ABSTRACT

WHITMAN, LISA. The Effectiveness of Interactivity in Multimedia Software Tutorials. (Under the direction of Dr. Eric Wiebe and Dr. Chris Mayhorn).

Many people face the challenge of finding effective computer-based software instruction, including employees who must learn how to use software applications for their job and students of distance education classes. Therefore, it is important to conduct research on how computer-based multimedia software tutorials should be designed so they are as effective as possible for instructing users on a software application and preparing them to use it for the tasks they will need to perform. This study investigated if comprehension of a software program and performance of accomplishing tasks using the software is aided by incorporating animation and user interactivity in a multimedia software tutorial. Effects of multimedia on user engagement and workload were also explored. The study compared common multimedia types of software tutorials – static, animated, and interactive. Results indicated that interactivity in multimedia software tutorials is effective in improving user engagement, the user’s perception of the learning experience, declarative knowledge of the software program being taught, and ability to use the software program to perform tasks quickly and accurately.
The Effectiveness of Interactivity in Multimedia Software Tutorials

by
Lisa Whitman

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Psychology

Raleigh, North Carolina

2013

APPROVED BY:

_______________________________  ______________________________
Eric Wiebe, Ph.D.                     Christopher B. Mayhorn, Ph.D.
Co-Chair of Advisory Committee        Co-Chair of Advisory Committee

_______________________________  ______________________________
Douglas J. Gillan, Ph.D.              Brad Mehlenbacher, Ph.D.
BIOGRAPHY

Lisa Whitman specializes in human factors research and applying it to the design of usable computer systems, especially instructional technology. Lisa finds it rewarding to help people utilize e-learning to accomplish their educational goals. She strives to research and communicate best practices for distance education and apply her knowledge to designing and developing effective and engaging e-learning programs. Lisa has worked in the field of human factors since 2001, analyzing user experiences and designing user interfaces for websites, software, mobile applications, and instructional technology. She has worked for SAS Institute, Inc. since 2005, currently in the role of Senior Sales Training and Development Specialist designing and developing e-learning programs for sales employees, and previously as a Usability Analyst designing SAS Business Intelligence web-based applications and mobile applications. Lisa has also previously done usability research and design work for Lenovo, Distance Learning Systems Group, and Dunlap and Associates. She has presented her research at conferences including Human-Computer Interaction International and the Annual Meeting of the Human Factors and Ergonomics Society. Prior to studying Human Factors and Ergonomics at North Carolina State University, Lisa earned her Bachelor of Science degree in Neuroscience at Fairfield University and earned a Masters of Arts degree in Human Factors Psychology from California State University, Northridge.
ACKNOWLEDGMENTS

I would like to express my gratitude to the members of my dissertation committee. They have not only been instrumental in helping me develop my knowledge of human factors psychology research, but they have all contributed to increasing my enthusiasm for human factors psychology and instructional technology design. I appreciate my professors for teaching and challenging me.

I am also grateful for having extraordinarily supportive friends and family, who have consistently given me much-needed encouragement throughout my doctoral program. I would especially like to acknowledge my grandparents who have always encouraged me to pursue my goals and inspired me to believe that with enough focus and hard work, any goal can be accomplished.
# TABLE OF CONTENTS

LIST OF TABLES ........................................................................................................... v
LIST OF FIGURES .......................................................................................................... vi

INTRODUCTION ............................................................................................................. 1
  Introduction Summary ................................................................................................. 6

LITERATURE REVIEW .................................................................................................... 9
  Learning with Multimedia ............................................................................................ 12
  Attracting and maintaining attention and engagement with e-learning .................. 12
  Information processing while learning with multimedia .......................................... 17
  Developing procedural knowledge through multimedia tutorials ......................... 42
  Comparing multimedia modalities for software tutorials ........................................ 44
  The potential of interactive software tutorials ......................................................... 49

Hypotheses ................................................................................................................... 51

METHOD ....................................................................................................................... 54
  Participants .................................................................................................................. 54
  Design ......................................................................................................................... 54
  Data Collection .......................................................................................................... 65
  Data Analysis ............................................................................................................. 66

RESULTS ...................................................................................................................... 68

DISCUSSION ................................................................................................................ 85

REFERENCES ............................................................................................................. 90

APPENDICES ............................................................................................................... 95
  Appendix A - Main concepts and associated tasks covered in the tutorials .......... 96
  Appendix B – Informed Consent Form .................................................................. 107
  Appendix C – Initial Experiment Participant Questionnaire ............................... 109
  Appendix D – Declarative Knowledge Test ............................................................ 111
  Appendix E – User Engagement Scale .................................................................. 114
  Appendix F – Online NASA-TLX used in the research study ......................... 115
  Appendix G – Final Experiment Participant Questionnaire ............................... 118
  Appendix H – Debriefing Form ............................................................................ 119
  Appendix I – Scale for scoring the procedural knowledge test ....................... 120
LIST OF TABLES

Table 1  Characterization of the assessment questions .................................................. 59
Table 2  Differences between the multimedia tutorials ................................................ 65
Table 3  User Engagement Scale Performance ............................................................. 70
Table 4  Declarative Knowledge Test Performance ....................................................... 72
Table 5  Procedural Knowledge Test Performance ......................................................... 74
Table 6  NASA-TLX composite scores .......................................................................... 76
Table 7  NASA-TLX unweighted performance scores .................................................... 78
Table 8  NASA-TLX unweighted effort scores .............................................................. 79
Table 9  NASA-TLX unweighted temporal scores ......................................................... 81
Table 10 NASA-TLX unweighted mental workload scores ............................................ 82
Table 11 NASA-TLX unweighted physical workload scores ........................................ 83
Table 12 NASA-TLX unweighted frustration workload scores ..................................... 83
Table 13 Time in seconds that participants spent studying from the tutorial ............... 84
LIST OF FIGURES

Figure 1  Interactive tutorial – Immersing the user in the software user interface ..........  62
Figure 2  Interactive tutorial – Locating components in the UI ....................................  63
Figure 3  User Engagement Scale scores for the three groups .......................................  70
Figure 4  Declarative knowledge test scores for the three groups .................................  72
Figure 5  Procedural knowledge test scores for the three groups ....................................  75
Figure 6  NASA-TLX composite scores for the three groups ........................................  77
Figure 7  NASA-TLX unweighted performance scores for the three groups ...............  78
Figure 8  NASA-TLX unweighted effort scores for the three groups .............................  80
Figure 9  NASA-TLX unweighted temporal scores for the three groups.......................  81
Figure 10  Time in seconds that participants spent studying from the tutorial ...............  84
INTRODUCTION

Computer users often have to learn new software programs in order to accomplish personal or professional tasks. People may find themselves having to learn software that comes with a new electronic device in order to use it, or become accustomed to a newer version of a familiar software application in order to take advantage of recently added features. Students may need to learn new software applications when they are taking classes. Many students find that they have to learn how to use software on their own if they are taking college courses through distance learning or if a college does not provide time for students in a class to learn the software in a computer lab together. A company may decide to use a new software application as a standard for accomplishing a task, and employees need to learn the software application quickly before productivity starts to decrease, but thoroughly enough to do their job well. New employees may find it daunting to be hired on to a new position if they are faced with the challenge of learning software applications that they will be required to use. The employer or the employee needs to find an effective way of learning the necessary software applications, so that the new employee can start doing his or her job as quickly as possible.

There is a diverse multitude of ways to learn about software applications and how to use them to accomplish tasks for personal use, school, or work. Technological advancements in instructional delivery methods and the increasing need for software training for jobs and distance education college students have moved instruction from manuals and classroom instruction to electronic formats including Web-based instruction, and software tutorials.
Help features are built into many software programs, providing the software user with tutorials or step-by-step instructions on how to accomplish common tasks and provide answers to frequently asked questions. Many software applications refer users to on-line help by providing links to the software company’s website. Computer-based tutorials are becoming an increasingly popular means of training (Gist, Schwoerer, & Rosen, 1989). Tutorials on many software applications can be purchased, and an increasing number of software applications include tutorials as a help feature or provide them on the support section of their website. Software tutorials are a type of computer-based instruction that usually contains illustrated examples, instruction, structured exercises, and feedback based on the user’s performance (Gist et al., 1989).

There are many advantages to providing software instruction in an electronic format (Atlas et al., 1997; Carpenter, Watson, Raffety, & Chabal, 2003; Reclam & Sexton, 1994; Steele-Johnson & Hyde, 1997). One advantage of computer-based instruction is convenience. Software tutorials included with software or available online are always available to a learner, unlike classroom instruction. Since they are located on the computer, they are also less bulky than manuals or books. Software tutorials are also easier to distribute and update, giving users easy access to the most current instructional materials. Also, computer-based instruction has capabilities that printed instruction lacks, such as providing students with instant performance-based feedback, and the opportunity to practice skills with the software.
If computer-based software tutorials are usable and designed well, then they can effectively teach the user about a software application and how to use it, which will not only improve efficiency and satisfaction with using the software, but financially benefit the software company in reduced product support and marketing costs as well (Atlas et al., 1997). It is in the interest of a software company to provide effective instruction to its users so that users will be able to take advantage of the software application’s capabilities, be satisfied enough with the software to recommend it to others and purchase newer versions, and not have to call for technical support. Since the need is evident for computer-based software tutorials, research should be conducted on how tutorials should be designed so they are as effective as possible for instructing users on a software application and preparing them to use it for the tasks they will need to perform. With the potential that computer-based software tutorials have, their design should be optimized to provide software users with highly effective instruction.

Computers have the potential to provide multimedia training environments that are engaging (Atlas et al., 1997; Carpenter et al., 2003; Hegarty, 2004) and effective (Mayer & Moreno, 2002a; Stemler, 1997). Computer-based training may be a more effective and lower-cost training method over classroom training for some computer users (Atlas et al., 1997). Therefore, further research is needed on how to make computer-based software training as effective as possible. In order for a software tutorial to be considered effective, it must: 1) help the user comprehend the purpose of the program, 2) introduce the user to the software user interface layout and tools available to perform tasks, 3) help users understand
the program’s capabilities, and 4) help the user learn the procedures and skills for performing
tasks. An effective tutorial will teach the user how to eventually use the software to perform
tasks and help the user acquire the skills necessary to perform the tasks without the tutorial’s
assistance. Part of the key to the success of the software tutorial will be effectively engaging
the user in the cognitive tasks needed to master the software tool.

Tutorials can be described as being designed in one of three different forms of
multimedia presentation: static, animated, and interactive. Static tutorials contain written
instruction, and sometimes include graphic illustrations and audio instruction. The user can
move from one section to the other as they view the tutorial and learn about the features of
the software program that they are interested in. An animated tutorial includes movement of
objects on the display with the intent to better demonstrate how one would perform an action
using the software. For example, the tutorial could include a video showing a screen shot of
the software application with the mouse cursor moving toward a menu, then selecting a
feature, the consequent dialogue box appearing, the mouse selecting the appropriate
response, and the result of that action. Animated tutorials allow the user to watch while the
tutorial demonstrates what the display should look like when the user is performing the
action on their own using the software. Interactive tutorials are also used to teach software.
After viewing animated or static instruction, an interactive tutorial allows the user to interact
with the software application in order to practice what they are being taught. For example, an
interactive tutorial, like Ribbon Hero for Microsoft Office (Microsoft Office Labs, 2011),
will instruct a user on the step-by-step process of performing various tasks involved in
editing a Microsoft Office document. Ribbon Hero will provide the user with a work file to open in an Office program so the user can use the program’s tools to practice editing the document based on what they learned. If the desired result is not obtained, the user can refer back to the tutorial and click on links to view images that guide them to the tools that they need to use to perform the task. An interactive tutorial like this provides instruction, gives the user an opportunity to practice, and provides additional help if it is needed.

While software tutorials hold the potential to be effective learning tools, if they are not designed well, then they can pose disadvantages to the user’s learning experience (Hegarty, 2004). For example, a poorly designed tutorial could increase a student’s cognitive workload. If the material being taught involves understanding how certain objects being changed can affect a system, it may be hard to understand these changes and effects if the material is taught in a static format. The student would have an increased cognitive workload as they try to visualize changes over time and corresponding effects in their head, and this increased workload could lead to cognitive overload as they try to comprehend the new material. Animation can help relieve the cognitive workload of visualizing changes to a system over time, but if the user is trying to follow along with an animation that is going too fast for them, then it can lead to increased cognitive workload that could interfere with learning. It is not enough for a tutorial to take advantage of the capabilities of computers and multimedia. Its design must be carefully considered and based on human factors research in order for the tutorial to be effective at instructing the user.
In order for a software tutorial to be designed well, the designer must know the goals of an effective software tutorial and apply psychology research to understand what makes a tutorial effective. An effective software tutorial will help a user understand how a software application works and give them the ability to perform required tasks on their own. In order to do this, tutorials must first be able to draw and maintain the user’s attention enough that they are motivated to use the tutorial and learn about the software. Secondly, effective tutorials should also be designed in a way that promotes learning, based on cognitive psychology research models, in order to promote efficient storage of information in long-term memory through schema construction and automation, and decrease extraneous cognitive workload. Finally, effective tutorials should be designed to optimize the capabilities of multimedia in order to help the user be able to learn, remember, and apply the concepts and skills taught in the tutorial. Since all of these factors must be considered in the design of a software tutorial for it to be effective, the application of research on how people learn, and specifically how people learn with multimedia instruction, is needed to design effective software tutorials (Hannafin & Kim, 2003; Hung, 2001; Langhorne & Swierenga, 2004; Winn, 2002; Yan, Hao, Hobbs, & Wen, 2003).

**Introduction Summary**

Discovering ways to improve training effectiveness is an important area of research. Traditional instructional delivery methods for teaching computer users about how to use software applications, such as manuals and classroom instruction, are still popular choices for instruction, but they may not be the best methods for teaching users how to understand an
application enough to be able to use it on their own to accomplish necessary tasks.

