-0.08	-0.46 <sup>c</sup>	0.01	0.25	0.15
Software Help	0.14	0.07	0.11	0.29	-0.04	0.04	0.52 <sup>b</sup>	0.49 <sup>b</sup>	0.18	0.30	0.10
Software (Using)	0.31	0.25	0.23	0.31	0.19	0.20	0.31	0.34 <sup>c</sup>	0.20	0.02	0.10
Keyboard	0.14	0.24	0.03	0.17	0.09	0.05	0.16	0.16	-0.08	0.14	0.09
Human	-0.03	-0.12	-0.05	-0.01	0.09	-0.07	0.03	0.15	-0.15	-0.20	0.09

<sup>a</sup> -  $p < .001$

<sup>b</sup> -  $p < .005$

<sup>c</sup> -  $p < .05$

### Individual Differences (Ability & Gender)

**Frequency of Resource Use.** A Pearson Chi-Square test showed a significant association among ability and frequency of resources used ( $\chi^2(12) = 74.31, p < .001$ ). From Table 6, it appears that beginners search the keyboard and use software help more often than intermediate or advanced users. Intermediate and advanced users, on the other hand, seem to use the software (both main menu and trial and error of other features) more than beginners. The manual, human assistance, and the screen were referred to by all ability groups equally.

**Table 6.** Frequency of Resource Use by Ability Level



	Ability			Total
	Beginner (% of Total)	Intermediate (% of Total)	Advanced (% of Total)	
Manual	195 (34%)	211 (36%)	175 (30%)	581
Human	40 (37%)	35 (32%)	33 (31%)	108
Keyboard	278 (40%)	191 (28%)	218 (32%)	687
Software (Using)	36 (23%)	65 (41%)	58 (36%)	159
Software Menu	291 (27%)	355 (33%)	434 (40%)	1080
Software Help	127 (45%)	78 (28%)	76 (27%)	281
Software Screen	92 (34%)	89 (33%)	92 (34%)	273

There was a significant association between males and females with respect to frequency of resources used ( $\chi^2(6) = 38.15, p < .001$ ). Males preferred to use the manual and read the screen, whereas females liked to ask for human assistance and use the software help (see Table 7).

**Table 7.** Frequency of Resource Use by Gender

	Female (% of Total)	Male (% of Total)	Total
Manual	258 (44%)	323 (56%)	581
Human	62 (57%)	46 (43%)	108
Keyboard	321 (47%)	366 (53%)	687
Software (Using)	85 (53%)	74 (47%)	159
Software Menu	575 (53%)	505 (47%)	1080
Software Help	172 (61%)	109 (39%)	281
Screen	116 (42%)	157 (58%)	273

**Mean Influence of Resources .** A 3-way ANOVA revealed three new significant effects (Table 8). First, advanced users ( $M = .74, S.D. = 1.10$ ) were significantly better at using resources than intermediate ( $M = .60, S.D. = 1.10$ ) or beginner users ( $M = .51, S.D. = 1.08$ ) (Scheffé post hoc analysis;  $p < .05$ ). Second, males ( $M = .71, S.D. = 1.06$ ) were significantly better at using resources, as a whole, than females ( $M = .53, S.D. = 1.13$ ) ( $p < .005$ ). Third, there was an interaction effect for ability level and use of resources ( $p < .005$ ). From Figure 1, it appears that beginners are less able to use the manual and the spreadsheet software (trial and error) than their more experience counterparts. Furthermore, advanced users seem to be better at using software help than intermediate or beginner users.

**Table 8.** Three-way ANOVA for Mean Influence on Learning Score as a Function of Ability, Gender and Resource Category

Source	SS	df	MS	F
Ability	10.45	2	5.22	4.55***
Gender	11.44	1	11.44	9.96**
Resource	91.05	6	15.17	13.21*
Ability * Gender	2.81	2	1.40	1.23
Ability * Resource	33.50	12	2.79	2.43**
Gender * Resource	6.25	6	1.04	0.91
Ability * Gender * Resource	18.57	12	1.54	1.35
Error	3591.45	3127	1.15	
Total	3805.28			

