The Macro-structure of Use of Help

Oscar D. Andrade, Nathaniel Bean, and David G. Novick,
Department of Computer Science
The University of Texas at El Paso
El Paso, TX 79968
+1 915-747-5480
odandrade@miners.utep.edu, nhbean@gmail.com, novick@utep.edu

ABSTRACT
Users of help systems often complain that they do not find them useful; while they still use help at least occasionally, they resort to other problem-solving strategies. In this paper, we analyze audiovisual recordings of people using a computer application, to identify (1) transition patterns among problem-solving approaches, and (2) the frequency of these transitions. Our analysis indicates that people switch frequently between consulting help and exploring the interface. Switching between problem-solving approaches appears to be an effective way of succeeding in tasks. Applications and their help systems can be better designed to support users who switch between help and non-help approaches to solving problems.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Evaluation/methodology, training, help, and documentation.*

General Terms
Documentation, Human Factors, Measurement

Keywords
Help systems, documentation, problem-solving

1. INTRODUCTION
When accomplishing new tasks in a computer application, some people use help systems ("help") often, and most use help occasionally [17],[18]. But more often, people prefer to muddle through the application through trial-and-error methods rather than use the application's help [3],[13],[14]. And even when they can be persuaded to use help, they still appear to prefer other strategies [13].

Simply adding a search function to an electronic version of a manual does not create a useful help system. Common complaints about help systems include vocabulary mismatches between the user and help, and cumbersome and confusing navigation that makes it difficult to find features [12]. Help systems tend to focus on the procedural steps of using a feature but often miss assisting users at the application's task level or have little support for problem solving. The lack of a task-based perspective or support for problem solving can lead users to have an incomplete or mistaken mental model of the task on which they are working and of the application they are using. These factors—task, application and user—were identified by Kearsley [6] as dimensions for help systems and formalized by Andrade and Novick [1] as the Tau model. Users routinely run into such issues and abandon the help system to try other approaches such as muddling through the application via trial and error [9],[14]. Other non-help problem-solving strategies include working around the problem to get a similar result with a different procedure and recalling how they completed a similar task or found the feature they needed.

Our goal is to examine why and how users of computer applications switch among three problem-solving approaches—consulting help, muddling through with trial and error, and recalling successful uses. In a companion paper [15], we look at the complementary "micro" issues of what happens within problem-solving episodes; here we focus on the "macro" aspects of the problem—the transitions among approaches. In this paper, we review research related to user preferences in problem-solving behaviors and common issues across the Tau dimensions, provide a framework for analyzing interactions of eight novice users of Microsoft Publisher and its help system, and analyze when and why the subjects switched among problem-solving approaches. In particular, we examine the issues that led subjects to switch between *help* and *non-help* approaches or to favor some strategies over others within the *non-help* group. We conclude with a summary of our observations and a discussion of the implications of our analysis for improving help systems.

2. BACKGROUND
How frequently do users switch problem-solving approaches, and what causes users to prefer one approach over another? As Newell [9] observed,

> Problem solving is always a matter of search — of starting from some initial position (state of knowledge) and exploring until a position is attained that includes the solution — the desired state of knowledge.

As users of computer systems solve problems, they generate a search space, using a variety of approaches. Rieman [16] concluded from an observational study that users learn about software by trial-and-error, consulting a manual, or asking for help. These approaches were prevalent across users of all experience levels, although novices had no consistent approach and experts learned to recognize similarities with other applications or had a task-driven approach to problem-solving. Rieman's study also showed that the strategy of choice was *trial and error* (also known as *exploratory learning*), which was used

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SIGDOC '09, October 5–7, 2009, Bloomington, IN, USA.
Copyright 2009 ACM 978-1-60558-559-8/09/10...$10.00.

"just-in-time" and was based on the task at hand—as opposed to being task-free exploration. Users in the study, however, did not learn by *trial and error* alone; they also read the manual or talked with co-workers, depending on resource availability. The subjects used help occasionally, but used other resources infrequently.

Similar attitudes among users of help systems were described by Novick and Ward [11], who interviewed a diverse set of users of computer applications. The participants indicated that they preferred documentation that is easy to navigate, provides explanations at an appropriate level of technical detail, enables solving problems through examples and scenarios, and is complete and correct. But participants' preferences sometimes conflicted, such as between a preference for coverage versus a preference for precision. Thus it is difficult to satisfy all users' preferences simultaneously. For example, providing both coverage and precision would increase the volume of help material, thus making navigation and use all the more difficult.