Computer-based learning tools have the potential to instruct users in ways that other materials cannot. Computers can provide multimedia instruction, engaging different senses with audio instruction, visual displays, and physical interaction. Research has shown that the use of multimedia in computer-based instruction has the potential to capture a user’s attention and increase their learning performance (Zolna & Catrambone, 2008). Computers also allow for animation so that students can see an analogous representation of the software being used exactly how they will use it. For example, a tutorial can show a video of what it would look like when the mouse moves across the screen and selects tools in the software’s user interface. In contrast to other forms of study aids, such as manuals or guides, computers have the potential to offer unique capabilities and opportunities for the creation of interactive elements between the student and the subject matter, giving students the ability to manipulate objects on the display and see the subsequent outcomes. For example, students can be instructed on how to accomplish a task using a software application’s tools and then be given the opportunity to practice the task using the software application. In addition, the computer can provide feedback on the student’s accuracy and level of understanding of the material through responses to the student’s interactions with the computer program. The effects of multimedia, tutorial design, and the capabilities of computers on computer-based instruction and software tutorials have yet to be fully explored. Knowing more about how the capabilities of computer tutorials affect learning will allow the computer to be utilized to its greatest potential as a learning and training environment. The purpose of this study was to
determine if comprehension of a software program and performance of accomplishing tasks using the software is aided by incorporating animation and user interactivity in a computer-based software tutorial. Also of interest is whether animation and interactivity raise perceived user engagement in the tutorial and whether this enhanced engagement leads to more effective training.
LITERATURE REVIEW

A review of psychology research will support that, in order to be effective, a software tutorial should: 1) engage the user enough so they maintain their attention on the instructional material and are motivated to use the tutorial and learn about the software, 2) provide effective instruction that minimizes extraneous cognitive workload and promotes efficient information processing for storage of information in long-term memory, and 3) teach the user the procedures and tools involved in performing tasks using the software program. Of particular interest is design strategies that can be used to present demonstrations of software use and whether animated and/or interactive elements in a software tutorial provide a mechanism for improved learning. In order to produce an effective software tutorial that will maximize user comprehension of a software program and promote good performance in accomplishing tasks using the software after using the tutorial, literature research must be conducted in areas including human factors and multimedia learning, and then applied to tutorial design.

By exploring research in these areas, software tutorial designers can address the unique challenges of designing effective multimedia instruction in a way that will promote learning and performance for the tutorial user. A software tutorial designer needs to make sure that the tutorial is meeting the needs of the user, and not just including the latest technologies without thought on how the technology affects the learning experience. Referring to research in human factors and multimedia learning, a tutorial designer can check that the tutorial meets the requirements to satisfy the cognitive needs of the user so that
comprehension and performance are enhanced. Research can also reveal to tutorial designers ways in which they can improve their design to be more effective.

Effective software tutorial production is not only dependent on tutorial designers reviewing research, but it also relies on researchers. Empirical research should be driving the design of software tutorials. In order for that to be accomplished, researchers must address the questions necessary to understand and meet the needs of software users. Researchers need to understand what the users are using the software for, how often they will be using the software, and by what means they are able to gain instruction on how the software works. Sometimes people do not use a software program frequently, so software tutorials need to be effective in order to maximize retention of information related to important aspects of the software. Computers are becoming a common way for many software users to obtain instruction (Atlas et al., 1997; Gist et al., 1989; Sims et al., 2008), therefore researchers need to investigate the optimal ways of utilizing a computer environment for providing instruction to software users. This requires research in psychology and education, specifically research in multimedia learning, in order to understand how learning is influenced by different multimedia modalities, and how it can be enhanced by multimedia instead of negatively affected by it. Researchers need to apply supported theories in psychology, education, and human factors in order to design experiments to test relevant hypotheses and develop guidelines for tutorials design. Researchers also need to notice what types of new technologies are being used to produce software and explore if there are any limitations to their use in instructional environments and determine how their potential can be optimized.
for providing effective instruction. Improving multimedia software instruction demands an iterative process between research and design to support the wide, and persistently growing, population of people who need to learn software in a computer-based environment.

As tutorial design and development have evolved along with technology to include different kinds of multimedia, human factors research can be conducted to provide design guidelines for producing more effective multimedia tutorials (Mayer & Moreno, 2002a). Design improvements based on research are likely to lead to more effective software tutorials (Stemler, 1997). An effective tutorial will help the learner understand how the software works and promote accurate performance of skills learned by providing the ability to perform required tasks using the software without the help of the tutorial. A software tutorial can be designed effectively by using multimedia to attract and maintain the user’s attention to important information, applying theories of information processing to promote effective use of cognitive resources for processing and long-term storage of information and minimize extraneous cognitive workload, promoting the development of an accurate mental model of the software program, and helping the user develop procedural knowledge while practicing skills. Research in these areas can provide guidelines on which multimedia modalities to use - static, animated, or interactive - and encourage methods for engaging users and maintaining their attention, and supporting optimal information processing for storage in long-term memory and retrieval for good performance when the information needs to be used.

Researchers have provided a foundation of psychological theories that can be applied to the effective design of multimedia software tutorials. Regardless of which emerging
technologies are involved in developing a multimedia software tutorial, the design of the tutorial should adhere to the principles established in empirical research in order to be effective in promoting optimal learning and performance, and utilize the multimedia modality that best promotes learner engagement, effective use of cognitive resources for information processing and long-term storage, mental model development, and development of procedural knowledge while rehearsing skills using the software.

**Learning with Multimedia**

Multimedia technology provides instructional designers with a variety of options for delivering computer-based software tutorials. It is important for designers to understand potential benefits and limitations of different technologies and modalities in order to choose the best delivery method for a software tutorial. Psychologists have conducted research comparing different multimedia modalities and various applications of emerging computer technologies in the context of computer-based tutorials in order to develop learning theories and provide guidelines for using technology to promote effective instruction.

*Attracting and maintaining attention and engagement with e-learning*

An instructional designer can manipulate the various ways that multimedia can be utilized to attract and maintain a learner’s attention in a computer-based tutorial. Therefore, it is important that a tutorial is designed based on research on how engagement with multimedia can lead to deeper learning and improved performance. Otherwise, multimedia may accidentally be implemented in a way that distracts the user’s attention rather than focus it on the information that must be learned and keep the user engaged in activities that
promote further comprehension of the subject matter. Useful multimedia draws the user’s attention to important information, maintains their attention on the instructional materials, and motivates them to interact with it. While designing a multimedia tutorial, the instructional designer must think about how to get learners to engage and focus on the instructionally important elements of the tutorial and how the multimedia instruction can be designed so that it is not only engaging, but also effective for instruction. Designing enjoyable and engaging training is important since it is difficult to motivate people to learn more than a minimal amount to get the job done. A training program that no one will use or refer to is useless. Increasing the learner’s motivation to interact with a training program is important, but the training program must also effectively teach the material if it is to be a good use of the learner’s time. (Atlas et al., 1997)

One modality for presenting computer-based instruction is through static words and images on a computer display. Research has shown that the time users will typically spend with static instructions and images, which do not contain animation, is not sufficient to learn how to use the software efficiently (Atlas et al., 1997). Studies have shown that using multimedia and animation in an instructional system increases learners’ motivation to interact with the system. Learners stated that they preferred animated instructions to text-based instructions. Research has also shown that learners pay attention to multimedia examples and refer back to them when solving problems. The examples are beneficial for learning because learners explain the examples to themselves, and effective self-explanation leads to improved learning (Atlas et al., 1997). Czarkowski (1996) found that learners prefer
audio and animation to static graphics and instructional materials without audio. Research has shown that multimedia can keep learners engaged more than passively reading a sequence of screens that contain instructional text and static images by incorporating multimedia and interactivity similar to that of video games, in a method known as serious gaming (e.g., Neill, 2009). The learner is engaged with this type of multimedia learning environment because the animated graphics and sound add a level of realism to the materials that static images and text do not portray, and with realism, the learner feels that the learning environment is relevant to the work environment that they will be executing real tasks in after the training. The learner feels more immersed in the environment when it is realistic and relevant and when interactivity is incorporated to maintain their engagement. Interactivity added should challenge the learner to perform tasks in the simulated environment that are similar to what they would do in their work environment, with hints and tips that can appear to guide the learner. If the learner is advanced, then tips should not appear to distract and frustrate the learner. The multimedia should support the learner’s level of expertise, with animated tips and feedback appearing as they are needed (Neill, 2009). Static graphics and instructional text cannot support the user in this way, so they may get frustrated if they come across sections that they don’t understand or may get bored and start skimming if they are not challenged enough. Multimedia has the potential to keep a user engaged through a learning environment that is realistic, relevant, and provides customized coaching.

Research has shown positive effects of animated displays on learner motivation (Hegarty, 2004). A theory presented by Kennedy (2004), the cognitive interaction model, also
predicted that effective interactive design of instructional content will improve learning by engaging students in deep-level cognitive processing through behavioral processes. Kennedy proposed that by using cognitive strategies to participate in activities and interactions with the instructional materials, students will have greater motivation to learn due to maintenance of interest in the materials through sustained meaningful cognitive processing. Kennedy also stated that students would have increased performance in recall and comprehension of information learned from interactive instructional materials due to interactivity actively promoting students to engage in learning strategies that promote both surface-level processing (such as rehearsal and memorization) and deeper level processing (including elaboration, organization, and critical thinking).

Research shows that active interaction with instructional materials is important to learning, and a valuable method for keeping students cognitively engaged in the instruction. As Kozielska (2000) states, “Learning can be defined as becoming aware of various relationships: the world with the object, the object with its features, a specific action with its results. This happens, primarily, as the result of carrying out specific educational tasks or assignments” (p. 162). Kozielska compared and analyzed the contribution to cognitive activeness of computer assistance in a physics laboratory at a technical university, and demonstrated that students who used educational computer programs showed more activeness in their cognitive behavior on three levels. Three groups of students were compared, a control group that did not use computer assistance, a group that used a demonstrative educational computer program which presented examples upon the students’
request, and a group that used a more interactive dialogue version of the program which responded directly to specific questions asked by the students with information, values, and examples. All three levels of cognitive activeness; problem-solving, research, and creativity; were enhanced with computer assistance over the control group. The demonstrative program aided activeness in problem-solving the most, and the dialogue program was the most helpful in assisting research activeness and accuracy. Kozielska’s (2000) research on cognitive activeness showed that student behavior was positively affected by interactivity, and it emphasized the importance of exploring the possible instructional benefits of interactive computer-based learning.

As technologies have emerged that allowed for user interactivity with computer-based tutorials, psychology researchers have developed experiments to investigate the potential advantages to utilizing these technologies. Results have revealed that interactive multimedia instruction can be more engaging than static or animated multimedia instruction. Mar, Chabal, Anderson, and Vore (2003) conducted a study that looked at user preferences of an interactive tutorial that instructed nurses on pain assessment. The tutorial provided its students with interactive multimedia instruction, that allowed them to go through simulations, receive feedback, and practice skills that were learned. The twenty-five nurses who took a usability questionnaire rated the program favorably for ease-of-use and learning method. Studies like the one described above show the promise of interactive tutorials for engaging students with the multimedia material. Carpenter, et al. (2003) found that an interactive software tutorial engages learners and keeps their interest. Participants who used
their computer-based tutorial gave positive feedback including comments such as “simple and straightforward, it made learning easy”, “challenging and engaging” and “interesting, easy to use, and informative.”

An interactive tutorial has the capacity to give students the most control over how they will view the materials, allowing them to select when they view animations, what animations they want to see, and if they want to replay an accompanying audio narration. By being able to interact with the instructional presentation, a student is enabled to design the presentation based on how they want to study and their learning pace and preferences, while the computer acts as a guide and tutor by showing the student what choices they have for material presentation modes and allowing the student opportunities to enhance and customize their learning experience through the manipulation of their study materials. Interactive tutorials may also help a user maintain interest in the subject, as well as provide motivation by discouraging passive learning and allowing the student to take a more involved and hands-on approach to grasping a concept. Research suggests that interactivity will expand on the instructional potential of animation by allowing students to manipulate and control the animations in order to better perceive and comprehend them (Tversky & Morrison, 2002).

When investigating the best way to design multimedia software tutorials in order to maximize usability and effectiveness, it is essential to include research on maintaining the user’s attention on important information (Carpenter et al., 2003; Hegarty, 2004; Shah & Freedman, 2003; Najjar, 1998).

*Information processing while learning with multimedia*
Psychology research that employs an information processing model can reveal the way the brain works as someone studies material from a tutorial, providing insight for guidelines for designing effective computer-based multimedia tutorials. A limited amount of information can be stored in working memory as it is processed, so it is essential to understand the literature on information processing to know how working memory can be used efficiently for processing information and storing it in long-term memory. Instructional materials need to be sensitive to processing limitations of working memory and allow enough cognitive resources for schema construction and transfer of knowledge. Dynamic visualizations and interaction with them can be useful, but only if they are directly related to learning and consider the learner’s knowledge base and prior experience (Chandler, 2004).

Understanding how information is stored in long-term memory is important for effective software tutorial design since the goal of software training is to support user performance after the training and promote good long-term performance of the skills learned (Atlas et al., 1997). In the extensive amount of psychology research in this area, the literature includes four theories that are especially relevant to information processing in the context of learning from computer-based multimedia tutorials. These theories are dual coding theory, cognitive load theory, multiple resource theory, and multimedia learning theory.