\*  $p < .001$

\*\*  $p < .005$

\*\*\*  $p < .05$

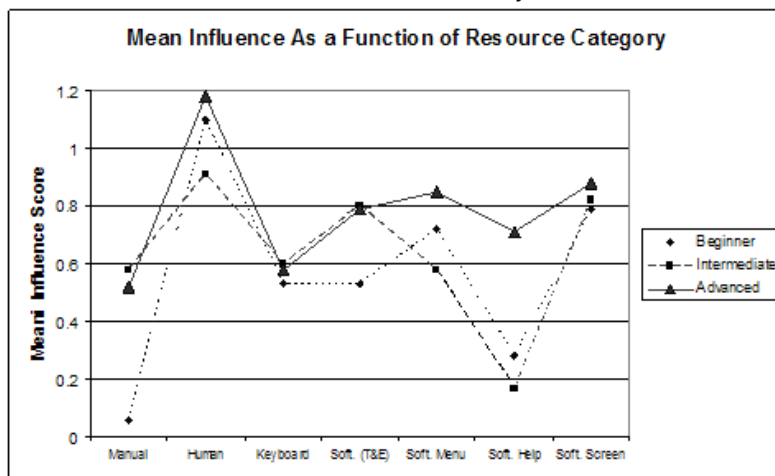


Figure 1. Interaction Effect for Mean Influence on Learning Score as a Function of Ability and Resource

## Discussion

The purpose of this study was to compare resources used to learn a new software package. Three research questions were asked:

1. Is there a significant difference among resources with respect to frequency of use?
2. Is there a significant difference among resources with respect their impact on learning?
3. Are there significant differences among ability levels and gender with respect to resources used?

### Frequency of Use

The frequent use of the software menu and searching the keyboard is consistent with a preference for a "trial and error" strategy reported in previous studies (Carroll, 1990, 1998; Dryburg, 2002; Rieman, 1996; Riemann & Neubert, 2000). However, minimal use of the software (other than the main menu) and the screen indicates that subjects in this study did not rely solely on an exploratory approach. In fact, they used a hybrid of resources, a pattern that was reported by Rieman and Neubert. The manual was the third most frequent resource employed. The relatively high use of the manual in this study was predicted by Dryburg's report on over 25,000 users, but is inconsistent with a number of other studies (Carroll, 1990; Rettig, 1991; Simmons & Wild, 1991; Taylor, 2003).

One possible explanation for why a trial and error strategy was used less and the manual was used more is that a majority of the individuals, had never used spreadsheet software. Guzdial (1990) noted that successful use of new software partially depends on previous software used. Since spreadsheet software was completely unfamiliar to most subjects in this study, successful use of a trial-and-error approach may have been compromised and more conservative resources, such as a manual, may have been needed. In other words, subjects may not have had the skill to begin exploring on their own because they had no idea where to begin, therefore they had to consult a manual.

Previous research suggests that most users tend to avoid software help (e.g., Alevén & Keodinger, 2000; Bartholomé et al., 2006) and the results in this study confirmed this claim. Software help was used relatively infrequently and by fewer subjects. One reason might be that software help was not particularly helpful with respect to immediate learning or overall spreadsheet task performance.

### Impact on Learning

Human assistance, reading the screen, the software main menu, and using other software features (trial and error) had the highest immediate impact on learning performance. This result supports previous studies suggesting that learners rely heavily on human assistance (Simmons & Wild, 1991) and simply "trying the software out" (Carroll, 1990, 1998; Dryburg, 2002; Rieman, 1996; Riemann & Neubert, 2000). However, correlations between the immediate influence that these resources had on learning and overall spreadsheet task performance revealed a more complicated pattern. Effective use of the software main menu was positively and significantly correlated with overall performance on all spreadsheet tasks. Accurate reading and searching of the screen was significantly and positively correlated with only one final task performance score, namely using the main menu. Effective use of other software features and human assistance were not correlated with any of the final performance tasks. The above result indicates that while human assistance and trying various software features may be helpful in the short term, they did not help improve final performance outcome scores.