When solving problems, users appear to be highly pragmatic; they learn about the program and the task just to the extent needed to accomplish the task. This perspective on users' strategy for accomplishing tasks in computer applications is supported by evidence from other computer-based problem-solving situations. In particular, the issue of problem-solving strategies was addressed in the context of programmers asked to modify a complex program. Koenemann and Robertson [8] reported that programmers followed a pragmatic as-needed strategy rather than a systematic strategy. The programmers tended to learn only about parts of the program that were directly relevant to their task.

As early as 1980, an experimental study of the effectiveness of help [4] concluded that consulting online help takes time away from problem-solving. It appears that users prefer to muddle through an application to complete a task, and even when they can be persuaded to use help they still resort to other strategies because help does not provide them with the assistance they need [14]. Some of the issues causing high failure rates are vocabulary issues, difficulties in navigation, uncertain boundaries, lack of topic sorting by relevance, and lack of specific examples [9],[12].

3. METHODOLOGY

To determine the transitions people make among computer application problem-solving approaches, we analyzed high-definition audiovisual recordings collected as part of a prior study [14][13]. The subjects were administrative secretaries at a large university who, as part of the study, used Microsoft Publisher to complete a set of four assigned tasks; they did not use Publisher in their jobs, but were interested in learning a new application; some were interested in creating flyers or brochures with Publisher after seeing samples created by their coworkers. Eight out of the 22 subjects from the previous study were selected because they had the highest number of accesses to the help system, averaging eight accesses per task. All of the subjects were novice users of Publisher who, as a part of the study, received 15 minutes of training in Publisher by reading an adaptation of the Microsoft tutorial or by following a help-based tutorial for the same material. The subjects were then asked to complete four tasks, which involved replicating sample Publisher documents.

The interactions the subjects had with the application and the help system were classified in episodes. Though the boundaries between episodes were sometimes not clear (see [15]), we attempted to define the episodes as beginning when:

- A subject began a new task.
- A subject began working on a part of the task that involved one of the Publisher skills that the reference documents were intended to elicit.
- A subject switched from one problem-solving approach to another. For example, the subject was reading help or the tutorial and then began working on a step in the task through trial-and error.
- A subject stayed in the same condition, but began a new attempt. For example, the subject started a search with a relatively general search term, read a help topic, and then began a new search with a more specific search term, perhaps based on knowledge gained from reading the help topic.

Similarly, episodes ended when:

- The subject completed a task.
- The subject completed a step.
- The subject abandoned a step after repeated trials.
- The subject was interrupted because time expired for that task.

The problem-solving approaches used by the subjects to accomplish tasks in the application were categorized as help-based and non-help-based. Approach preference varied across subjects, especially as they gained more experience with the application.

We found two help-based approaches:

- Starting a search in the help system or looking back at a topic window they left open.
- Looking back at the tutorial to review how to use a feature they perceived as necessary to do a particular step. In the case of the help-based tutorial, they scanned the list of topics for suggestions on what to search in the help system.

We found two non-help-based approaches:

- Exploring application menus and toolbars to find relevant or similar features for what subjects needed to do, learning about the features by trial-and-error. That is, the subjects were trying to find features that appeared to be pertinent to what they wanted to do. In some cases, subjects using this trial-and-error ("T&E") approach accomplished tasks through work-arounds—solutions that approximated or provided the appearance of doing something rather than actually doing that thing such as, for example, arranging text in columns by inserting tabs rather than setting the number of columns.
- Recalling how to use a feature, or related feature, about which the subject had learned in a previous task. *Recall* differed from other approaches because the subjects made precise moves to the features they needed, taking little time to access the feature. *Recall* episodes averaged less than 30 seconds, compared with about 77 seconds for *T&E* episodes and about 71 seconds for *help* episodes. This suggests that subjects remembered how to find these features and did not rely on exploration to find them. *Recall* was used more in the last two tasks because by then most subjects had gained enough experience with Publisher to recall features that they had used. The tasks were designed based on the feature set covered in the training materials, which covered a small set of basic functions; thus many features were similar across different kinds of objects (e.g., borders for text boxes and images).

In the 124 episodes we studied, the subjects used the *T&E* approach more than any other. To take one episode, for example, a

subject intended to add a shadow effect to text. The subject inserted text in a text box and then clicked on **Word Art** in the **Objects** toolbar. She tried different Word Art styles during six minutes to see find one that would approximate the format of the message in the reference task. She clicked on **Text Box**, typed part of the message and highlighted it, looked for a text formatting option or a font type that would have a shadow but found nothing relevant, clicked on **Format**, paused for eight seconds on **Text**, looked at the available effects, clicked on **Shadow,** and clicked on **OK** to make the changes.