According to Paivio’s Dual Coding Theory (Paivio, 1986), cognitive behavior (including encoding, organizing, transforming, storing, and retrieving information) is mediated by two independent systems, a non-verbal (image or visual) system and a verbal system. The image or visual system handles visual information and the verbal systems processes linguistic
information. Learning is considered to be better when information is processed through the
two channels rather than through one channel alone. (Michas & Berry, 2000; Shah &
Freedman, 2003)

Cognitive load theory (CLT) is another information processing theory that can lend
insight to instructional design. For a software tutorial to be effective, extraneous cognitive
load should be minimized so the learner can have more working-memory resources to devote
to learning the instructional material and schema acquisition. Material in working memory
can be transferred to long-term memory in the form of schemas, which can vary in their
amount of automaticity. Schema construction and automation reduce the load on working
memory (Sweller, Merrienboer, & Pass, 1998). Reducing extraneous cognitive load will
allow for more resources to focus on developing schemas from the knowledge in working
memory using germane cognitive load. The more germane cognitive load available, the more
schemas can be created, decreasing intrinsic cognitive load for processing elements of
information. Decreased intrinsic cognitive load is the result of advanced schemas being able
to more efficiently process information being received, setting the stage for using available
working memory capacity to build even more advanced schemas. Scaffolding can be
provided to the learner through a fading technique, by giving worked examples at first,
followed by completion problems, and then full problems. Learners can also be assisted in
forming schema if general information precedes procedural information (Paas, Renkl, &
Sweller, 2003b).
It is important for instructional designers to understand cognitive load theory and the cognitive architecture of learners since the way that information is presented has the potential to add to cognitive load, leaving less capacity in working memory available for processing new information and schema construction (Shnotz & Kürcchner, 2007). The purpose of constructing schemas is to organize and store the information learned into long-term memory and to reduce working memory load. Since working memory has a limited capacity, but long-term memory has an unlimited capacity, it is the goal of instructional materials to help a learner construct schemas so that information is stored in long-term memory. Working memory is where a learner consciously attends to information that they are receiving. The learner is not conscious of their long-term memory, which hides away information, where it is organized and stored until it is needed. When the information in long-term memory is needed, someone can recall the information to bring it back into consciousness. If the information was stored as a schema with a lot of other information, then all of that information can be retrieved at once without the person having to recall each individual element of the information. For example, someone can recall everything they experienced at a restaurant, without having to separately recall the way the restaurant looked, and then what their waiter looked like, and then what type of food was served, and so on. In this way, a schema helps to decrease cognitive load in a person’s working memory, since instead of having to remember several items individually, they are remembering one single unit of information. Since working memory is only capable of holding about seven pieces of information at a time, a schema helps a person to have capacity in working memory available
Besides just the schema that is being recalled. A person can hold a limited amount of information in their working memory, but if they are actively attending to information in order to learn it, including processing novel information and trying to organize and store it, then working memory only has capacity for two or three items (Paas et al., 2003b; Sweller et al., 1998).

A tutorial should be designed to help the user focus on relevant information that they need to learn in order to help them more effectively and efficiently process, organize, and store the information as a schema that can be recalled later. Once someone has become an expert, then years of exposure and practice will allow them to store larger and more complicated sets of information elements as a schema. However, as someone is first learning a piece of information, there is more demand on working memory to learn the unfamiliar information, so smaller pieces must be processed and stored at a time. The more elements that a schema consists of, the less load it causes on working memory because it takes less capacity in working memory and contains many elements of information.

Since someone who is learning new information can only attend to a couple of items in working memory, it is very important that multimedia instructional materials allow working memory to work efficiently and not overwhelm it. It is very easy to overload the cognitive capacity of working memory as someone is learning new information. According to cognitive load theory, there are three different ways that cognitive load can add up to fill the capacity of working memory while someone is learning. Therefore, it is important to minimize cognitive load in any way that does not contribute to the processing of information.
and schema construction. Those who research cognitive load theory have been able to devise useful guidelines and methods for instructional design because they understand how a learner’s working memory attends to and stores information that is being learned (Paas et al., 2003b). If information that is being learned has high element interactivity, then it is difficult to learn and understand because there are many elements and connections between those elements, but only a limited amount of space in working memory. Something with high element interactivity like this would have high intrinsic cognitive load, meaning that the information is so complex that the act of trying to understand it takes up a lot of the capacity of working memory. The demands that a complex piece of information puts on working memory are intrinsic to the information itself. The only way to diminish intrinsic cognitive load is to break down the elements of a complex system so that there are less elements and connections, and parts of the system can be learned separately in isolation. Eliminating elements and connections of the system while it is being learned can compromise the level of understanding that a learner has of the entire system, but if a system is too complex with very high intrinsic cognitive load, then breaking down the system into smaller elements might be the only way that a learner who is new to the material can begin to comprehend it. Once the learner understands the different elements, and then has time and experiences to understand how they interact together, the learner can become an expert, and start developing a more sophisticated and complete schema of the entire system. When someone becomes an expert, the system will have less intrinsic cognitive load because the user will understand it at a level that starts to seem less complex to them.
Another type of cognitive load defined in cognitive load theory is called extraneous (or ineffective) cognitive load, and should be reduced as much as possible (Paas et al., 2003b). Intrinsic cognitive load is relatively fixed, since it is determined by how complex the information is that is being processed and how much element interconnectivity there is. Extraneous workload, on the other hand, can be diminished by presenting information in a way that efficiently allows for processing and storing of information. If an instructional designer does not take care to design a tutorial in a way that takes into account the limited working memory capacity of learners and how schema construction occurs, then they run the risk of introducing extraneous cognitive load into the instructional materials, giving the learner less working memory capacity to devote to the intrinsic cognitive load which could result in the learner not completely understanding the full intrinsic complexity of the system being learned. Instead of devoting conscious effort to learning about the system, the learner is distracted by irrelevant elements that are not assisting with schema construction. For example, if the instructional material is presented in a way that is unclear or does not fully explain a concept so that a learner has to search for and refer to other instructional materials, then extraneous cognitive workload is increased. The process of searching for the information to make the instruction delivery clear demands capacity of the learner’s working memory that could otherwise be devoted to learning the new information. Therefore, it is a more efficient use of a learner’s cognitive workload if instructional material is presented in a way that is clear so they can efficiently develop schema of the elements in the information. Allowing the user to be distracted with irrelevant information can also increase the learner’s
cognitive workload since a portion of their working memory capacity would attend to the irrelevant information, leaving less capacity for schema construction of the important information. Therefore, instructional designers should avoid using irrelevant materials in their presentations, including stories, graphics, animations, or pop-up windows that do not relate to the important information that should be focused on. Since cognitive load is additive, if the system that is being learned is complex and it has high intrinsic cognitive load, then it is especially important that extraneous cognitive load be decreased as much as possible since the high intrinsic cognitive load will require as much of the capacity of working memory as possible to be understood, processed, and stored into long-term memory. If a learner is being presented with information that is familiar to them or is less complex, then the intrinsic cognitive load of the material is low, and extraneous cognitive load will have less of an impact on learning performance.

There is a third type of cognitive load, called germane cognitive load. It is another type of cognitive load that the instructional designer has control over, but unlike extraneous cognitive load, it is effective in helping the learner to process and store information. A learner experiences germane cognitive load when the instructional materials require the user to utilize working memory toward an instructional process that will enhance learning and schema production. For example, the instructional designer can introduce the learner to a game, challenge, or assessment designed to motivate the learner or provide practice applying the instructional material. Such a process would require the user to devote extra working memory capacity, but it would be an efficient use of working memory because the effort
would help the user understand the material better and result in better schema construction (Paas et al., 2003b). Since the three types of cognitive load are additive, if too much working memory is taken up by extraneous cognitive load, then there will not be much remaining capacity for germane cognitive load, which could compromise the level of understanding that a learner would be able to achieve. Therefore, an instructional designer can increase learning potential by designing the tutorial to reduce extraneous cognitive load.

An instructional designer should evaluate their design to ensure that it is providing instruction as efficiently and effectively as possible. Measuring the effort that a learner puts into learning, as well as their performance, can give the designer an idea of how much effort (working memory capacity) was required to produce the output (knowledge demonstrated on a test). A smaller amount of effort needed to demonstrate successful learning would suggest that the instructional materials were designed efficiently. If a large amount of effort was measured, but successful learning was obtained, then the instructional designer would learn that they should adjust the instructional materials to be more efficient by decreasing extraneous cognitive workload. They can analyze the tasks that the learner performs to see where extraneous effort is spent, and then work to reduce the amount of extraneous cognitive load that is interfering with learning (Paas, Tuovinen, Tabbers, & Van Gerven, 2003a).

A multiple resource theory was developed to explain how resources are used for perceptual and cognitive activities, involving processing stages, perceptual modalities, visual channels, and processing codes (Wickens, 2002). Multiple resource theory addresses how workload is affected by multi-tasking. This theory explains how resources can be devoted to
multiple tasks in terms of three dimensions of information processing. One of the three
dimensions, called the stages of processing, states that resources used for perceptual and
cognitive tasks are different than the resources used for selection and execution of tasks. The
next dimension, the codes of processing, states that tasks involving spatial activity use
different perception, cognitive, and action sources than verbal/linguistic activities. The third
dimension, the modalities dimension, states that tasks involving auditory perception use
different resources than tasks involving visual perception (Wickens, 2008). The dimensions
can guide a designer on how to reduce workload for a user when they have to multi-task
while using a system. If two tasks that a user participates in at the same time use different
levels along the three dimensions, then time-sharing will be better and workload will be less.
However, a designer needs to take into account the particular tasks that the user will
participate in when considering if multi-tasking will cause cognitive overload. For example,
if a user is time-sharing an auditory task and a visual task, generally their cognitive workload
will not be overloaded because the information is being processed by two different
perceptual modalities, but if both tasks involve speech perception (listening to speech and
reading), then they may compete for common perceptual resources and code-defined
resources, which could affect performance (Wickens, 2008). Therefore, while cognitive load
type and multiple resource theory help to guide a designer on how they can design tasks in
a tutorial to reduce workload, the designer still needs to evaluate the tutorial and assess user
performance to ensure that the particular characteristics of the tasks are not affecting
performance in real-world tasks.
Multiple resource theory suggests reducing cognitive load when a user is multi-tasking by promoting efficient time-sharing by dividing tasks across the different resources in processing dimensions. For example, if a driver is involved in the act of manually driving a car, they should not also try to engage in manually entering a phone number on their cell phone or driving performance would suffer. Therefore, using a voice command would be a better way for the driver to perform the task of dialing a phone number since the verbal task will not interfere with the manual task. Another example would be if a person is processing important information visually by looking at an animation, then additional important information should not be presented at the same time as text. The additional information would compete on the visual channel of perceptual modalities and could overload resources resulting in poor performance of processing the information. Instead the designer should use cross-modal time sharing, utilizing the auditory channel for the additional information so that the user can see one piece of information and hear the other piece of information. This will more efficiently utilize resources, decrease cognitive load, and improve performance (Wickens, 2002).

Mayer’s multimedia learning theory discusses how learning is improved, cognitive workload is decreased, and working memory is used most efficiently when multimedia information is received through both a visual channel and an auditory channel simultaneously. Working memory has two independent processing channels, one that can processes visual information and another for processing auditory information (Mayer & Moreno, 2003a). Both channels can be used simultaneously to process multimedia
information without increasing cognitive workload. It is when information competes within one channel of working memory, like text and a picture competing in the visual channel, that the learner risks cognitive overload and information is not sufficiently processed. In this situation, the student is forced to choose relevant pieces of information instead of being able to view and process all of it. This was demonstrated by Mayer and Moreno (2002b) who tested the effects of different narrated presentations of a multimedia scientific explanation of lightning formation. They found that learning comprehension was increased when verbal instruction was presented in both visual and audio formats, providing that the animation was not presented at the same time as the verbal information, and thus, not competing for processing in visual working memory. Cognitive workload is increased and information is not stored as efficiently if one channel is forced to process more than one piece of information at once. For example, if the visual channel attempts to process both an illustration and written text (Mayer, 2001). The theory states that e-learning systems should be designed bearing in mind that people process visual/pictorial and auditory/verbal information in separate channels of working memory, people have limited capacity for each channel, and learning results from attending to relevant information, organizing the information, and integrating the information with previous knowledge (Mayer, 2003a).

Research supports that cross-modal time-sharing is better than when attention is being shared by two auditory channels or two visual channels (Wickens, 2002). Research has shown that dual-mode presentation (audio and visual) of instructional information is more effective than single-mode presentation (visual alone) in teaching with static graphics (Mousavi, Low, &
Sweller, 1995) and animated graphics (Mayer & Moreno, 1998). The authors concluded that these results were due to a reduction in cognitive load. Students did not experience cognitive overload in viewing verbal instruction with simultaneous audio because visual working memory and auditory working memory process independently of one another. These results suggest that tutorials should use verbal information displayed as text on the screen with accompanying audio to provide information and steps to be performed by the participants. Also, audio should accompany visuals, including graphics or animations, in order to use dual-mode presentation. Mayer and Moreno (2003a) call this strategy synchronizing, which involves presenting corresponding segments of animation and narration at the same time in order to decrease cognitive load. Decaptioning, eliminating redundant captions from a narrated animation, will also reduce extraneous cognitive load for a user by reducing the amount of information that is competing for a user’s visual attention. A multimedia online tutorial is promoting meaningful learning when a learner is able to efficiently use their working memory (both visual and auditory channels) to select, retain, organize, and integrate, relevant information so it can be stored in long-term memory (Mayer & Moreno, 1998).

These theories can help designers develop software tutorials that improve problem-solving performance. Research has shown that educational displays that required processing in both the visual and verbal channels (i.e. an illustration with an explanation) enabled learners to perform better on problem-solving tests which required the generation of creative solutions to transfer questions.
Theories on information processing based on psychology research has inspired human factors researchers to devise empirical research studies to test predictions on how systems can be designed to minimize workload and improve user performance. These theory-based studies have participants use systems designed with varying characteristics that could affect workload, and then assess user performance to compare how the different designs changed performance. Many research studies have been conducted to see how workload and performance are affected as users process different types of information and as users process information presented in different modalities (Mayer & Moreno, 2003a). Since cognitive load is additive, if the system that is being learned is complex and it has high intrinsic cognitive load, then it is especially important that extraneous cognitive load be decreased as much as possible since the high intrinsic cognitive load will require as much of the capacity of working memory as possible to be understood, processed, and stored into long-term memory. Human factors researchers can apply the research findings to design guidelines to help designers decrease the extraneous workload that a system imposes on a user so the user’s resources can be devoted to intrinsic or germane workload that are involved in learning. Research experiments have revealed a split-attention effect, where learners demonstrate that their visual attention is split between viewing one important piece of visual information and another. Based on this research, it is recommended that designers organize essential information so that two pieces of information do not compete in the visual channel or the auditory channel (Mayer & Moreno, 2003a). Research based on multiple resource theory has found similar results by showing advantages in user performance if attention was
divided across the auditory channel and the visual channel so that users who were engaging in multiple tasks were practicing cross-modal time-sharing. Research found that if intra-modal time-sharing occurred, where a user had to perform two tasks that used the same perceptual modality, performance suffered (Wickens, 2002). The application of these theories is important when designing multimedia tutorials in order for the content and presentation of the tutorials to be designed in a way that is sensitive to the processing limitations of working memory and allows enough cognitive resources for schema construction and transfer of knowledge.