It could be speculated that successful use of software menus are based on understanding the terms and language behind spreadsheets. Subjects who are good at using the menu may have a better conceptual understanding of software. Use of non menu related features (e.g., column and row labels, status bar, insert key indicator) of the software, on the other hand, may be more random, requiring less organization and concept formation. Therefore, proficiency in non-menus aspects of software use may not be related to increases in overall performance. Finally, human assistance in this study was deliberately designed to be as

unobtrusive as possible, therefore vague hints were given in response to most requests for help. It is not surprising, then, that this form of help did not improve final spreadsheet task performance.

Even though subjects in this study relied on searching the keyboard and the manual for help, these two resources were not particularly helpful in terms of immediate learning performance. However, effective use of the manual was significantly correlated to final task performance with respect to moving around the screen and using the menu. Rieman (1996) and Bannert (2000) noted that manuals do actually improve learning, despite the fact that they are rarely the resource of choice. While the book used in this study was not a minimal manual with ample error information (e.g., Carroll, 1990; Lazoner & Van Der Meij, 1995), it was a best selling, third party, spreadsheet resource with extensive indexing. Clearly, it was of some benefit for certain tasks.

Software help was the least effective resource with respect to immediate influence on learning and overall spreadsheet task performance. While previous research indicates that properly designed software help can foster learning (e.g., Bartholomé et al., 2006; Wood & Wood, 1999), the design for the spreadsheet software help used in this study did not appear to be particularly promising. Users became better at using the software help, but not more proficient at using the actual software. It is possible that the software help was too complicated, requiring too much time and cognitive effort, thereby compromising the final goal of getting tasks done. According to cognitive load theorists (Chandler & Sweller, 1991; Kester et al., 2006; Sweller, 1988, Sweller et al., 1998) software help appeared to be maximizing extraneous cognitive load (engaging in processes that are not beneficial to learning) and minimizing germane cognitive load (engaging in process that help to solve the problem at hand). Finally it is worth noting that, users might be more adept and successful at using current and better designed software help features than the ones available in the software package examined at the time this study was done.

### ***Individual Differences***

Advanced and intermediate users in this study were more comfortable and more effective at using exploratory resources to learn (e.g., software menu, trial and error use of other software features). This is a well documented finding in the literature (Kamouri et al., 1986; Kluwe et al., 1990; Poloson & Lewis, 1990; Reimann & Neubert, 2000). Advanced users were also better at using the manual and software help than their less experienced counterparts, but not necessarily more resourceful. All subjects, regardless of ability level, used all resources, but beginners, and, to a lesser extent, intermediate users, were less able to benefit from these resources.

A typical beginner may not have enough knowledge to use the trial-and-error approach, so he/she relies on resources like software help, the keyboard, and to a lesser extent the manual. However, software help and manuals are arguably as complicated as the spreadsheet software itself, so beginners struggled to advance their knowledge. For beginners too much time is being spent on extraneous cognitive load (Chandler & Sweller, 1991; Kester et al., 2006; Sweller, 1988, Sweller et al., 1998) and not enough attention is being directed to learning the software. Human assistance and scaffolds in the form of minimal manuals (e.g., Carroll, 1990; Lazoner & Van Der Meij, 1995) and contextual help (e.g., Bartholomé et al., 2006; Patrick & McGurgan, 1993) may be critical in reducing cognitive load enough to help less able users make progress.

It is interesting to note that males were significantly better than females at using resources, as a whole, yet they were not significantly more experienced. This difference may be partially explained by significant differences in the kind of resources that males and females liked to use. Females appeared to prefer human assistance and software help, both of which were minimally helpful in learning performance. Males, on the other hand, choose to use the manual and the screen for help, and these resources were significantly correlated with learning performance. Females, because of their preference for facilitated learning (also observed by Dryburg, 2002), may have been at a significant disadvantage in this study because the reduced effectiveness of human assistance was built into the design. Subjects were asked to do as much on their own before asking for help. One might speculate that in situations where facilitated help is not available, females may struggle more with learning new software packages.