Within the *T&E* approach, users created work-arounds in some tasks more than in others, particularly in task 3, the "Golden Apple Advertisement" task shown in Figure 1. In this task, the subjects often simulated the two columns on the lower text box by typing text and using the space bar or tabs to push the text over to the far right of the text box. Also, some subjects separated the top and bottom sentences of the upper textbox by typing them and using the Enter key to insert lines rather use Publisher's text-wrapping options for images. In contrast, task 2, shown in Figure 2, offered fewer opportunities for work-arounds.

Figure 1. "Golden Apple" task

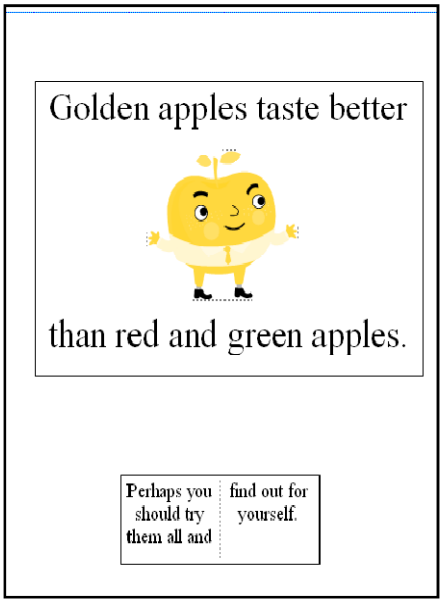
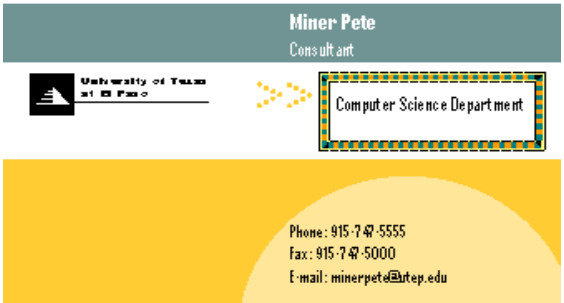


Figure 2. "Business Card" task



The subjects used the *recall* approach when they had used similar features earlier in their session. For example, some subjects recalled recoloring the apple in the advertisement and were able to find and use a similar function in completing task 4. In study, subjects used the *recall* approach in 34 episodes.

In 64 of the episodes, subjects used a help approach to problem-solving. In one episode, for example, a subject intended to rotate an image. She searched for "rotate" in the help system, clicked on the first topic ("Rotate or flip objects") from the list of topics resulting from the query , and then clicked on first sub-topic ("Rotate objects freely"). The subject followed the instructions and was able to rotate the arrow.

4. RESULTS

From our annotations of the subjects' problem-solving episodes, we developed a representation of their patterns of transition among problem-solving approaches, analyzed possible causes of these transitions, and examined the subjects' choices of problem-solving approach for initial episodes and for transition episodes.

4.1 Transition Patterns

Newell [9] tracked problem-solvers' searches in a representation called a problem behavior graph, which described the search space in terms of transitions between states of knowledge. In our study, we developed a more abstract representation that describes the subjects' transitions between problem-solving approaches rather than the detailed problem-solving steps taken using the approaches. For each episode, we created a state-transition diagram that represented the subject's use of a problem-solving strategy, including a start state, a problem-solving approach, and a transition state. Figure 3 shows the state-transition representation for two episodes in sequence in which a subject tries to accomplish a task step through the *T&E* approach, fails, switches to the *help* approach, and succeeds.

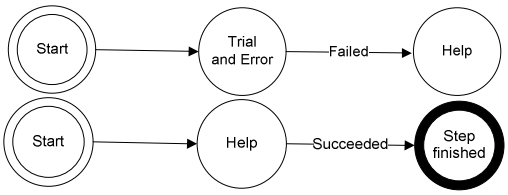


Figure 3. A subject fails using trial-and-error, switches to help, and succeeds

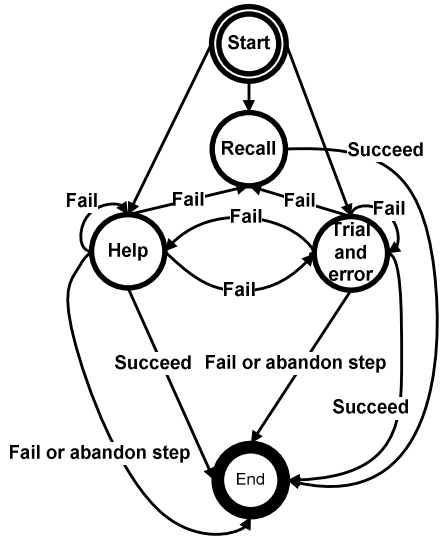


Figure 4. Diagram of transitions between problem-solving approaches

Summing up all the individual state transitions from 124 episodes, we generated a finite state diagram of the transitions among the problem-solving approaches used by the subjects in the study, as shown in Figure 4.