When computer-based instruction is being designed, it is important to consider the impact of multimedia on cognitive load. Knowing what psychology research has shown on the effects of different technology and display types on learning can help an instructional designer choose an optimal format for a computer-based software tutorial that will utilize the technology to its full potential in a way that enhances learning instead of hindering it. Computer-based tutorials that teach software users how to use a software program are presented either in static (pdf or html format with text and static images), animated (including video demonstrations of the software program), or interactive (user-controlled, self-paced) formats. Psychology research on information processing can help a designer choose which format is optimal for promoting learning of a software program.

If a software tutorial is presented in a static format, then it will contain static images of the software program (i.e., screen shots) and its components, as well as text. Visualizations, such as these static screen shots, are beneficial in that they may reduce the cognitive load of
the learner by providing a representation of the information so that the learner does not have to create one mentally (Shah & Freedman, 2003). Research has shown that dual-mode presentation (audio and visual) of instructional information is more effective than single-mode presentation (visual alone) in teaching with static graphics (Mousavi et al., 1995) and animated graphics (Mayer & Moreno, 1998). The authors concluded that these results were due to a reduction in cognitive load. Working memory has a limited capacity, but its capacity can be enhanced if information is processed using both the visual and auditory channels. Therefore, it would be more beneficial to include recorded audio narration to accompany the static text and images for learners who are viewing a static software tutorial. Mentally animating visualizations that are presented in a static format can provide germane cognitive workload, keeping the learner engaged in active learning. However, if the concept being displayed is too complex, then the learner might have difficulty picturing the spatial relationships and may become overloaded trying to visualize a realistic animated version of the visualization in their heard (Hegarty, 2004).

Research in information processing suggests that animated tutorials are better for information processing than static tutorials. Animations can decrease the cognitive workload of a learner by presenting situational dynamics explicitly so the learner does not have to internally animate an illustration and can devote more processing capacity to comprehending the instructional content (Hasler, Kersten, & Sweller, 2007; Hegarty, 2004; Lowe, 2004). However, visual working memory can become overloaded if there are competing visual stimuli, such as an animation and text instruction, as demonstrated by Mayer and Moreno
(2002b), or multiple animations presented at the same time. In this situation, the learner is forced to choose relevant pieces of information instead of being able to view and process all of it. Therefore, a designer must remember to not have information competing on the same channels while being presented in order for the information to be processed efficiently by the learner.

Interactive software tutorials show even more promise for promoting learning than animated tutorials. Animations that are interactive and can be manipulated by learners further reduce the likelihood of information processing overload, as long as the interface for interacting with the animations does not increase cognitive workload. A guide on how to interact with the animations may be beneficial to the learner so that they can devote cognitive workload to the animations instead of the user interface of the tutorial (Hasler et al., 2007; Hegarty, 2004; Lowe, 2004). If animations are too complex or play too quickly, they cannot be accurately perceived, and fail the apprehension principle of good graphics design (Tversky, Morrison, & Betrancourt, 2002). If the animation is too complex or plays too quickly, the learner will have a difficult time processing the information and then have a difficult time integrating that information with additional relevant information that may be presented later in the tutorial. It puts a heavy demand on the learner to remember the information from the previous animation, and integrate it with information presented later. Static tutorials have an advantage over animated in that the learner can scan the static information to refer back to information that they need to integrate with relevant information that is presented later (Hegarty, 2004). In this way, interactive tutorials have an advantage
over both static and animated tutorials because they provide the benefit of animations, but
provide the learner with control to interact with the animations, so they can more easily
navigate through them and process them at their own pace, allowing them to refer back to or
repeat important information. Interactive animations can provide learners with the ability to
deal with information selectively (by pausing, stopping, and replaying) to avoid excessive
processing demands (Lowe, 2004).

It is important to consider information processing research, the information being
displayed, and the tasks required of a learner when choosing a display type for a multimedia
software tutorial. Based on psychology literature on information processing, interactive
software tutorials show the most promise for reducing extraneous cognitive workload, so that
the working memory of learners can focus on important information and developing schemas
for long-term memory storage. Interactive tutorials also help the learner manage the intrinsic
cognitive workload by presenting complex information in a way that is self-paced, so the
user can pause or replay animations that they need to refer back to as they process and
integrate information. Interactivity allows learners to think about whether they have
sufficiently learned from an animation, and gives them the control to replay the animation if
they don’t feel confident that they have learned the material (Scheiter, Gerjets, &
Catrambone, 2006). Interactive tutorials also provide the opportunity for learners to use
germane cognitive load by engaging in interactions that allow them to solve problems with
the material in step, seeing how one interaction affects the display and guiding them to the
next interaction. In this way they are guided to practice interactions that are similar to tasks
they would perform in real-world tasks, providing practice and helping to automate schemas of integrated information on the software program.

Research on mental models and how they are used by learners who are processing and storing new information should be considered when designing multimedia software tutorials. Developing an accurate mental model of a software program will help a user understand the layout of the program, the location of tools, and how to perform a task using the program. Developing an accurate mental model of the program improves the user’s ability to perform tasks with the program (Ben-Ari & Yeshno, 2006). This is because the more complete a user’s understanding of a program is, the more they will be able to know which tools in the program they should interact with to produce desired outcomes. As a user learns a new piece of software, they develop a mental model of what features the software program contains, how the software program works, and why it works that way. The more that the user learns the program, the more they build on their mental model with new knowledge and experiences with the program, developing an understanding of how different parts of the program interact and hypotheses on what the results of their interactions with the program will be. A user’s interactions with the program are guided by their mental model, and as they interact with the program, see how it behaves, and observe the effects of their interactions with the program, the user develops a more complete understanding of the software program and a more accurate mental model (Potosnak, 1989). An accurate mental model of a software program would consist of an accurate understanding of the components of the software program and
how they work, as well as an accurate understanding of the effects that components have on other components and the consequences of changing components in the software program.

To determine a mental model for a piece of software, an instructional designer could start by breaking down the components - the tools and displays in the user interface. For example, in a program like Microsoft Word, the tool components would include the top toolbar, the ribbon toolbar, the buttons and menus within the toolbars, the tabs of the ribbon toolbar, and the buttons on the task bar at the bottom of the user interface. Tool components would also include right-click context menu options and keyboard shortcut options that are hidden from view, but still available to a user if they want to use them. Display components would include the blank page canvas where the user can type and see the results; as well as user interface components that are added to the display to assist the user, like the blue rectangle that appears when text is selected and the red squiggly lines that are spell-check indicators. When the user has a complete and accurate understanding of the different components in the software program, they have developed a component model (Mayer & Moreno, 2003b).

An accurate mental model would include an understanding of the components and how they work, but also how they interact with each other and the display. For example, an accurate mental model of Word would include the understanding that both the blue rectangle around text (showing that the text is selected) and the act of clicking on the button for Bold would result in making the text that is selected appear in a bold style. When the user has developed an accurate and complete understanding of how components interact with each
other to create changes in the display, the user has developed a causal model (Mayer & Moreno, 2003b).

To develop the software training tutorial, the instructional designer should know accurate information about the components and component interactions involved in the task that is being taught in the tutorial. The instructional designer does not need to cover every component and component relationship at once in the tutorial. Rather than overwhelming the user, it is better to break the description of the system up. One way to break up the components would be to cover a relevant section of a user interface. For example, the instructional designer may want to provide an introductory module in the tutorial that teaches the user about the components in the right-click context menu, and how they affect the words typed on the page display. Or, the instructional designer may want to break up the information by task, teaching the user only about the components involved in the task of creating a cover page and how the components used affect the display. Careful tutorial design is important because if a user has an inaccurate mental model, then it can have a negative impact on performance and the user will commit more errors (Wickens, 2008).

When a user of a software program has completed the two steps of creating a component model and a causal model, they will have developed a mental model of the system (Mayer & Moreno, 2003b). Therefore, a tutorial that promotes the development of an accurate mental model will involve steps that will assist the user in focusing on important components and component interactions, and then organizing, integrating, and storing that information into long-term memory.
Since there are many components that need to be learned, schema construction can assist a user with learning the different components together as a single representation, or schema. For example, in order to be able to more efficiently store and retrieve information on the many buttons related to changing font styles, a user can develop a schema related to font styles. When the font style buttons are learned together as one representation and effectively organized and processed as a schema, then when they are retrieved they are retrieved as one unit in working memory, instead of several separate units that can fill the limited capacity of working memory. If a user has the components for changing font, font size, bold, italicize, and so on processed as one unit, then when they are retrieved the user will still have space in their working memory to notice and process other information because the font style schema will only take the space of one unit in working memory capacity. As the user becomes familiar with other components, he or she can integrate the knowledge of those components into the schema, making the schema bigger so that it not only covers font styles, but also all main toolbar user interface components. It is important that the user’s training promotes schema construction so that when the user is learning about new components or component interactions, they will have as much available working memory space available as possible to devote to processing and storing the new information. If information in training is stored efficiently, then when a user is working with the software program after training, they will have available working memory space to devote to concentrating on the tasks they are performing because the mental model of the program will be organized efficiently into a schema so it only takes up the space of one unit of information in working memory (Shnotz
& Kürschner, 2007). Providing conceptual information is helpful for building a mental framework on which learners can base related procedural information. (Atlas et al., 1997)
The double-fading support approach may also help guide students in building a mental model of software applications (Leutner, 2000). This method involves two types of user support to assist with developing a mental model of a software program as the learner gets more familiar with the program during the course of the tutorial. One form of user support is limiting the functionality of a software program in the beginning of the tutorial, and then gradually providing additional functionality until the learner is using the fully-functional software. The other form of user support removes the detailed guidance, which can slowly fade throughout the course of the tutorial until the user is working with the program unassisted.

When designing a multimedia software tutorial, an instructional designer should consider how the information should be displayed in order to support processing of the information into long-term memory. Animated tutorials appear to be more effective at promoting long-term information storage than static tutorials since research has shown that animations could be useful in the construction of a mental model that includes a dynamic mental representation of a change in time (Lowe, 2004) or a spatial transformation, especially for those with low prior knowledge. This finding seems to hold as long as the animation does not have a high cognitive processing demand and follows the apprehension principle (Boucheix & Guignard, 2005). Animations could be useful for learners who are viewing software tutorials to help them develop procedural knowledge of a task, what actions are carried out to access tools
required to perform a task, and what manipulations of the tools and user interface
components will create the desired outcome. Mayer has done research showing a contiguity
effect, demonstrated by learners generating a median of 50% more creative solutions to
transfer problems when verbal and visual explanations were coordinated rather than
presented separately (Mayer & Anderson, 1992). It is proposed that the coordinated
presentation helps the learners to develop procedural knowledge of the system being
described by helping them build referential connections between the visual and verbal
information, especially if the learner is inexperienced with the system (Michas & Berry,
2000). The synchronized presentation may help learners to use dual coding (using verbal and
pictorial channels) to increase cognitive interconnections between the two forms of studied
information to prior knowledge. Studies have shown that recognition and performance on
problem-solving transfer tests are improved with synchronized presentation of auditory
narration and an animation or movie (Najjar, 1998).

Interaction can lead to better support of information processing than animation alone.
Animations have the potential to help learners build good coherent mental models of
complex change processes since they can depict situational dynamics explicitly. In addition,
interactive animations can provide learners with the ability to deal with information
selectively (by pausing, stopping, and replaying) to avoid excessive processing demands
(Lowe, 2004). Research has found that performance in transfer of training is improved if
instructional animations have controllable speed and playback (Boucheix & Guignard, 2005).
In a study on young subjects learning about the way gears function, Boucheix and Guignard
(2005) also found that animations improve comprehension, especially if the learner has control of the presentation of animations. Allowing the learner to control the material makes it closer to the learner’s pace of integrating information, as well as letting them be more actively engaged in the learning task. Self-control is especially helpful when the processing demand of the task and the cognitive load increase. Learners with low prior knowledge of the material benefited the most from animations and self-control of the presentation. Research has also shown controllable (interactive) animations to be beneficial for helping learners create an effective mental model by integrating a complex mechanical system presented with textual explanation (Schneider & Boucheix, 2006; Tversky & Morrison, 2002). According to Mayer, meaningful learning occurs when learners select relevant information from what is displayed, organize the information into a coherent mental representation, and ingrate that representation with others that are stored in memory (Mayer, 1997).

Previous research suggests that interactive tutorials are superior for promoting efficient and effective information processing over other modes of multimedia. Illustrations assist in information processing, but only if they are explicitly linked to concepts evoked in the text. However, research has not shown better performance with animations due to the difference in information conveyed between static and animated versions and the cognitive difficulty to process information (Boucheix & Guignard, 2005). Interactive tutorials seem to have an advantage in promoting efficient and effective information processing in learners since the ability to control and interact with animations and video has shown to assist in integration of complex material and improve transfer of training.
Developing procedural knowledge through multimedia tutorials

Effective computer-based software tutorials will also utilize multimedia in a way that develops procedural knowledge. Multimedia has the potential to assist learners in developing procedural knowledge as they learn the tasks and procedures involved in using a software program. Research has shown the positive effects of combining verbal and visual information for learning of procedural tasks. Learning of procedural tasks requires understanding of the steps in the procedure, retention of the information, and ability to apply the information by performing the procedure. A study done on instruction of a bandaging procedure showed that the most effective type of instructional presentations were a presentation including illustration with text, and a video presentation (Michas & Berry, 2000). When teaching procedural tasks, it is important to clearly present action information that shows learners how to get from one step of the procedure to the next. Animated presentations have been shown to be better than static presentations for teaching procedural tasks where it is important to convey the sequential actions within the tasks (Michas & Berry, 2000). Training using animated demonstrations closely resembles the physical world method that people use: with observation of an instructor performing a procedure, listening to verbal explanations, and attempting to imitate the procedure. Animation is effective for training procedural skills in a graphical software environment if the animations are designed to promote deep cognitive processing to help the user develop a meaningful conceptual framework, or mental model, of the environment (Atlas et al., 1997).
Interactive tutorials show promise in being more effective than static or animated tutorials in promoting the development of procedural knowledge for learners. Most people prefer to explore a new software program and learn about it by using it (Potosnak, 1989). Interactive tutorials provide the opportunity to practice skills and procedures taught by a tutorial. In problem solving, more practice has been found to increase the long-term understanding of a subject matter (Shute, Gawlick, & Gluck, 1998). Practicing material in a context similar to the operational environment in which it will be applied after it is learned also increases long-term retention, as shown in a study by Conway, Cohen, and Stanhope (1991) involving subjects learning research methods in the context of designing and conducting their own experiments. Learning tasks in a context similar to the operational environment helps users to integrate the knowledge and skills necessary for effective task performance, give them the opportunity to learn to coordinate skills that make up complex task performance, and enable them to transfer what is learned to their work settings (Boot, van Merriënboer, & Theunissen, 2008). Interactive tutorials also have the advantage of being able to offer feedback on the learner’s level of mastery of the concepts presented, and the application of their knowledge.