### **Summary**

This study explored seven resources used to learning a new spreadsheet software package: software main menu, the screen, the manual, human assistance, the keyboard, the screen, the software (other than main menu), and software help. The software main menu was used most often and had a significant positive impact on all learning tasks. Observing or searching the screen was used infrequently, preferred by males, and had a significant, positive effect on performance of some spreadsheet learning tasks. The manual was used in moderation, also preferred by males, and had a significant, positive effect on performance of some spreadsheet learning tasks. Human assistance was used infrequently, preferred by females, and did not improve overall learning performance, however these finding should be taken with a proverbial grain of salt, since all users were instructed to "ask" for help only after they had tried all other resources. Searching the keyboard was used frequently, especially by beginners, but was not significantly correlated with learning performance. Trial and error use of software features other than the menu was used infrequently, but effectively by advanced users, although this strategy was not significantly related to task performance.

**Table 9.** *Overall Comparison of Resources Used*

Resource	Freq	% Who Used Resource	Immediate Influence on Learning	Correlation with Task Performance	Ability Diff. (Freq)	Ability Diff. (Immediate. Effect on learning)	Gender Diff.
Main Menu	High	Med	Med	High	Adv/Int	None	None
Screen	Low	High	Med	Med	None	None	Males
Manual	Med	Med	Low	Med	None	Adv/Int	Males
Human Assistance	Low	Med	High	Low	None	None	Females
Keyboard	Med	High	Low	Low	Beg.	None	None
Using Software	Low	High	Med	Low	Adv/Int	Adv/Int	None
Software Help	Low	Med	Low	Low	Beg.	Adv	Females

Finally software help was used infrequently, although it was preferred by females, and had no significant effect on learning performance. Table 9 provides a comparison of resources examined in this study. Suggestions for Educators

Based on the results of this study, several suggestions can be offered to educators of computer studies:

Encourage students to look at the screen. Slowing down and observing closely might give important cues, especially to beginners who do not have the search skills of more advanced users;

Recognize that the manual and software help can be difficult to use. Provide coaching and support in the use of these resources, or they could hinder learning;

Provide students with a rich source of terms and concepts that will be used with the software being learned and make sure that the vocabulary matches that of the manual, software help, and the main men;

It might be wise to provide guidance with respect to using software menus. Subjects in this study made considerable use of the main menu as a resource tool;

No one strategy will work for all students. Beginners will need more help with using resources (handouts, minimal manuals, direct coaching), while advanced users will probably be able use trial and error strategies effectively on their own.

### Opportunities for Future Research

This study was designed to examine the use of resources used to learn computer software in a semi-structured, natural setting. A number of compromises had to be made with respect to capturing and analyzing data, and these decisions have to be considered when interpreting the results.

First, a specific software package had to be chosen—spreadsheets. It was chosen because the type of software was unfamiliar to most subjects and the specific software package was unfamiliar to all subjects. In addition, the software selected was dated compared to more advanced and popular spreadsheet packages. While using resources to learn this spreadsheet software may generalize to other software, that claim cannot be made from the results in this study.

Second, a semi-structured approach to learning where all subjects learned the same tasks in the same order may not be representative of the preferred learning scenario for some subjects. For example, if a subject preferred more a more formal learning approach with a textbook or software help, then a semi-structured approach might be a reasonable facsimile of his/her typical learning approach. On the other hand, if a subject preferred to use trial-and-error and a more non-directed learning style, the semi-structured format might be restrictive.

Third, subjects were allowed to choose any of the seven resources they wanted in any order (with the exception of human help). This approach was used to capture “natural” selection of resources; however, interpretation of the results is messy. For example, not all resources are equal. Human assistance, the manual, and software help potentially provide a much larger range of solutions than searching the keyboard or trying out the software. Therefore, the comparison of resources in this study must be considered formative, a first step into understanding potential use and impact.

Fourth, although over 3100 learning activities were analyzed, the sample consisted of only 36 subjects, who were highly educated, and in their thirties. The number of subjects had to be restricted due to the time required to transcribe, code and analyze the data—in this study the process took over a year. Nonetheless, the results might be quite different for other populations.

Fifth, the resources selected for this study were based on an extensive review of the literature. However, more current resources such as instant messaging, online tutorials, video clips, and web based instruction should be added to the resource list of future studies.

Finally, and perhaps most importantly, the study focused on short-term gains in learning. The results and conclusions do not necessarily apply to long term gains. This is an empirical question to be studied in the

future.