Users reached the end state when they succeeded in using any of the problem-solving approaches and switched to another approach when the current one had failed. *Recall* was by definition successful; if subjects tried to use a remembered feature but failed to accomplish the task step, the episode was coded as using the *T&E* approach.

4.2 Possible Causes

For each episode, we analyzed the possible causes of transitioning from one strategy to another. Our analysis was based on six elements:

- What the subject was looking for when typing in help or exploring the interface. For exploring, we were able to infer what the subject was looking for because we had the task specification at hand from the previous study [13] and watched up to one additional minute forward on the video to see on which step she was working.
- Whether the subject's activity was directly related to accomplishing the task, or if it was for some other purpose, such as trying to understand instructions that were not clear on a previous help topic.
- The result of the subject's consulting help or searching for the feature. This involved following the text and mouse-overs that followed on the video.
- The amount of time of the episode.
- The reasons for the subject's success or failure in the episode. We noted the point at which the subject failed and looked at the actions that preceded the failure.
- Whether the subject was able to apply any help she received and the reasons for this.

We used these elements and the state transition diagram to classify transition patterns and to identify the factors that appear to lead to the transitions, as discussed in Section 5.

4.3 Choice of Approaches

As the subjects interacted with Publisher and its help system, they began each task or major task step with an initial approach. Table 1 presents the distribution of the approaches used in these initial episodes. Subjects could transition between episodes with the same approach; for example, a subject could end a *T&E* episode and transition to a new *T&E* episode if she started a new step with in the task. Similarly, a subject could end a help episode and start a new help episode if she searched for a different term.

Some initial episodes were directly successful—the subject completed the task step without having to transition to another episode. We note that the recall approach was successful by definition, so its high success rate in this table is expected.

For episodes in which the use of an approach was not successful, subjects would have to transition to another approach or abandon the task step, as shown in Table 2. The table presents the sequence of approaches involved in the transition, the number of episodes resulting from the transition that succeeded or failed for the task step, and the rate of success for resulting from the transition (i.e., the percentage of episodes into which the subject transitioned resulted in task success).

The subjects' most frequent transitions were from the *T&E* approach to the *help* approach (25) and from the *help* approach to *T&E* approach (24). In both cases, seven of these transitions led subjects to succeed by switching approach. The transitions to the *recall* approach were successful by definition. Otherwise, the transition with the highest resulting success rate was the transition from the *help* approach to the *T&E* approach; 42% of these transitions resulted in success for the task or step.

Table 1. Initial Episode Successes

Approach	Instances	Successes	Success Rate
Recall	14	14	100%
T&E	30	15	50%
Help	3	3	100%
Total	**47**	**32**	**68%**

Table 2. Transitions between Approaches

Transition	Instances	Successes	Success Rate
Help → T&E	25	7	28%
T&E → Help	25	10	40%
T&E → T&E	10	1	10%
Help → Help	11	3	27%
Help → Recall	3	3	100%
T&E → Recall	3	3	100%
Total	**77**	**27**	**35%**

5. ANALYSIS

We now turn to an analysis of the subjects' patterns of problem-solving, their choice of approach, and possible strategies for transitioning between approaches.

5.1 Initial Approaches

When subjects started a new task, they began with one of the initial episodes summarized in Table 1. The *recall* and *help* approaches had much higher success rates than the *T&E* approach. The reasons for this difference are due to the definition of recall, subjects' perception of the appropriateness of help, and the actual difficulty of completing the tasks using exploratory methods.

Recall. If subjects understood the task step and remembered an appropriate way to accomplish the task, then they would of course use the *recall* approach. In practice, subjects probably experienced degrees of recall. That is, they might have recalled something closely similar to, orthogonal to, or simply hinting at to the new task. In this study, we coded all of these as recall.

In our study, the subjects used the *recall* approach 14 times out of 47 initial episodes. Our analysis of the individual episodes suggests that the necessary knowledge of task and application came from the subjects' initial training, from the subjects' prior use of a similar function earlier in their session, and, increasingly as they continued the session, from the subjects' prior use of exactly the needed function.

If subjects were not aware that they had the necessary task and application knowledge to take advantage of *recall*, then they would have to start with either the *T&E* or *help* approach.