The human-computer interaction is unique in an instructional context since the computer is assigned the role of instructor, while the user assumes the role of learner. The goal of this relationship is to allow the computer and user to interact in ways that will provide the most effective and efficient way of presenting and teaching the subject material to the user. The design of multimedia instructional materials must be based on research that shows
what types and uses of multimedia are beneficial for learning. The instructional effectiveness of a multimedia lesson should always be the design focus. A software tutorial must help the user comprehend the purpose of the program, the program capabilities, the layout of the program, and the function of the program’s tools. It should also teach the user how to perform a task in a way that will enable the user to use the software to perform the task, or a similar one, in a real-life situation without the tutorial’s assistance.

*Comparing multimedia modalities for software tutorials*

There is a base of empirical research that can be studied to better understand how people can effectively learn from various multimedia instructional presentations. Research has shown that students learn more deeply from well-designed multimedia presentations that contain both words and pictures than from text alone, thus multimedia learning shows promise in improved training over some traditional text-only methods, such as textbooks (Mayer, 2003c). Multimedia research is important to consider in the design of software tutorials. Since users now interact with many software applications through a graphical user interface, a tutorial should use visuals to demonstrate what objects in the GUI look like, where they are located, and what they look like when they are interacted with (Atlas et al., 1997).

Animated materials have been shown to be more effective for instruction than static materials since research has shown that they draw and maintain the user’s attention on information, and help the user store and process the information with less extraneous cognitive workload, in many cases. Research studies have shown a positive effect of
animation in complex system comprehension (Schneider & Boucheix, 2006). Animated diagrams were found to be more effective than text alone in a study based on the instruction of tree diagrams (Cox, McKendree, Tobin, Lee, & Mayes, 1999). Animated diagrams have been found to be more effective than static diagrams in a study that analyzed the effectiveness of instruction of computer algorithms (Catrambone & Seay, 2002). Animations have been used in software tutorials for a while now (Atlas et al., 1997), but is animation effective for learning software usage and improving task performance? Will interacting with animations improve learning? Research has shown that animation may be useful for introducing a software application and procedural tasks to a user, but animation alone may not be enough to help a user remember long tasks.

The unique capabilities of interactivity in computer-based learning have yet to be fully explored, though prior research suggests that interactive multimedia software tutorials will provide increased comprehension and transfer for those who receive interactive instruction over animated and static tutorials that represent the same concepts. While animated tutorials have the capability to demonstrate instructional content in a way that facilitates learning through the illustration of real-time movements across the screen, interactive tutorials go beyond animated by enabling students to choose what they will learn. For example, students can choose which menu in a software program’s user interface they want to see instruction on. Interactivity can also allow students to practice performing a task within the tutorial while receiving instruction and after seeing a video of the task being performed with the software. This gives the student the opportunity to practice what they are
learning since, through manipulation of objects on the display, they can reinforce concepts, such as finding and using certain tools in the software program to perform a task. Interactive tutorials also let students receive immediate feedback about their progress of learning by seeing the results of their manipulations and having the computer check if the outcome is correct. Thus, interactive tutorials may facilitate learning like a tutor, guiding the student and allowing them to choose what they learn, practice what they learn, and receive feedback on their learning progress; unlike static and animated tutorials that simply present information, like a book or video. Based on these ideas, interactive tutorials should prove helpful for students who are learning about software applications.

Research suggests that interactive software tutorials have the potential to be better than passive animated or static tutorials for instruction by being more effective at maintaining the user’s attention, supporting processing and long-term storage of information, and developing the user’s skills to perform the procedures being taught. An interactive tutorial has the capacity to give students the most control over how they will view the materials, allowing them to select when they view animations, what animations they want to see, and if they want to replay an accompanying audio narration. By being able to interact with the instructional presentation, a student is enabled to move through the presentation based on how they want to study and their learning pace and preferences. The computer acts as a guide and tutor by showing the student what choices they have for material presentation modes and allowing the student opportunities to enhance and customize their learning experience through the manipulation of their study materials. Interactive tutorials may also
help a user maintain interest in the subject, as well as provide motivation by discouraging passive learning and allowing the student to take a more involved and hands-on approach to grasping a concept. Learners enjoy being able to try things out with a software application as they are learning about it by performing meaningful tasks that show them how the application works and reacts (Atlas et al., 1997). An interactive educational multimedia user interface can allow learners to control, manipulate, and explore the learning material, and can ask learners to answer questions that integrate the material. Interactive user interfaces seem to have a significant positive effect on learning from multimedia (Najjar, 1998). Tasks that encourage learners to actively process and integrate information may focus their attention on the information and cause them to process the information with more elaboration, especially when the processing focuses on the meaning of the information instead of its appearance (Najjar, 1998). Interactivity should expand on the instructional potential of animation by allowing students to manipulate and control the animations in order to better perceive and comprehend them (Harp, Taylor, & Satzinger, 1998; Hegarty, 2004; Tversky et al., 2002).

Another advantage of using interactive tutorials for computer-based instruction is the opportunity to provide practice. Interactive components of a tutorial can be manipulated by the user so that the user has the chance to practice using their knowledge by applying it to an exercise within the tutorial. The tutorial can be used and manipulated as many times as the user feels they need to practice in order to achieve mastery of the concept. In problem solving, more practice has been found to increase the long-term understanding of a subject matter (Shute et al., 1998). Interacting with animations and the learning material may
increase long-term retention of the content by increasing active learning. Long-term retention is important so that users do not have to repeatedly refer back to the software tutorial. Primary school students who were presented with learner-paced educational animations showed higher test performance on more difficult, high element interactivity questions and relatively lower cognitive load compared to students in groups that received system-paced educational animations (Hasler et al., 2007). Kennedy (2004) proposed that by using cognitive strategies to participate in activities and interactions with the instructional materials, learners will have greater motivation to learn due to maintenance of interest in the materials through sustained meaningful cognitive processing. Kennedy also stated that learners would have increased performance in recall and comprehension of information learned from interactive instructional materials due to interactivity actively promoting learners to engage in learning strategies that promote both surface-level processing (such as rehearsal and memorization) and deeper level processing (including elaboration, organization, and critical thinking).

Interactive tutorials also have the advantage of being able to offer feedback on the student’s level of mastery of the concepts presented, and the application of their knowledge. After the user has manipulated a display, they can instantly see the results. For example, students can use tools and objects within a tutorial to practice editing a graphic in order to practice performing a certain graphic-editing task, like cropping. Feedback given by an instructional computer program in response to a user’s manipulation of the display allows the user to determine their progress in acquiring knowledge and achieving mastery of a concept.
For example, a user can practice cropping a sample graphic in the tutorial and then compare their product to the computer’s example.

Interactive multimedia software tutorials have the potential to be effective if their design is based on human factors research, instructional design principles, and the individual students’ needs. Research is essential to the design of e-learning that takes advantage of the capabilities of multimedia to meet the needs of learners. Additional research is still needed on how to make the latest multimedia technology educationally effective for teaching subject matter, like the use of software programs (Stemler, 1997).

*The potential of interactive software tutorials*

Learning how to use software in a computer-based training format can be convenient and economical for people who need to learn new software programs for school, work, or personal use. Software companies, universities, and third-party vendors are recognizing the convenience of electronic software tutorials for software users, and are producing them at an increasing rate. With an increasing number of electronic software tutorials being produced and used, it is important to evaluate what type of multimedia techniques should be used to create software tutorials that are valuable to users. In order for multimedia software tutorials to be valuable, they must be designed in a way that promotes effective learning of the software application so that the user can quickly develop a level of comprehension that allows them to use the software application on their own without the aid of the tutorial to accomplish their tasks. The goal of this study was to explore factors that lead to the most effective type of multimedia tutorial designs for teaching software. An effective software
tutorial will help the user understand how a software application works and promote long-term performance of skills learned by giving users the ability to perform required tasks using the software without the help of the tutorial. Efficacy of a software tutorial can therefore be evaluated by using a declarative knowledge test to demonstrate a tutorial user’s understanding of the software program and a procedural knowledge test, which asks users to perform a task on their own that is similar to the task taught by the tutorial using the same software program. The features of a tutorial that are likely to lead to more meaningful learning of a software application and good transfer performance include:

- Using multimedia to attract and maintain the user’s attention to important information
- Applying theories of information processing to promote effective use of cognitive resources for processing and long-term storage of information, including minimizing extraneous cognitive workload
- Helping the user develop procedural knowledge and practice skills

Research literature shows that interactive tutorials can support these features of effective training. Interactive tutorials should draw and maintain a user’s attention better than passive animated or static tutorials by keeping the user actively engaged and motivated. Sustained attention should lead to higher self-reports of engagement. Interactive tutorials should promote processing of information for long-term storage and minimize cognitive workload during learning. More efficient information processing should lead to better performance on declarative knowledge and procedural knowledge tests. In addition, interactive tutorials should provide the learner the ability to control instructional and user
interface features of the tutorial, like animations, strengthening their knowledge of the material being taught. This should lead to better performance on declarative knowledge and procedural knowledge tests. Finally, interactive tutorials should provide the opportunity to rehearse software features, strengthening the user’s procedural knowledge. A stronger procedural knowledge and rehearsal of tasks should lead to better performance and faster completion of declarative knowledge and procedural knowledge tests.

Hypotheses

Three different tutorial designs arise from the reviewed literature:

- Static tutorial design: tutorials that make use of static images (e.g., a book-like format)
- Animated tutorial design: tutorials that make use of animations that demonstrate software usage, but that have minimal interactivity
- Interactive tutorial design: tutorials that are highly interactive, providing opportunity for rehearsal and the ability to control animation clips

The research literature review suggests the following hypotheses, which were tested in this study.

H1: The effect of multimedia on engagement while viewing a multimedia software tutorial.

H1a: The animated tutorial should attract and maintain the user’s attention better than a static tutorial, so participants who view the animated tutorial...
should score significantly higher on the user engagement scale than participants who learn from the static tutorial.

H1b: The interactive tutorial should attract and maintain the user’s attention better than an animated tutorial, so participants who view the interactive tutorial should score significantly higher on the user engagement scale than participants who learn from the animated or static tutorials.

H2: The effect of multimedia on declarative knowledge of a software application.

H2a: The animated tutorial should promote more effective storage of information than the static tutorial, so participants who learn from the animated tutorial should score significantly higher on the declarative knowledge test than participants who learn from the static tutorial.

H2b: The interactive tutorial should promote more effective storage of information than the animated tutorial, so participants who learn from the interactive tutorial should score significantly higher on the declarative knowledge test than participants who learn from the animated or static tutorials.

H3: The effect of multimedia on procedural knowledge of a software application.

H3a: The animated tutorial should establish better procedural knowledge than the static tutorial, so participants who learn from the animated tutorial should score significantly higher on the procedural knowledge test than participants who learn from the static tutorial.
H3b: The interactive tutorial should establish better procedural knowledge than the animated tutorial, so participants who learn from the interactive tutorial should score significantly higher on the procedural knowledge test than participants who learn from the animated or static tutorials.

H3c: The interactive tutorial should establish better procedural knowledge than the animated and static tutorials, so participants who learn from the interactive tutorial should complete the procedural knowledge test in significantly less time than participants who learn from the animated or static tutorials.
METHOD

Participants

A sample of 93 participants were used in this experiment, including students drawn from the research participant pool of undergraduate students at North Carolina State University taking an Introductory Psychology course. Students voluntarily signed up for the experiment using the Experimetrix website to partially fulfill a participation requirement for their Psychology course. The sample also included students of undergraduate courses in Cognitive Psychology and Ergonomics volunteering to participate in the experiment for extra credit in their course. Participants were randomly assigned to one of three conditions as they entered the laboratory, placing 31 students in each condition.

Participants were screened for prior knowledge or experience with Camtasia Studio™, the software program that the tutorial provides instruction on. The goal of the experiment was to determine if the independent variable (multimedia presentation style) affects learning of the software program, so if a participant was already familiar with the software program, their data would not be useful. Participants were also screened to ensure that they were fluent in English, comfortable using a computer, and had knowledge of how to use a mouse and keyboard.

Design

The experimental design of this research study included a pre-test for screening (Appendix C), three conditions of the independent variable, a User Engagement Scale (Appendix E), the NASA-TLX (Appendix F), and two tests to explore the level of learning
achieved by the participants – DV1: declarative knowledge and DV2: procedural knowledge. The declarative knowledge test (DV1) showed the participants’ level of declarative knowledge of the Camtasia software program, and the procedural knowledge test (DV2) revealed the accuracy of the participants’ procedural knowledge of the tasks they saw performed in Camtasia in the tutorial. A participant’s level of knowledge of the software program was explored in this study by examining the scores from the two tests, one revealing the level of declarative knowledge and the other revealing the level of procedural knowledge of the software program after going through a tutorial. The independent variable was the multimedia presentation type of the tutorial, so participants in one condition viewed a static tutorial (IV1), participants in the second condition viewed the animated tutorial (IV2), and participants in the third condition worked with an interactive tutorial (IV3). The dependent variables of this study were measured by performance on a test of declarative knowledge of the parts of Camtasia covered in the tutorial (DV1) and by a procedural knowledge test (DV2). After viewing a static or animated tutorial or interacting with an interactive tutorial in Camtasia, participants were given the declarative knowledge test (DV1, Appendix D) and then the procedural knowledge test (DV2). There was not a time limit for completing the tests, but the amount of time the participant took to complete the procedural knowledge test was recorded and compared across the three independent variables. In addition to the knowledge tests, a user engagement scale (Appendix E) was used to compare the participants’ self-reported level of engagement while going through the tutorial. Engagement, the average rating of the entire User Engagement Scale (DV3), was compared across the
three independent variables. As an investigation of the meta-cognitive aspects of the task of going through the tutorial, the sub-scales of the NASA-TLX (DV 4) were analyzed. Analyzing the sub-scales of the NASA-TLX compared differences in self-reported workload for participants in each of the groups.