## References

- Aleven, V., & Koedinger, K. R. (2000). Limitations of student control: Do students know when they need help. In C. F. G. Gauthier & K. VanLehn (Eds.), *Proceedings of the 5th international conference on intelligent tutoring systems, ITS 2000* (pp. 292–303). Berlin: Springer Verlag.
- Allwood, C. M., & Kalen, T. (1993). User-competence and other usability aspects when introducing a patient administrative system: A case study. *Interacting with Computers, 5*, 167–191.
- Baecker, R. M., & Buxton, W. A. S. (1987). *Readings in human-computer interaction: A multidisciplinary approach*. San Francisco: Morgan Kaufmann Publishers Inc.
- Baecker, R. M., Grudin, J., Buxton, W. A. S., & Greenberg, S. (1995). *Readings in human-computer interaction: Toward the year 2000 (2nd ed.)*. San Francisco: Morgan Kaufmann Publishers Inc.
- Bakeman, R. (2000). Behavioral observation and coding. In H. T. Reis & C. M. Judge (Eds.), *Handbook of research methods in social and personality psychology* (pp. 138–159). New York: Cambridge University Press.
- Bannert, M. (2000). The effects of training wheels and self learning materials in software training. *Journal of Computer Assisted Learning, 16*(4), 336–346.
- Bannon, L. J. (1986). Helping users help each other. In D. A. Norman & S. W. Draper (Eds.) *User Centered System Design: New perspectives on human-computer interaction* (pp. 399–410). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Bartholomé, T., Stahl, E., Pieschl, S., & Bromme, R. (2006). What matters in help-seeking? A study of help effectiveness and learner-related factors. *Computers in Human Behavior, 22*(1), 113–129.
- Bellis, M. B. (2004). *Microsoft Windows*. Retrieved February 29, 2004 from <http://inventors.about.com/library/inventors/blwindows.htm>.
- Belanger, F., & Van Slyke, C. (2000). End-user learning through application play. *Information Technology, Learning, and Performance Journal, 18*(1), 61–70.
- Borenstein, N. S. (1985). *The design and evaluation of on-line help systems*. Pittsburgh: Carnegie-Melon University.
- Brandt, S., & Uden, L. (2003). Insight into mental models of novice Internet searchers. *Communications of the ACM, 46*(7), 133–136.
- Buxton, W. (1986). There's more to interaction than meets the eye: Some issues in manual input. In D. A. Norman & S. W. Draper (Eds.), *User centred system design: New perspectives on human-computer interaction* (pp. 319–337). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Carroll, J. B. (1990). *The Nurnberg funnel*. Cambridge, MA: MIT Press.
- Carroll, J. B. (1991). Introduction: The Kittle House manifesto. In J. M. Carroll (Ed.), *Designing interaction* (pp. 1–16). Cambridge, UK: Cambridge University Press.
- Carroll, J. M. (1998). *Minimalism beyond the Nurnberg funnel*. Cambridge, MA: MIT Press.
- Carroll, J. B., Smith-Kerker, P. L., Ford, J. R., & Mazur-Rimetz, S. A. (1987/88). The minimal manual. *Human Computer Interaction, 3*, 123–153.
- Chandler, P., & Sweller, J. (1991). Cognitive load theory and the format of instruction. *Cognition and Instruction, 8*, 293–332.
- Cohen, J., (1960), A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*, 37–46.
- Cross, J. (2006). *Informal learning : Rediscovering the natural pathways that inspire innovation and performance*. Hoboken, NJ: Wiley Press.
- Dewey, M. E. (1983). Coefficients of agreement. *British Journal of Psychiatry, 143*, 487–489.
- Draper, S. W. (1999). Supporting use, learning, and education. *Journal of Computer Documentation, 23*(2), 19–24.
- Dryburg, H. (2002). Learning computer skills. *Canadian Social Trends, Spring*, 20–23.
- Duffy, T. M., Mehlenbacher, B., & Palmer, J. E. (1992). *On line help: Design and evaluation*. Norwood, NJ: Ablex.
- Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review, 87*(3), 215–251.
- Franzke, M. & Rieman, J. (1993). Neutral training wheels: Learning and transfer between two versions of a computer application. In *Proceedings of the Vienna Conference on Human Computer Interaction, 93*. Springer-Verlag, Berlin, 317–328.