T&E. Episodes using the *T&E* approach amounted to about 64% of all the initial episodes. Of these initial *T&E* episodes, 50% were successful. However, this success rate reflects a higher success rate for work-around solutions and deeper solutions. In work-around episodes, subjects succeeded (again as a matter of definition) in 100% (13 out of 13 episodes) of the initial episodes. In *T&E* episodes where the subjects attempted to do better than a surface solution, the success rate was only about 12% (2 of 17 episodes). This suggests that subjects applied approximate methods if they could, and that, naturally enough, solutions through these approximate methods were easier or faster to achieve than solutions through methods that would have been more appropriate because, for example, they would be more robust for future situations. Moreover, in the cases of the *recall* and *help* approaches, and in the case of the work-around category of *T&E* approaches, the subjects apparently had enough knowledge to proceed reasonably directly to a solution. It is in all of the other cases—those that fall into the category of using exploratory learning to achieve the more robust solutions that experts would apply—that the subjects faced problems that posed substantial difficulty. Thus it is not surprising that the subjects' success rate was only 12% for these cases.

Help. When used as an initial approach for a task step, help was always successful, but subjects did this in only 3 of their 47 total initial episodes—and these were the 8 of 22 subjects who used help the most. In all three instances, these were cases of subjects who searched the help system and used the information in a help topic to complete the task step, rather than cases where the subjects returned to the training materials. One interpretation of this result is that the subjects were aware of their knowledge levels and chose to consult help only when they knew enough to formulate an appropriate query. Alternatively, subjects may have under-used help because they overestimated the utility of the *T&E* approach [15]. Indeed, subjects may overestimate the utility of *T&E* in part because, from their perspective, a work-around is an acceptable solution. That is, the number of perceived solutions through *T&E* is much higher than the number of actual, deeper solutions, leading to the perception that the success rate for *T&E* is 50% rather than 12%.

5.2 Transition Approaches

Where the subjects' initial approach was unsuccessful, subjects transitioned to another episode (or abandoned the task). Subjects could either initiate a new episode of the same approach (e.g., search on a new term, pick a different task step) or initiate a new episode with a different approach. In this section, we first review the results of transitions to an episode with the same approach and then review the results of transitions between approaches.

In 22 of the 77 total transitions, the subjects began a new episode with the same approach as the current episode.

Help → Help. In transitions from *help* to *help*, it appears that the subjects were not finding what they needed in the current help episode, so they refined or changed their query. However, the subjects' missing or incomplete knowledge of the task or the application that led their query to fail in the first place appears not to have been ameliorated by the help topics they had been able to find so far: only one of the *help* → *help* transitions led to a successful help episode, for a successful outcome rate of 10%.

T&E → T&E. In transitions from *T&E* to *T&E*, the subjects typically abandoned a step inside a task and began work a different step in the same task. Subjects could do this because some steps did not depend on other steps in the task. In the "golden apple" task, for example, the steps of inserting the apple image and creating the two-column text box did not depend on each other. As a strategy, this transition was successful in about 27% of cases. This low success rate was likely due to the circumstance that these were the difficult task steps that had not been amenable to *recall* or work-around methods. If there had been a work-around solution for this step, the subject would have already applied that approach.

In all of the other 56 transitions, the subjects began a new episode with a problem-solving approach that differed from the subject's current approach.

Help → Recall. Of the 37 episodes where the subjects transitioned from *help*, in two instances the subjects transitioned to *recall*. These relatively rare cases occurred where subjects did not follow the help instructions but instead for some reason recalled the way to accomplish the step. As in the case of the initial episodes, if the subjects knew how to solve the problem then they did so.

T&E → Recall. Of the 38 episodes where the subjects transitioned from *T&E*, in two instances the subjects transitioned to *recall*. These are cases where the subjects' exploratory learning in the application interface apparently reminded them that they already knew how to accomplish the task step. Because they reflect learning, albeit exploratory, leading to recall of how to do something, these transitions seem similar to those where learning from the help system also led to recall. Indeed, the success rates (2 of 37 and 2 of 38) for the two kinds of transitions reinforce the idea that they reflect a common factor of effectiveness of learning as a trigger for recall.

Help → T&E. There were 25 transitions from *help* to *T&E*. Of these transitions, 40% (10 out of 25) led to successful outcomes in the *T&E* episode. Because, as we have seen, transitions from *help* to *help* (i.e., refining or changing a search) produced almost no direct successes, transitioning to *T&E* was a much more promising strategy. Perhaps this disparity between outcomes is because the subject's lack of knowledge or incomplete model of the task and application would lead to a second search as fruitless as the first, and the subject could learn more effectively from exploration of the interface.

T&E → Help. There were also 25 transitions from *T&E* to *help.* About 28% (7 out of 25) transitions from *T&E* to *help* led to successful episode outcomes.