**Materials and Procedures**

The experiment took place in a usability research laboratory on the campus of North Carolina State University. The room offered a comfortable workstation allowing a participant to view a computer monitor that displayed the tutorial and tests and have access to a keyboard and mouse to interact with them.

Upon entering the laboratory, participants received an Informed Consent Form (Appendix B) to review and sign, after which participants received one questionnaire (Appendix C) before beginning the experimental condition that they were assigned to. The questionnaire was developed by the experimenter to gain information about the participants’ demographics, level of comfort with using a computer, and level of experience with using Camtasia or video editing using a similar software program. This questionnaire ensured that all participants were fluent in English, were capable of using a computer mouse and keyboard, and assessed prior knowledge of the software program or tasks that they would be tested on. Prior knowledge or experience with Camtasia would interfere with the test results since it could not be inferred that the skill level demonstrated on the tests was from knowledge gained from the tutorial alone. Therefore, only data collected from participants who did not report prior experience with Camtasia or tasks covered in the tutorials, or
declared extensive previous experience in video editing would be included in analysis. A total of 94 participants were tested, none of which had previous experience with Camtasia Studio or with tasks and software features similar to those covered in the tutorials. One participant declared extensive previous experience using video editing software, and their data was eliminated from the sample, leaving a remaining sample of 93 participants for data analysis, 31 participants for each condition.

After viewing the tutorial for the condition that they were assigned to, participants then took a survey instrument providing a post-hoc self-reported measure of engagement while going through the tutorial, the User Engagement Scale (O’Brien & Toms, 2010), (Appendix E). The phrases used in the User Engagement Scale were modified from the original to pertain to learning from a software tutorial rather than using a website for online shopping.

After completing the User Engagement Scale, a self-reported measure of mental workload was collected by having participants complete an online version of the NASA-TLX, illustrated in Appendix F (Hart & Staveland, 1988; Sharek, 2011). The NASA-TLX composite score and subscales were collected to provide insights into how the different tutorials helped the participants to manage their cognitive resources while learning, and to determine if extraneous cognitive workload presented a problem to participants in any of the groups.

Each participant then took two learning tests. Tests were compared across conditions to see which presentation style; static, animated, or interactive; was most effective in
teaching the participants about Camtasia, and promoting greater task performance on video-editing tasks taught in the tutorial. In order for a software tutorial to be considered effective, it must help the user comprehend the layout of the software program, the function of the program’s tools, and the program’s capabilities. Knowledge of these aspects of the software program was tested in DV1, the declarative knowledge test (Appendix D). An effective tutorial will also teach the user how to perform a task in a way that will enable the user to use the software to perform the task, or a similar one, in a real-life situation without the tutorial’s assistance. The ability to perform tasks using Camtasia without the assistance of the tutorial was tested with DV2, the procedural knowledge test. The tests were developed by the experimenter to assess the level of instructional effectiveness of the tutorials. Similar to a previous study on learning software applications (Ben-Ari & Yeshno, 2006), the tests contained three different types of questions to evaluate the level of completeness of a participant’s knowledge of the software program after viewing a tutorial. The characteristics of the question types are outlined in Table 1.
Table 1

*Characterization of the assessment questions*

<table>
<thead>
<tr>
<th>Type</th>
<th>Characterization</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Questions that check for technical knowledge</td>
<td>What does the Camtasia timeline allow a user to do?</td>
</tr>
<tr>
<td>2</td>
<td>Questions that check the ability to reproduce knowledge given in the tutorial</td>
<td>Add a title slide to the timeline your video project.</td>
</tr>
<tr>
<td>3</td>
<td>Questions that check the ability to make inferences from knowledge given in the tutorial</td>
<td>Create a transition between the title slide and the recorded video.</td>
</tr>
</tbody>
</table>

The declarative knowledge test (DV1) determined the participants’ level of understanding of the program by having the participants answer questions about what the different tools in Camtasia do. The concepts described in the tutorials that were covered in the declarative knowledge test are outlined in Appendix A. The students were guided through labeled diagrams of the user interface with names to key features and tools, and asked to answer open-ended questions about what type of tasks the program would be able to accomplish and how one would accomplish a specified task using the program’s features and tools. For example, the student was asked questions about what different parts of the timeline (such as the playhead) do. For each question, the participant was asked to recall (and was sometimes shown a picture of) parts of the user interface and asked to describe the tool or functionality that was mentioned in the tutorial. If a user does not correctly understand the components of a system, it can have a negative impact on task performance (Wickens, 2000).
The procedural knowledge assessment (DV2) measured a participant’s ability to perform a task, similar to those taught in the tutorial, but on their own using the software program, demonstrating the completeness of the participant’s procedural knowledge. This test revealed a participant’s knowledge of the associations between the components of the software program and showed which causal connections the participant made between the components. If a participant gained a sufficient level of knowledge of the software program, then they would be able to select the correct components and make the proper interactions that resulted in the desired outcomes. In other words, someone who has developed sufficient knowledge of the software program from the tutorial would not use trial and error to go through the procedural knowledge test, but would draw upon their memory of the tutorial and how the user interface components and features interacted to lead to different outcomes, and then based on the knowledge they developed from observing the tutorial, anticipate which interactions they should make with the user interface to achieve similar outcomes (Potosnak, 1989).

In the procedural knowledge test, the participant was shown a video and then asked to create an identical one using Camtasia. Without any assistance, and with the ability to view the sample video as many times as they would like, they had to re-create the sample movie in a new file within Camtasia Studio. Their performance was timed, observations of their actions were collected, and their final product was saved. The time that was needed to complete the procedural knowledge test was compared across groups. The final product was compared to the sample video to check for accuracy. Accuracy was measured based on
whether or not participants accomplished 25 key tasks (for a maximum score of 25) to create a video that matched the sample video (Appendix I). Demonstrating familiarity with the program’s layout and features, and lack of errors in the final creation represented a high level of procedural knowledge. The tasks involved in the procedural knowledge test and the concepts that they map to are outlined, and several are illustrated for further clarification, in Appendix A. The tasks involved in creating the video included creating a screen capture movie of a Google search, creating a title slide and transition between the title slide and the screen capture, adding zoom effects, and then saving and publishing the complete project as a video.

The experimenter created three tutorials, one for each condition. Static (IV1), animated (IV2), and interactive (IV3) versions of the tutorial were made, ensuring that all information conveyed in the tutorial was identical across conditions so that only multimedia presentation style was different across the tutorials. Audio narration files, screenshots of Camtasia software, and video clips demonstrating use of Camtasia software were taken from the standard Camtasia tutorials provided on the Support website of TechSmith (TechSmith Corporation, 2012) in order to utilize multimedia materials that are considered to be standard for the online software training industry. The author developed the three versions of the tutorials, which presented the instructional content differently in each tutorial to display the content in different multimedia presentation styles - static (IV1), animated (IV2), and interactive (IV3). The interactive version (IV3) provided instruction while allowing the participants to practice the skills being taught in the software program. The instructional
tutorial played within Camtasia Studio (as shown in Figure 1), allowing for instruction to proceed at the participant’s pace, and enabling the participant to practice using Camtasia’s user interface while going through the tutorial.

*Figure 1.* The interactive tutorial allowed the user to view the tutorial within the Camtasia software program so that they could see the program’s user interface and interact with it.

In the interactive tutorial (IV3), the participants were able to pause the tutorial, rewind, and replay it at any time in order to repeat a section that they felt the need to review. Throughout
the interactive tutorial, the participant was prompted to pause the tutorial video after a main concept was taught and encouraged to locate components in the user interface that were being covered in the tutorial (as shown in Figure 2), and then practice using the software to create and edit video projects as instructed by the tutorial. The interactive tutorial was the only tutorial that gave participants an opportunity to practice using the software.

Figure 2. In the interactive tutorial, the participant was challenged to locate components mentioned in the tutorial in the software’s user interface.
The animated tutorial (IV2) had animated demonstrations of how the display looks when it is being manipulated to accomplish the tasks described. The participants were able to pause the animated tutorial, rewind, and replay it at any time in order to repeat a section that they felt the need to review. The static version (IV1) demonstrated the same steps, but using static pictures in a presentation similar to a PowerPoint slide show, with controls to allow the participant to progress from one slide to the next when they finished viewing the information on a slide. Audio narration was used in the tutorials and was identical across the three conditions. The narration helped the students hear the information while looking at text, pictures, and videos in the tutorial, applying multimedia instructional design principles supported by previous research to decrease extraneous cognitive workload. Participants used as much time as they needed to complete each tutorial. Each tutorial included clear instructions on how to go through the tutorial and motivated and encouraged participants to engage with the tutorial. The differences that characterized the three different tutorials are outlined in Table 2.
Table 2

**Differences between the multimedia tutorials**

<table>
<thead>
<tr>
<th>Static Tutorial (IV1)</th>
<th>Animated Tutorial (IV2)</th>
<th>Interactive Tutorial (IV3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Presented instruction through static images and audio narration</td>
<td>Presented instruction through animated video demonstrating the software accompanied with audio narration</td>
<td>Presented instruction through animated video demonstrating the software accompanied with audio narration</td>
</tr>
<tr>
<td>Allowed users to control the pace of instruction by clicking a button to advance to the next image.</td>
<td>Provided controls to allow the user to play, pause, and rewind the instructional video</td>
<td>Provided controls to allow the user to play, pause, and rewind the instructional video</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instructed the user to pause after a main concept was covered in order to locate a tool and interact with it</td>
</tr>
<tr>
<td>Allowed the user to replay the tutorial, or portions of the tutorial as many times as needed</td>
<td>Allowed the user to replay the tutorial, or portions of the tutorial as many times as needed</td>
<td>Allowed the user to replay the tutorial, or portions of the tutorial as many times as needed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Allowed the user to try the practice exercises as many times as needed</td>
</tr>
<tr>
<td>Did not give the user access to the software program while using the tutorial</td>
<td>Did not give the user access to the software program while using the tutorial</td>
<td>Interacted within the software so that the user could familiarize themselves with the software and interact with it while they were learning about it</td>
</tr>
</tbody>
</table>

**Data Collection**

As participants entered the lab, they were randomly assigned to one of the three conditions. The participants were asked to read and sign a consent form, and then complete a
questionnaire (Appendix C). The participants were then asked to sit in front of the computer. They were then instructed on how to view and navigate through the tutorial, adjust the speakers to hear the narration, and interact with the tutorial. All participants were encouraged to spend as much time as they needed in order to sufficiently study the tutorial. The total amount of time that participants spent going through the tutorial was recorded. After the participants studied the tutorial, they completed the User Engagement Scale (O’Brien & Toms, 2010), the online version of the NASA-TLX (Hart & Staveland, 1988; Sharek, 2009), and then they completed two knowledge tests, one on declarative knowledge and one on procedural knowledge. The participants were allowed to spend as much time as they needed to complete the tests to the best of their ability. The declarative knowledge test was saved on the computer after it was completed. The procedural knowledge test required the student to watch a video on the computer, and then use Camtasia to complete a series of tasks to create a video that was as identical as possible to the video that they saw. The test administrator collected observations and notes of the participants’ actions, recorded the time needed for the participants to complete the procedural knowledge test, and ensured that the student’s work was saved for later comparison with the sample video.

Data Analysis

The tests were scored and analyzed to compare the instructional effectiveness of each tutorial. The declarative knowledge test was scored on accuracy, with students receiving one point for each possible correct answer to the test questions. The participants received zero points for unanswered questions or incorrect answers. The higher the number of points
received on the test, the higher the level of comprehension of the software program the participant gained from the tutorial. The procedural knowledge test was scored based on accuracy of the video the participant created compared to the sample video. Accuracy was measured by how well the participant created a movie that resembled the one that they were shown and asked to replicate with Camtasia. Participants received points for each task accomplished successfully in the procedure to create a video that replicated the sample video. There were 25 key tasks that were checked for, allowing the participants to receive a maximum of 25 points, one point for each of the tasks completed to create their video.
RESULTS

ANOVAs were used to compare each dependent variable across the conditions. If the results obtained from the ANOVAs supported the hypotheses, then a main effect would be observed for the dependent variables. If main effects were found, a post-hoc test, Tukey’s HSD, was used to see which conditions showed significant differences.

It should be inferred from any significant results that there is a difference in tutorial effectiveness based on multimedia presentation style since the study was designed to control that only multimedia presentation style would have an effect on the measures of instructional effectiveness. The software program, Camtasia, was selected because it is unfamiliar to a majority of the population from which the sample was drawn, but is interesting to them. It also contains a complex user interface, with many components and interactions for a user to learn. All tutorials were designed to convey the same instructional content, just with a different style of multimedia presentation, static, animated, or interactive. Also, the tutorials were designed based on empirical research on human-computer interaction, multimedia learning theory, and usable instructional multimedia design so that confusion over an unclear or unusable tutorial design was avoided and extraneous cognitive load was decreased as much as possible to reduce affecting the performance of the participants. Therefore, the results obtained from the data analysis should allow the experimenter to draw valid inferences about the effectiveness of different multimedia presentation styles in the design of software tutorials, allowing software tutorial designers to know which is the most effective way to utilize multimedia to instruct software users.
Did the type of multimedia used in the software tutorial affect user engagement? It was hypothesized that the type of multimedia used in the software tutorial would affect the level of engagement of the participant learning from the tutorial.

H1: The effect of multimedia on engagement while viewing a multimedia software tutorial.

H1a: The animated tutorial should attract and maintain the user’s attention better than a static tutorial, so participants who view the animated tutorial should score significantly higher on the user engagement scale than participants who learn from the static tutorial.

H1b: The interactive tutorial should attract and maintain the user’s attention better than an animated tutorial, so participants who view the interactive tutorial should score significantly higher on the user engagement scale than participants who learn from the animated or static tutorials.