- Greif, S. (1994) Computer systems as exploratory environments. In H. Keller, K. Schneider & B. Henderson (Eds.), *Curiosity and Exploration*, pp. 287–306. New York: Springer-Verlag.
- Guzdial, M. (1999). Supporting learners as users. *Journal of Computer Documentation*, 23(2), 3–13.
- Jackson, S. L., Krajcik, J., & Soloway, E. (1998). *The design of guide learner-adaptive scaffolding in interactive learning environments*. Los Angeles, CA: CHI.
- Kamouri, A. L., Kamouri, J., & Smith, K. H. (1986). Training by exploration: Facilitating the transfer of procedural knowledge through analogical reasoning. *International Journal of Man-Machine Studies*, 24, 171–191.
- Kay, R. H. (1992). An analysis of methods used to examine gender differences in computer-related behaviour. *Journal of Educational Computing Research*, 8(3), 323–336.
- Kay, R. H. (in press). Addressing gender differences in computer ability, attitudes, and use: the laptop effect. *Journal of Educational Computing Research*.
- Kester, L. Lehnen, C., Van Gerven, P. W. M., & Kirschner, P. A. (2006). Just-in-time schematic supportive information presentation during cognitive skill acquisition. *Computers in Human Behavior*, 22(1), 93–116.
- Kluwe, R. H., Misiak, C., & Haider, H. (1990). Learning by doing in the control of a complex system. In H. Mandl, E de Corte, N. Bennet & H. F. Friedrichs (Eds.), *Learning and Instruction, Vol 2.1* (pp.197–218). New York: Pergamon.
- Lambrecht, J. J. (1999). Teaching technology-related skills. *Journal of Education for Business, Jan/Feb*, 144–151.
- Lazonder, A. W. (1994). Minimalist computer documentation and the effective control of errors. In M. Steehouder, C. Jansen, P. Van Der Poort, & R. Verheijen, (Eds.), *Quality of technical documentation* (pp. 85–98). Amsterdam: Rodopi.
- Lazonder, A. W. (2000). Exploring novice users; training needs in searching information on the WWW. *Journal of Computer Assisted Learning*, 16(4), 326–335.
- Lazonder, A. W., & Van Der Meij, H. (1995). *International Journal of Human-Computer Studies*, 42, 185–206.
- Leutner, D. (2000). Double-fading support – a training approach to complex software systems. *Journal of Computer Assisted Learning*, 16(4), 347–357.
- Lombard, M., Snyder-Duch, J., & Bracken, C. C. (2004). *Practical resources for assessing and reporting intercoder reliability in content analysis research projects*. Retrieved September, 2004 from <http://www.temple.edu/mmc/reliability>
- Mahaptra, R., & Lai, V. S. (2005). Evaluating end-user training programs. *Communications of the ACM*, 48(1), 67–70.
- Mangold, R. (1997) The contribution of media psychology to user-friendly computers: a proposal for cooperative work. In P. Winterhoff-Spurk & T. A. van der Voort (Eds.), *New horizons in media psychology. Research cooperation and projects in Europe*(pp. 73–86). Opladen: Westdeutscher-Verlag.
- McEwen, B. C. (1996). Teaching microcomputer software skills. *Business Education Forum*, 50(4), 15–19.
- Modesitt, K. L., Maxim, B. R., & Akingbehin, K. (1999). Just-in-time learning in software engineering. *Journal of Computers in Mathematics and Science Teaching*, 18(3), 287–301.
- Niederman, F., & Webster, J. (1998). Trends in end-user training: A research agenda. *CPR*, 224–232.
- Norman, D. A., & Draper, S. W. (Eds.) (1986). *User centered system design: New perspectives on human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Oblinger, D. G., & Maruyama, M. K. (1996). Distributed learning. *CAUSE Professional Paper Series*, #14, Boulder, CO: CAUSE.
- Olfman, L., & Bostrom, R. P. (1991). End-user software training: An experimental comparison of methods to enhance motivation. *Journal of Information Systems*, 1, 249–266.
- Olfman, L., Bostrom, R. P., & Sein, M. K. (2003). A best-practice based model for information technology learning strategy formulation. *SIGMIS Conference*, Philadelphia, PA.
- Olfman, L. & Mandviwalla, M. (1995). An experimental analysis of end-user software training manuals. *Information Systems Journal*, 5(1), 19–36.
- O'Malley, C. E. (1986). Helping users help themselves. In D. A. Norman & S. W. Draper (Eds.) *User Centered System Design: New perspectives on human-computer interaction* (pp. 377–398). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Patrick, A., & McGurgan, A. (1993). One proven methodology for designing robust online help systems. *ACM SIGDOC*, 223–232.
- Polson, P. G. & Lewis, C. H. (1990). Theory-based design for easily learned interfaces. *Human Computer*