5.3 Strategy

Given these results, what strategy for choice of problem-solving approach should a rational user of computer applications employ? In sum, the data indicate that:

(a) The initial episode success rate for *help* is 100% (but for N = 3);
(b) The first-episode success rate for *T&E* is 50% (but all except two successes were shallow-solution work-arounds);
(c) The overall transition-to-*help* success rate is 23% (8 of 35); and
(d) The overall transition-to-*T&E* success rate is 34% (12 of 35).

From the standpoint of a user of a computer application, an interesting issue is the choice of what do after a failed *T&E* episode. Should the user stick with *T&E* or switch to *help*? The chances of success in the next episode are nearly equal (27% vs. 28%). Some subjects retried a *T&E* approach in the next episode, but in most cases they ended up switching approaches or abandoning a step after retrying either strategy; in many cases switching led to eventual success.

Our analysis suggests that the rational user should pursue a strategy of changing approaches back and forth between *help* and *T&E* rather than a strategy of sticking with either one. And, in fact, the most common transitions by far were from *help* to *T&E* and from *T&E* to *help.*

It is possible that a strategy of switching approaches enables users to overcome two of the key causes of failure within episodes: vocabulary mismatch and incomplete or mistaken mental models of the task or the computer application.

Vocabulary mismatches occur because help systems are likely to use the same terms as the application [14], which requires users to familiarize themselves with such terms to learn about the application. Users often do not know which words to use when searching for help on a feature [12]. In our study, for example, a subject searched for "Background color" to learn how to add a colored background behind an image but did not find appropriate help because the feature name was "Fill color." Another subject was unfamiliar with the terms "menu" and "toolbar;" when help instructed her to find a feature in the *Objects* toolbar she kept looking for the *Objects* option inside the menus at the top of the application window. Sticking with *help* or *T&E* as an approach may tend to reinforce rather than solve vocabulary mismatches. For example, vocabulary mismatches can lead users into "dead ends" in the interface that look appropriate but are, in fact, not [15]. Instead, switching between problem-solving approaches may enable users to gain a different perspective from which they can resolve vocabulary mismatches.

Likewise, switching approach may also help subjects overcome problems of incomplete or mistaken mental models of the task or the application. In our study, subjects sometimes appeared to have incomplete or mistaken understandings that led to using work-arounds to approximate some task steps, particularly on task 3, the "Golden Apple Advertisement." Similarly, some subjects, while working on task 2, the "Business Card," used work-arounds to change the logo. In other cases, some subjects were not sure that a function existed in the dialog box or toolbar that they were using; this led them to cancel out of an appropriate dialog box or to hover away from an appropriate toolbar. Through interplay of exploratory learning from *T&E* episodes and directed learning from *help* episodes, users might have a greater chance of filling in these gaps in their mental models, such as distinguishing images from text boxes, or grasping that some images are grouped.

6. CONCLUSION

The subjects' experiences in our study suggest that, from the subjects' perspective, more successful outcomes occur through the *T&E* approach than by consulting help, although consulting help does lead to some outright successes and can lead to eventual success in a later *T&E* episode. Our analysis also indicates that subjects are able to achieve successful task outcomes by switching between *help* and *T&E* approaches. These findings have implications for the design of interfaces and help systems, and especially for the interaction between the two.

6.1 Supporting Problem-Solving

In our companion paper [15], we discuss ways of improving the effectiveness of help systems for use within episodes, such as working through the affordances for both application and its help system in tandem, ensuring that the application's vocabulary matches users' expectations, using the help system to provide readers with accelerated understanding of the applications' model, developing interfaces that are the minimal counterparts of the minimal manual [2], and dynamically minimizing complexity in the help system. In this paper, we focus on support for switching between approaches and support for exploratory learning.

The most important lesson to draw from our results is that computer applications can support more effective problem-solving by providing features that enable users to switch more effectively between exploratory learning in the application's interface and consulting the application's help system. Help systems might support switching between problem-solving methods in these ways:

- **Automatic History**. Help systems could provide a history of the user's interaction, visible by default, that provides quick access back to searches, help topics, and particular sections of help topics previously visited.
- **User-Managed History**. Help systems could also support switching by providing "save search," "save topic," "save help section" and similar features that enable users can actively manage a list of helpful topics or leave signposts as to where they left off when learning about application functions.
- **User Annotation of Help**. Help systems could include facilities for annotation so that users could append their own understanding to the topics they search, such as the meanings of unfamiliar terms.
- **Shared Annotations.** If help systems support user annotation, then these annotations could be shared among users, providing much more direct access to the sort of peer-user information for which users now search the Internet. Access to shared annotations could be controlled by users and organizations.
- **Links to Application**. Help systems could provide hyperlinks from help text back to the application interface. This could help novice users who are confused between, for example, a menu item and a toolbar with similar names.