An ANOVA performed in JMP, with user engagement as the dependent variable, indicated a significant main effect for multimedia tutorial type, $F(2,90) = 3.63, p < 0.05$. The possible scores for the User Engagement Scale ranged between 0 and 5. As displayed in Table 3 and Figure 3, participants who viewed the static tutorial had the lowest scores for user engagement, while participants who viewed the animated tutorial had a higher level of user engagement, and participants who viewed the interactive tutorial had the highest level of user engagement. A Tukey HSD post-hoc analysis showed that user engagement was significantly higher for participants who learned how to use the software program from the interactive tutorial ($M = 3.76, SD = 0.39$), over the static tutorial ($M = 3.42, SD = 0.60$), $p < .05$, but not the animated tutorial ($M = 3.63, SD = 0.51$). While user engagement was higher
for those who viewed the animated tutorial over those who viewed the static tutorial, there
was not a significant difference.

Table 3

User Engagement Scale Performance

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>N</th>
<th>M</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>3.42</td>
<td>0.60</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>3.63</td>
<td>0.51</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>3.76</td>
<td>0.39</td>
</tr>
</tbody>
</table>

Figure 3. User Engagement Scale scores for the three groups. This graph shows the means
and standard error for User Engagement Scale scores of each group of participants.
Did the type of multimedia used in the software tutorial affect performance on tests of declarative knowledge of the software program? It was hypothesized that the type of multimedia used in the software tutorial would affect the level of declarative knowledge gained from the tutorial on the software program.

H2: The effect of multimedia on declarative knowledge of a software application.

H2a: The animated tutorial should promote more effective storage of information than the static tutorial, so participants who learn from the animated tutorial should score significantly higher on the declarative knowledge test than participants who learn from the static tutorial.

H2b: The interactive tutorial should promote more effective storage of information than the animated tutorial, so participants who learn from the interactive tutorial should score significantly higher on the declarative knowledge test than participants who learn from the animated or static tutorials.

An ANOVA performed in JMP, with declarative knowledge test scores as the dependent variable, indicated a significant main effect for multimedia type, $F(2,90) = 9.84, p < 0.01$. As displayed in Table 4 and Figure 4, participants who viewed the static tutorial had performance in declarative knowledge that was slightly higher than participants who viewed the animated tutorial. Participants who viewed the animated tutorial had the lowest scores for declarative knowledge. Participants who viewed the interactive tutorial had the highest performance in declarative knowledge. A Tukey HSD post-hoc analysis showed that declarative knowledge was significantly higher for participants who learned how to use the
software program from the interactive tutorial ($M = 29.84$, $SD = 6.70$), over the animated tutorial ($M = 22.26$, $SD = 7.07$), $p < .01$ and static tutorial ($M = 23.26$, $SD = 8.09$), $p < .01$, supporting hypothesis H2b. There was not a significant difference between declarative knowledge for participants who viewed the static and animated tutorials.

Table 4

**Declarative Knowledge Test Performance**

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>$N$</th>
<th>$M$</th>
<th>$SD$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>23.26</td>
<td>8.09</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>22.26</td>
<td>7.07</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>29.84</td>
<td>6.70</td>
</tr>
</tbody>
</table>

*Figure 4.* Declarative knowledge test scores for the three groups. This graph shows the means and standard error for declarative knowledge test scores of each group of participants.
Did the type of multimedia used in the software tutorial affect performance on tests of procedural knowledge of the software program? It was hypothesized that the type of multimedia used in the software tutorial would affect the level of procedural knowledge gained from the tutorial on the software program.

H3: The effect of multimedia on procedural knowledge of a software application.

H3a: The animated tutorial should establish better procedural knowledge than the static tutorial, so participants who learn from the animated tutorial should score significantly higher on the procedural knowledge test than participants who learn from the static tutorial.

H3b: The interactive tutorial should establish better procedural knowledge than the animated tutorial, so participants who learn from the interactive tutorial should score significantly higher on the procedural knowledge test than participants who learn from the animated or static tutorials.

H3c: The interactive tutorial should establish better procedural knowledge than the animated and static tutorials, so participants who learn from the interactive tutorial should complete the procedural knowledge test in significantly less time than participants who learn from the animated or static tutorials.

An ANOVA performed in JMP, with procedural knowledge test scores as the dependent variable, indicated a significant main effect for multimedia type, \( F(2,90) = 3.17, \ p < 0.05 \). As shown in Table 5 and Figure 5, participants who viewed the static tutorial had slightly higher performance in procedural knowledge than participants who viewed the animated tutorial. Participants who viewed the animated tutorial had the lowest scores for
procedural knowledge. Participants who viewed the interactive tutorial had the highest performance in procedural knowledge. A Tukey HSD post-hoc analysis showed that procedural knowledge was significantly higher for participants who learned how to use the software program from the interactive tutorial ($M = 23.48$, $SD = 1.98$), over the animated tutorial ($M = 21.19$, $SD = 4.05$), $p < .05$, but not the static tutorial ($M = 21.77$, $SD = 4.60$). There was not a significant difference between procedural knowledge for participants who viewed the static and animated tutorials.

Table 5

*Procedural Knowledge Test Performance*

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>$N$</th>
<th>$M$</th>
<th>$SD$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>21.77</td>
<td>4.60</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>21.19</td>
<td>4.05</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>23.48</td>
<td>1.98</td>
</tr>
</tbody>
</table>
Figure 5. Procedural knowledge test scores for the three groups. This graph shows the means and standard error for procedural knowledge test scores of each group of participants.

An ANOVA performed in JMP, with the amount of time it took participants to complete the procedural knowledge test as the dependent variable, did not indicate a significant main effect for multimedia type, $F(2,90) = 2.50, p < 0.10$. Participants who viewed the static tutorial took the most amount of time (measured in seconds) to complete the procedural knowledge test ($M = 2240.29$), while participants who viewed the animated tutorial had faster performance ($M = 1958.65$), and participants who viewed the interactive tutorial had the fastest performance ($M = 1760.87$).

Did the type of multimedia used in the software tutorial affect workload? An ANOVA performed in JMP, with the NASA-TLX composite unweighted score as the dependent variable, indicated a significant main effect for multimedia type, $F(2,90) = 3.16, p < 0.05$. Possible scores for the NASA-TLX ranged from 0 to 100. As displayed in Table 6
and Figure 6, participants who viewed the static tutorial had lower NASA-TLX scores than participants who viewed the animated tutorial. Participants who viewed the animated tutorial had the highest NASA-TLX scores. Participants who viewed the interactive tutorial had the lowest NASA-TLX score. A Tukey HSD post-hoc analysis showed that workload was significantly lower for participants who learned how to use the software program from the interactive tutorial \((M = 27.67, SD = 13.52)\), over the animated tutorial \((M = 37.40, SD = 15.89)\), \(p < .05\), but not the static tutorial \((M = 33.92, SD = 16.78)\). There was not a significant difference between NASA-TLX scores for participants who viewed the static and animated tutorials.

Table 6

*NASA-TLX composite scores*

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>(N)</th>
<th>(M)</th>
<th>(SD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>33.92</td>
<td>16.78</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>37.40</td>
<td>15.89</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>27.67</td>
<td>13.52</td>
</tr>
</tbody>
</table>
Figure 6. NASA-TLX composite scores for the three groups. This graph shows the means and standard error for workload of each group of participants.

Did the type of multimedia used in the software tutorial affect the participants’ perception of how much they learned with the tutorial? An ANOVA performed in JMP, with the NASA-TLX unweighted performance sub-score as the dependent variable, indicated a significant main effect for multimedia type, $F(2,90) = 5.54$, $p < 0.05$. Possible values for the NASA-TLX sub-score ranged from 0 to 1. As displayed in Table 7 and Graph 7, participants who viewed the static tutorial had higher NASA-TLX unweighted performance scores than the participants who studied from the animated tutorial. Participants who viewed the animated tutorial had the lowest NASA-TLX performance scores. Participants who viewed the interactive tutorial had the highest NASA-TLX performance scores. A Tukey HSD post-hoc analysis showed that the NASA-TLX unweighted performance score was significantly higher for participants who learned how to use the software program from the interactive
tutorial \((M = 0.25, SD = 0.07)\), over the animated tutorial \((M = 0.19, SD = 0.02)\), \(p < .05\), and
the static tutorial \((M = 0.19, SD = 0.09)\). There was not a significant difference between
NASA-TLX performance scores for participants who viewed the static and animated
tutorials.

Table 7

**NASA-TLX unweighted performance scores**

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>N</th>
<th>M</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.19</td>
<td>0.09</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.19</td>
<td>0.02</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.25</td>
<td>0.07</td>
</tr>
</tbody>
</table>

*Figure 7.* NASA-TLX unweighted performance scores for the three groups. This graph shows the means and standard error for NASA-TLX unweighted performance scores of each group of participants.
Did the type of multimedia used in the software tutorial affect the participants’ perception of how much effort they put into learning from the tutorial? An ANOVA performed in JMP, with the NASA-TLX unweighted effort sub-score as the dependent variable, indicated a significant main effect for multimedia type, $F(2,90) = 5.97, p < 0.05$. Possible values for the NASA-TLX sub-score ranged from 0 to 1. As displayed in Table 8 and Figure 8, participants who viewed the static tutorial had the highest NASA-TLX unweighted effort scores, while participants who studied from the animated tutorial had slightly lower NASA-TLX unweighted effort scores. Participants who viewed the interactive tutorial had the lowest NASA-TLX unweighted effort score. A Tukey HSD post-hoc analysis showed that the NASA-TLX unweighted effort score was significantly lower for participants who learned how to use the software program from the interactive tutorial ($M = 0.18$, $SD = 0.06$), over both the animated tutorial ($M = 0.23$, $SD = 0.07$), $p < .05$, and the static tutorial ($M = 0.23$, $SD = 0.07$). There was not a significant difference between NASA-TLX effort scores for participants who viewed the static and animated tutorials.

Table 8

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>$N$</th>
<th>$M$</th>
<th>$SD$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.23</td>
<td>0.07</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.23</td>
<td>0.07</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.18</td>
<td>0.06</td>
</tr>
</tbody>
</table>
Figure 8. NASA-TLX unweighted effort scores for the three groups. This graph shows the means and standard error for NASA-TLX unweighted effort scores of each group of participants.

Did the type of multimedia used in the software tutorial affect the participants’ perception of how rushed they felt? An ANOVA performed in JMP, with the NASA-TLX unweighted temporal sub-score as the dependent variable, indicated a significant main effect for multimedia type, $F(2,90) = 4.33, p < 0.05$. Possible values for the NASA-TLX sub-score ranged from 0 to 1. As displayed in the following graph and tables, participants who viewed the animated tutorial had the highest NASA-TLX unweighted temporal score, while participants who viewed the interactive tutorial had lower NASA-TLX temporal scores, and participants who viewed the static tutorial had the lowest NASA-TLX temporal scores. A Tukey HSD post-hoc analysis showed that the NASA-TLX unweighted temporal score was significantly higher for participants who learned how to use the software program from the
animated tutorial ($M = 0.18, SD = 0.10$), over the static tutorial ($M = 0.12, SD = 0.08$), $p < .05$.

Table 9

**NASA-TLX unweighted temporal scores**

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>$N$</th>
<th>$M$</th>
<th>$SD$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.12</td>
<td>0.08</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.18</td>
<td>0.10</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.16</td>
<td>0.09</td>
</tr>
</tbody>
</table>

*Figure 9.* NASA-TLX unweighted temporal scores for the three groups. This graph shows the means and standard error for NASA-TLX unweighted temporal scores of each group of participants.
Performance for the other NASA-TLX unweighted sub-scores did not indicate statistically significant results across the three groups. Possible values for the NASA-TLX sub-score ranged from 0 to 1. Table 10 shows the results across the groups for the NASA-TLX mental workload sub-score.

Table 10

*NASA-TLX unweighted mental workload sub-scores*

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>N</th>
<th>M</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.29</td>
<td>0.07</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.28</td>
<td>0.06</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.28</td>
<td>0.06</td>
</tr>
</tbody>
</table>

Table 11 shows the results across the groups for the NASA-TLX physical workload sub-score. Possible values for the NASA-TLX sub-score ranged from 0 to 1.
Table 11

NASA-TLX unweighted physical workload sub-scores

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>N</th>
<th>M</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.04</td>
<td>0.07</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.02</td>
<td>0.03</td>
</tr>
</tbody>
</table>

Table 12 shows the results across the groups for the NASA-TLX frustration workload sub-score. Possible values for the NASA-TLX sub-score ranged from 0 to 1.

Table 12

NASA-TLX unweighted frustration workload sub-scores

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>N</th>
<th>M</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>0.08</td>
<td>0.09</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>0.06</td>
<td>0.08</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>0.09</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Did the type of multimedia used in the software tutorial affect the amount of time that the participants spent studying the tutorial? An ANOVA performed in JMP, with the amount of time it took participants to complete the tutorial as the dependent variable, did indicate a
significant main effect for multimedia type, $F(2,90) = 45.59, p < 0.01$. A Tukey HSD post-hoc analysis showed that participants spent a significantly longer period of time studying from the interactive tutorial ($M = 1839.35, SD = 426.02$) than the animated tutorial ($M = 1110.32, SD = 285.64$), $p < .01$ or the static tutorial ($M = 1316.81, SD = 158.03$), $p < .01$.

Table 13

*Time in seconds that participants spent studying from the tutorials*

<table>
<thead>
<tr>
<th>Tutorial Type</th>
<th>$N$</th>
<th>$M$</th>
<th>$SD$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>31</td>
<td>1316.81</td>
<td>158.03</td>
</tr>
<tr>
<td>Animated</td>
<td>31</td>
<td>1110.32</td>
<td>285.64</td>
</tr>
<tr>
<td>Interactive</td>
<td>31</td>
<td>1839.35</td>
<td>426.02</td>
</tr>
</tbody>
</table>

*Figure 10.* Time in seconds that participants spent studying from the tutorial. This graph shows the means and standard error for times spent on the tutorial for each group of participants.
DISCUSSION

This study was designed to investigate if interactivity with a multimedia software tutorial would increase a learner’s level of engagement with the tutorial and improve the level of declarative knowledge and procedural knowledge that a learner would have of the software program that was taught in the tutorial. Participants who studied from an interactive tutorial had the highest level of user engagement over those who studied from the animated or static tutorials, significantly higher than those who studied from the static tutorial. This result seems to indicate that the interactive tutorial allowed users to maintain a higher level of engagement with the tutorial than those who learned from a tutorial that showed the same material in a slide show format that contained static slides. While the animated tutorial seemed to engage users more than the static tutorial, the interactivity of the third condition also contributed to higher engagement. Many distance learning and corporate training e-learning courses utilize a format that incorporates a slide show of images with audio. If the goal is to maintain the learner’s interest in the subject matter being taught, then this study shows that incorporating both animation and interactivity into the tutorial is worthwhile.