*Interaction*, 6, 191–220.

Rettig, M. (1991). Nobody reads documentation. *Communications of the ACM*, 34(7), 19–24.

Rieman, J. (1996). A field study of exploratory learning strategies. *ACM Transactions on Computer-Human Interaction*, 3(3), 189–218.

Reimann, P., & Neubert, C. (2000). The role of self-explanation in learning to use a spreadsheet through examples. *Journal of Computer Assisted Learning*, 16(4), 316–325.

Sanders, J. (in press). Gender and technology: A research review. In C. Skelton, B. Francis, & L. Smulyan (Eds.), *Handbook of Gender and Education*. London: Sage.

Shayo, C., & Olfamn, L. (1993). Is the effectiveness of formal end-user software training a mirage? *CPR*, 98–99.

Simmons, C., & Wild, P. (1991). Student teachers learning to learn through information technology. *Educational Research*, 33(3), 163–171.

Sweller, J. (1988). Cognitive load during problem solving: effects on learning. *Cognitive Science*, 12, 257–285.

Sweller, J., van Merriënboer, J. J. G., & Paas, F. (1998). Cognitive architecture and instructional design. *Educational Psychology Review*, 10, 251–296.

Taylor, L. (2003). ICT skills learning strategies and histories of trainee teachers. *Journal of Computer Assisted Learning*, 19(1), 129–140.

United States Economic Census - Educational Services (2002). Retrieved Oct 6, 2006 from [http://www.census.gov/econ/census02/data/us/US000\\_61.HTM](http://www.census.gov/econ/census02/data/us/US000_61.HTM)

Van Der Linden, D., Sonnentag, S., Frese, M., & Van Dyck, C. (2001). Exploration strategies, performance, and error consequences when learning a complex computer task. *Behaviour & Information Technology*, 20(3), 189–198.

Van Der Meij, H. (2000). The role and design of screen images in software documentation. *Journal of Computer Assisted Learning*, 16(4), 294–306.

Van Der Meij, H., & Carroll, J. M. (1995). Principles and heuristics for designing minimalist instruction. *Technical Communication*, 42(2), 243–261.

Whitley, B. E., Jr. (1997). Gender differences in computer-related attitudes and behaviors: A meta-analysis. *Computers in Human Behavior*, 13, 1–22.

Wiedenback, S., Zavala, J. A., & Nawyn, J. (2000). An activity-based analysis of hands-on practice methods. *Journal of Computer Assisted Learning*, 16(4), 358–365.

Wiedenback, S., & Zila, P. L. (1997). Hands-on practice in learning to use software: A comparison of exercise, exploration, and combined formats. *ACM Transactions on Computer-Human Interaction*, 4(2), 169–196.

Wood, H., & Wood, D. (1999). Help-seeking, learning and contingent tutoring. *Computers and Education*, 33(2), 153–169.

## Appendix A - Specific Spreadsheet Tasks Presented to Subjects

### ***Specific Tasks 2:***

1. Move the cursor to B5.
2. Move the cursor to B161
3. Move the cursor to Z12.
4. Move the cursor to A1.
5. Move the cursor to HA1235.
6. Move the cursor to the bottom left corner of the entire spreadsheet.

### **General Task 2: Using the Command Menu**

#### ***Specific Tasks 2:***

- a) Move to the command menu, then back to the worksheet.
- b) Move to the command: Save.
- c) Move to the command: Sort.
- d) Move to the command: Retrieve.
- e) Move to the command: Set Width.
- f) Move to the command: Currency.