In addition to supporting switching between problem-solving approaches through better help systems, computer applications could be designed with the understanding that most use, especially for novices, will occur through exploratory learning rather than formal consultation of help. These adaptations could include:

- **Tool Tips**. Application interfaces could provide more extensive roll-over help. This might include tool tips, which are more meaningful that function names, that are shown after an extended hover. Microsoft's "SuperTooltip" [5] is a good step in this direction. Providing this kind of just-in-time, right-in-context help should generate much of the benefit of switching problem-solving approaches without the disruptive overhead of invoking and navigating the help system. This approach might guide users to avoid false affordances [6], such as reaching Publisher's "Design Gallery" when they intended to find a gallery of clip art.
- **Context-Aware Interface**. If the user has just consulted help, the application interface could reinforce the affordances—the cues and signposts—that would guide the user to the appropriate part of the interface. For example, the application could show or highlight a toolbar, just referenced in help, to assist the user in finding the toolbar rather than mistakenly reaching another part of the interface that has similar terminology. Similarly, help systems could include better topic-classification algorithms (such as maximum entropy models or the nearest neighbor algorithm) instead of simple keyword-based indexing. It would probably be infeasible to model all possible words for matching searches, but the indexing component could be trained with

synonyms or related words so that users could approximate a reference to a feature even if they do not know the exact term.

- **Context-Aware Help**. Conversely, if the help system were aware of the user's interaction in application interface, the help system could do a better job of suggesting help topics. For example, if a user was working with a text box and searched in the help system for "add border," the help system would begin with help relevant to text boxes rather than, say, tables.

Based on our analysis of the difficulties our subjects encountered and the problem-solving methods they employed, we have these further suggestions:

- Put the procedure early in help topics. Subjects in our study would consult help, scan for a few keywords and then try to use a feature without reading the entire topic. Help systems could address this by putting the procedure first and the explanation, for those who want it, second. The most common scenarios for the use of a feature can be listed at the beginning, and other less likely scenarios could follow. The direction of the user's attention to the procedural parts of the help could be made more direct by using headers that explicitly identify the components of the why-what-how model [10] of help, placing the *how* section first.
- Add guidance for common tasks to help systems. This would be a counterpart to the familiar FAQ: frequently performed tasks. While it would likely be impossible to predict the entire set of tasks that users would want to perform, developers of computer applications could collect data from which the most common tasks, especially for novices, could be identified. Help systems could highlight these tasks, and application interfaces could provide more evident affordances for their accomplishment.
- Help users build their mental models of the application. The application and the help system could teach users to distinguish among the application's basic kinds of components and functions that relate to these components. Techniques could include roll-overs in the help system, even in lists of help topics generated by search, that showed pictures or other simple explanations of the components to which the help relates. For example, a help topic about borders for tables could have a roll-over that shows examples of borders and a roll-over that shows an image of table. Our analysis further suggests that trying to build the users' mental models of the application through the help system topics themselves is less likely to be effective. First, as the research shows, users tend to consult help for pragmatic rather than didactic reasons. Second, if, as we suggested earlier, help topics should have the procedure early, then it would be difficult to put the model before the procedure. One possible solution would be to preface the procedures with minimal explanations of the appropriate contexts of their use.
- Study the expectations and vocabulary of novice users. People come to the application with a vocabulary and semantics born of their experiences in a world other than that of a particular computer application. Designers and writers could alleviate problems, particularly for novice users, by studying what people expect from an application and how they refer to things and concepts and associated with the application. At a minimum, this could be useful in providing indexes of synonyms that could be offered by the help system as pointers, with appropriate explanation, to topics and functions that have the application's own vocabulary and meaning.

Developers of computer applications and writers of the documentation for these systems can suffer understandable frustration when they see users in usability tests muddling through instead of following the obvious affordances of the application's design or scanning help text and returning to the application instead of reading and following the clear procedure provided by the help system. The subjects' behaviors in this study, like the subjects' behaviors in previous studies that show low levels of consulting documentation and help systems, can be seen for what they are: pragmatic actions of people using exploratory learning and actively switching among problem-solving approaches. Developers and writers can help the users of their applications and documentation by understanding, accepting and supporting the users' actual, rather than ideal, approaches to accomplishing tasks in computer applications.

6.2 Limitations

The results reported in this paper, like those in the companion "micro" paper [15], were based on recordings obtained in a study comparing effectiveness of tutorials and reflect the limitations of the recordings.