#### ***General Task 2: Entering Data into a Cell***

**General Task 3: Entering Data into a Cell**

*Specific Tasks 3:*

Name	Telephone	Sex	Date Due	Amount
Robin	900-0100	M	07/14/92	300.12
Mary	800-0200	F	06/16/92	20046.23

- a) Please start in cell A1 and enter all the information above.
- b) Centre the title SEX.
- c) Right justify the title AMOUNT.
- d) Widen the TELEPHONE column to 15 spaces.
- e) Narrow the SEX column to 5 spaces.

**General Task 4: Deleting, Copying, and Moving Data**

*Specific Tasks 4:*

DATA A	DATA C	DATA B	DATA D
10	1	100	11
15	2	200	22
20	3	300	33
25	4	400	44
30	5	500	55

- a) In the table above, move everything in Column A to Column B.
- b) Delete Row 4.
- c) Delete Column A.
- d) Delete the numbers 300-500 in the DATA B column.
- e) Name the range of data in DATA C column. Call this range DATAC.
- f) Copy the underline under DATA 1 to the cells under DATA B, C and D.

**General Task 5: Editing Data**

*Specific Tasks 5:*

Canadian
American
70002
Mistake
Replace Me

- a) In the table above, delete the space in Canadian.
- b) Add an "e" to American.
- c) Change 70002 to 80002.
- d) Delete the word Mistake.
- e) Replace the phrase Replace Me with the phrase New Me.

**Appendix B – Sample Scoring for Task Performance – Moving Around the Screen**

1 = Minor Knowledge		2 = Major Knowledge	
Task 0 - Open ended exploration			
Score	Description		
1	1001	NumPad vs other Cursor Keys	
1	1002	Cursor within cell pointer	
1	1003	Cursor Keys	
2	1004	PgUp/PgDn	
2	1005	Home	



1	1006	End
2	1007	End & cursor
2	1008	End and Home
2	1009	Tab
2	1010	Shift Tab
2	1011	Ctrl-RC and Ctrl-LC
1	1012	Move beyond initial horiz bar
2	1013	Illogical BLC task
2	1014	GOTO (F5)
2	1015	Figure out how active area works
1	1030	End Indicator
2	1040	Illogical End Task
<hr/>		
<b>Task 1 – Move the cursor to B5</b>		
<b>Score</b>	<b>Description</b>	
1	1101	Just need cursor keys
1	1102	Combination keys
<hr/>		
<b>Task 2 – Move the cursor to B161</b>		
<b>Score</b>	<b>Description</b>	
1	1201	If used only cursor keys
2	1202	If used PgDn/Cursor keys
2	1203	If used Goto method primarily
2	1204	Home or other combinations
<hr/>		
<b>Task 3 – Move the cursor to Z12</b>		
<b>Score</b>	<b>Description</b>	
1	1301	If only used cursor key
2	1302	If used PgUp key
1	1303	Understand what COL Z means
2	1304	If used Goto method only
2	1305	Home / tab keys
<hr/>		
<b>Task 4 – Move the cursor to A1</b>		
<b>Score</b>	<b>Description</b>	
2	1401	If used home
1	1402	If used cursor keys
2	1403	If used PgUp
2	1404	If used Shift-Tab/Ctrl-LC
2	1405	If used Goto method primarily
<hr/>		
<b>Task 5 – Move the cursor to HA1235</b>		
<b>Score</b>	<b>Description</b>	
1	1501	If used cursor keys
2	1502	If used PgUp, PgDn
2	1503	If used Tab/Ctrl- RC
2	1504	If used Goto method primarily
2	1505	Understand what COL HA means
<hr/>		
<b>Task 6 – Move the cursor to bottom left corner of the worksheet</b>		
<b>Score</b>	<b>Description</b>	
1	1601	if used cursor keys
2	1602	if used PgUp, PgDn
2	1603	used horz movement keys
2	1604	if used Goto method primarily

- 2 1605 checked book for row limits
- 2 1606 if used End arrow keys primarily
- 1 1607 Check if really at end
- 2 1609 Know what end REALLY does

---

© Canadian Journal of Learning and Technology

**ISSN: 1499-6685**



This work is licensed under a Creative Commons Attribution 3.0 License.