One limitation is that the four tasks were designed to test a variety of skills, particularly as we sought to have some tasks covered by the tutorials and other tasks not covered. Thus the functions in the tasks differ in ways that led to unanticipated behaviors on the part of the subjects, such as the greater number of work-arounds in task 3. Additionally, although the steps in each task were intended to be relatively simple, many subjects were unable to complete the tasks in the time available. In future work, we would pre-test the tasks to determine that the tasks were all capable of being completed within the allowed time and we might schedule longer sessions, if the subjects were amenable. We would also check that the tasks required reasonably equivalent skills, although this is a difficult constraint precisely because each task is supposed to require new skills.

A second limitation involves the methodology of the sessions. The original study focused on task completion times and task outcomes, so we tried to make the sessions as realistic as possible. This meant, though, that our interpretations of the subjects' mental states was based, other than occasional comments they made to the experimenter, on the subjects' actions in the interface. In future work, we would use a think-aloud methodology that would provide a more explicit indication of what the subjects were thinking.

7. ACKNOWLEDGMENTS

This work was supported in part by an endowment from AT&T. The authors thank Nigel Ward and the SIGDOC 2009 reviewers for insights and comments that substantially improved this paper. We also thank the other members of UTEP's Interactive Systems Group for their valuable comments.

8. REFERENCES

[1] Andrade, O., and Novick, D. (2008). Expressing help at appropriate levels, *Proceedings of SIGDOC 2008*, Lisbon, Portugal, September 22-25, 2008, 125-129.

[2] Carroll, J., Smith-Kerker, P., Ford, J., and Mazur-Rimetz, S. (1987). The minimal manual. *Hum.-Comput. Interact.* 3(2), 123-153.

[3] Ceaparu, I., Lazar, J., Bessiere, K., Robinson, J., and Shneiderman, B. (2004). Determining causes and severity of end-user frustration, *International Journal of Human-Computer Interaction*, 17(3), 333-356.

[4] Dunsmore, H. (1980). Designing an interactive facility for non-programmers. *Proceedings of ACM 80,* Nashville, TN, Oct. 27-29, 1980, 475-483.

[5] Elsehemy, A. (2007). *SuperTooltip*. Available from http://dotnetslackers.com/articles/net/SuperToolTip.aspx; accessed August 5, 2009.

[6] Gaver, W. (1991). Technology affordances, *Proceedings of CHI 91*, New Orleans, LA, April 27-May 2, 1991, 79-84.

[7] Kearsley, G. (1988). *Online help systems: Design and implementation*. Intellect Press, Ablex.

[8] Koenemann, J. and Robertson, S. (1991). Expert problem solving strategies for program comprehension, *Proceedings of SIGCHI 1991,* New Orleans, LA, April 27 - May 2, 1991, 125-130.

[9] Newell, A. (1966). On the analysis of human problem solving protocols, *International Symposium on Mathematical and Computational Methods in the Social Sciences*, Rome, Italy, July 4-9, 1966.

[10] Novick, D. (2000). A why-what-how tool for development and documentation of operating procedures, *Proceedings of IPCC/SIGDOC 2000*, Cambridge, MA, September, 2000, 559-570.

[11] Novick, D., and Ward, K. (2006). What users say they want in documentation *Proceedings of SIGDOC 2006*, Myrtle Beach, SC, October 18-20, 2006, 84-91.

[12] Novick, D., and Ward, K. (2006). Why don't people read the manual? *Proceedings of SIGDOC 2006*, Myrtle Beach, SC, October 18-20, 2006, 11-18.

[13] Novick, D., Elizalde, E., and Bean, N. (2007). Toward a more accurate view of when and how people seek help with computer applications, *Proceedings of SIGDOC 2007*, El Paso, TX, October 22-24, 2007, 95-102.

[14] Novick, D., Andrade, O., Bean, N., and Elizalde, E. (2008). Help-based tutorials, *Proceedings of SIGDOC 2008*, Lisbon, Portugal, September 22-25, 2008, 1-8.

[15] Novick, D, Andrade, O., and Bean, N. (in press). The micro-structure of use of help, *Proceedings of SIGDOC 2009*, Bloomington, IN, October 5-7, 2009.

[16] Rieman, J. 1996. A field study of exploratory learning strategies. *ACM Transactions on Computer-Human Interaction 3*, 198-218.

[17] Smart, K., De Tienne, K., Whiting, M. (1998). Customers' use of documentation: The enduring legacy of print, *Proceedings of SIGDOC 98*, September 24-26, 1998, Quebec, Canada, 23-28.

[18] Smart, K., Whiting, M., and De Tienne, K (2001). Assessing the need for printed and online documentation: A study of customer preference and use, *Journal of Business Communication* 38, 285-314.