The results of this study showed that those who studied from an interactive software tutorial performed significantly better on a test of declarative knowledge of the software program over those who studied from the animated or static tutorials. The interactive multimedia tutorial allowed the learner to practice (rehearse) using the software to accomplish tasks that they were learning about. Just as importantly, the interactive design allowed the user to see how the user interface is arranged and the location and purpose of the
tools in the software used to accomplish tasks. This allowed users to mentally encode both the name of the tool and also its spatial location, as well as see the tool in action, strengthening a mental model of these declarative elements. There was not a significant difference of declarative knowledge between those who studied from the animated tutorial and those who studied from the static tutorial, seeming to indicate that simply viewing the location and function of the tool (in either static or dynamic form) without rehearsal did not accomplish the same level of mental encoding for the type of declarative knowledge tested here.

Participants who studied from an interactive tutorial demonstrated the highest level of procedural knowledge over those who studied from the animated or static tutorials, significantly higher than those who studied from the animated tutorial. The results seem to indicate that animation alone did not play a key role in improving a learner’s ability to perform tasks using the software after studying the tutorial. However, the results do suggest that providing a self-paced learning environment contributed the most to teaching learners procedural knowledge that allowed them to perform tasks accurately on their own using the software program that they learned about. From an efficiency standpoint, the study also found that participants who studied from the interactive tutorial performed the procedural knowledge test faster than those who studied from the animated tutorial, and those who studied from the static tutorial took the most amount of time to complete the procedural knowledge test. Thus, in order to prepare learners to perform tasks using the software
program that they learned about both accurately and quickly, an interactive multimedia tutorial seems to be the best form of instruction of the three.

NASA-TLX workload scores, both the composite and individual subscale scores, provide insight as to why the differences in learning seen between the three conditions may have occurred. Those who studied from an interactive multimedia software tutorial reported significantly lower overall workload on the NASA-TLX test compared to those who studied from the animated tutorial. Extraneous cognitive workload should be reduced as much as possible (Paas et al., 2003b) and it may be that animation, without the ability to self-pace as one can in both the static and interactive conditions, presents the subject matter at a pace that is too fast to comfortably comprehend (Tversky, Morrison, & Betancourt, 2002). This interpretation is also supported by NASA-TLX sub-scores, which also showed that animated tutorial learners had the highest temporal workload scores, significantly higher than those who studied from the static tutorial. While there was not a significant difference between the animated and interactive conditions on the temporal sub-scale, users reported putting in significantly more effort on both the static and animated conditions when compared to the interactive condition. This seems to indicate that animations—as used in both the animated and interactive conditions—put some temporal pressure on the learner, but that the interactive condition’s design helped lower the overall effort put into using it. The results of this study suggest that providing the user with more user control over the pace that the material is presented is helpful for reducing cognitive workload.
Appropriately, the NASA-TLX sub-scores also revealed that users had a positive perception of learning from the interactive multimedia software tutorial. Those who studied from the interactive tutorial had significantly higher scores for the performance workload sub-scale than those who studied from the static or animated tutorial. While interactivity significantly decreased the learners’ perception of how much effort they had to expend in order to learn the material that was taught in the tutorial, the subscale for mental workload did not show any differences between the three conditions. This seeming contradiction would need to be studied further.

In conclusion, this study showed that interactivity in multimedia software tutorials is effective in improving engagement with the tutorial, the user’s perception of the learning experience, declarative knowledge of the software program being taught, and ability to use the software program after the tutorial to perform procedural tasks quickly and accurately. The study supports the strategy of incorporating interactivity when teaching software users with multimedia tutorials so that users can control the pace of the material being taught and practice using the software right after they learn about a feature or task. Interactive tutorials can be beneficial to those who need to learn how to use a software program through e-learning, including those who are new to a software program that they recently purchased, university students learning how to use a software program through distance learning, and employees who need to quickly learn about a software program that they will be required to utilize on the job. Further research can apply methods from this study to investigate the effects of interactivity for teaching other subjects through e-learning.
As part of planned future research, a deeper investigation will be conducted to determine the factors behind the differences in time that participants spent completing the three multimedia tutorials. For example, observations that were collected during the study will be examined to see which factors led to students pausing and re-playing sections of the tutorial, and how reported level of engagement relates to the amount of time that a participant spent on a tutorial. Similarly, the amount of time that participants spent on the tutorial will be compared to their performance on declarative and procedural test scores. Many of the possible relationships between tutorial type, training time, and self-reported engagement will be explored using hierarchical regression. This analysis will take advantage of the individual User Engagement Sub-scales. This may reveal individual attributes of engagement that contributed to the learners’ experience with the tutorial. The results from the User Engagement Scale can identify levels of six factors that contributed to engagement – perceived usability, aesthetics, novelty, felt involvement, and focused attention (O’Brien & Toms, 2010). Exploring the scores that contributed to these factors can further identify attributes that can be enhanced in multimedia software tutorials to improve usability and engagement. Finally, part of this further investigation of test scores will be to reconfirm the reliability of the scores with double-blind inter-rater analysis. In addition, the open-ended responses from the final questionnaire will be examined to reveal insights into factors that can contribute to multimedia tutorial usability.
REFERENCES


Mayer, R. E. (2003c). The promise of multimedia learning: Using the same instructional design methods across different media. Learning and Instruction, 13, 125-139.


Appendix A - *Main concepts and associated tasks covered in the tutorials*

**Recording a video with Camtasia**

Task 1. Locate and open the recording controls in Camtasia.

Task 2. Choose the area of screen to record and start recording.

Task 3. Record the actions of the screen and stop the recording when a pre-assigned task of conducting a Google maps search has been completed.
Task 4. Preview the video of the screen capture.

Task 5. Save the video as a camrec file and open it in Camtasia.
Task 6. Locate the video in the clip bin.

Task 7. Add the video to the project timeline.

Getting acquainted with the Camtasia user interface components

Task 1. Learn the functions of the timeline features
Task 2. Learn about the layers of the timeline.

Task 3. Learn about the playhead.

Task 4. Learn about the Clip Bin.
Task 5. Learn how to import objects to the Clip Bin.

Task 6. Adding objects from the Clip Bin to the timeline.
Task 7. Learn about the library.

Task 8. Learn about where to find special effects that can be added to a video project.
Task 9. Learn how to produce a video project.

Task 10. Learn that additional features and settings can be accessed from the application menu bar.

Task 11. Learn about the Preview menu and its control buttons.

Zooming effects

Task 1. Learn about Smart Focus keyframes.
Task 2. Previewing a Smart Focus zooming event.

Task 3. Moving a zoom keyframe in the Zoom layer of the timeline.

Task 4. Editing a zooming effect in the Zoom-n-Pan tab.

Task 5. Manually add a zoom effect keyframe to the timeline.

Task 6. Delete a keyframe from the timeline by right-clicking on it and selecting Remove from timeline from the context menu.

Editing clips in the timeline

Task 1. Select an area of in the timeline and click the cut icon to remove it.
Task 2. Learn how all linked tracks are affected by edits and how locked tracks are not.

Task 3. Split a clip into multiple clips, add media between clips or right-click on a section to delete.

Adding editable media, like a title clip

Task 1. Learn that new media is added at the playhead on the timeline.

Task 2. Click Add Title Clip in the Title Clips tab.
Task 3. Add and edit text and images in the title clip.

Task 4. Learn to adjust the duration of a clip.

Task 5. Add the title clip from the clip bin to the timeline.

Adding a transition
  Task 1. View available transitions to add to the video project.
Task 2. Work in the Storyboard view to see where transitions can be added and preview them.

Task 3. Drag a transition to add it between clips or replace another transition.

Task 4. Adjust the duration or select other transitions form the right-click menu.

Task 5. Preview the video with the transition.
Appendix B – Informed Consent Form

North Carolina State University
INFORMED CONSENT FORM for RESEARCH

Title of Study: Interactive Multimedia Study

Principal Investigator: Lisa Whitman
Faculty Sponsor: Dr. Eric Wiebe

What are some general things you should know about research studies?
You are being asked to take part in a research study. Your participation in this study is voluntary. You have the right to be a part of this study, to choose not to participate or to stop participating at any time without penalty. The purpose of research studies is to gain a better understanding of a certain topic or issue. You are not guaranteed any personal benefits from being in a study. Research studies also may pose risks to those that participate. In this consent form you will find specific details about the research in which you are being asked to participate. If you do not understand something in this form it is your right to ask the researcher for clarification or more information. A copy of this consent form will be provided to you. If at any time you have questions about your participation, do not hesitate to contact the researcher(s) named above.

What is the purpose of this study?
The Interactive Multimedia Study is designed to gather information on the impact that different presentation styles of multimedia software tutorials have on learning. Many people are utilizing computers to obtain training on how to use software applications. However, more research is needed on how computers can be used to optimize learning of software applications and how to use them. Through this research, we hope to identify the most effective ways to deliver electronic software training, and provide guidelines for designers and developers of electronic software tutorials.

What will happen if you take part in the study?
If you agree to participate in this study, you will be asked to view a tutorial about video production software, Camtasia Studio. You will have as much time as you need to view the instructional sessions, and may repeat the information given in the session as many times as needed to study the material as well as possible. Testing on knowledge retention and comprehension of the material taught in the tutorial will follow the tutorial. A questionnaire will also be given to you after watching the tutorial.

Risks
The risks from participating in this study, though unlikely, may include discomfort while using the computer or stress. If at any time you feel stressed or uncomfortable while using the computer, please notify the researcher and your workspace will be adjusted to accommodate your needs.

Benefits
After participating in this study, you will be benefited by increased knowledge of the video production software, Camtasia Studio, and the process of creating videos utilizing screen captures. The research conducted in this study will add to research literature on the use of interaction in multimedia learning environments. This information will be of assistance to teachers, students, instructional designers, and psychologists by providing knowledge about how students learn from computers and how multimedia learning materials can be designed for optimal effectiveness.

Confidentiality
The information in the study records will be kept confidential to the full extent allowed by law. Data will be collected anonymously and stored securely. Data collected cannot be traced to individual participants. No
reference will be made in oral or written reports which could link you to the study. You will NOT be asked to write your name on any study materials so that no one can match your identity to the answers that you provide.

**Compensation**
For participating in this study you will receive 3 participation credits towards the PSY 200 research requirement, or extra credit for the Psychology class you are enrolled in. If you withdraw from the study prior to its completion, you will not receive credit.

**What if you are a NCSU student?**
Participation in this study is not a course requirement and your participation or lack thereof, will not affect your class standing or grades at NC State.

**What if you have questions about this study?**
If you have questions at any time about the study or the procedures, you may contact the researcher, Lisa Whitman, at Lisa_Whitman@ncsu.edu.

**What if you have questions about your rights as a research participant?**
If you feel you have not been treated according to the descriptions in this form, or your rights as a participant in research have been violated during the course of this project, you may contact Deb Paxton, Regulatory Compliance Administrator, Box 7514, NCSU Campus (919/515-4514).

**Consent To Participate**
“I have read and understand the above information. I have received a copy of this form. I agree to participate in this study with the understanding that I may choose not to participate or to stop participating at any time without penalty or loss of benefits to which I am otherwise entitled.”

Subject's signature_______________________________________ Date _________________

Investigator's signature__________________________________ Date _______________
Appendix C – Initial Experiment Participant Questionnaire

Initial Experiment Participant Questionnaire

Before participating in the experiment, please answer the following questions:

1. Gender (Circle One) M F
2. Primary Language: ___________________
3. Student Status (Circle One): Freshman Sophomore Junior Senior Grad N/A
4. Academic Major: ________________________ Class Receiving Credit For: __________
5. Are you currently employed? Y N Position: ________________________________
6. Do you know how to do a screen capture on a computer? Y N
7. Do you know how to create a screen capture video? Y N
8. Have you ever worked with video editing software? Y N
9. How extensive is your knowledge about video editing software? (Circle One)

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I don’t know anything about video editing</td>
<td>I know how to edit videos</td>
<td>I have basic skills in editing videos</td>
<td>I have skills in using video editing software</td>
<td>I have an extensive working knowledge of video editing</td>
</tr>
</tbody>
</table>

10. Have you done any of the following using video editing software? (Check all that apply.)

☐ Record a screen capture ☐ Cut part of a video clip ☐ Create video effects

12. Have you previously used video editing software? (Circle One) Y N

If Yes, please describe which software you’ve used and what tasks you performed:

____________________________________________________________________________________

____________________________________________________________________________________
13. Do you know how to use a computer and mouse? (Circle One)  Y    N

14. Please rate how comfortable you are using computers. (Circle One)

1   2   3   4   5
I don’t know how to use a computer or mouse  I have very little experience using a computer  I am a novice computer user  I am comfortable using a computer  I have extensive experience using computers
Appendix D – *Declarative Knowledge Test*

**Testing Your Knowledge of Camtasia Studio Software**

Please answer the following questions to the best of your ability and as thoroughly as possible, recalling as much information as you can from the tutorial. The more information that you can remember from the tutorial, the higher your score will be. Notify the experiment facilitator when you have completed the test.

1. What happens when you click the button that reads, “Record the screen?” (Shown in the picture below.)

2. What is Camtasia capable of recording?
3. What is the timeline (shown in the picture below) used for?

4. In the timeline toolbar (shown below), what do the zoom controls do?

5. What does the Playhead do?
6. What is the Clip Bin used for?

7. How would you add media to the timeline?

8. What is the Preview Window used for?

9. What does Camtasia SmartFocus do?

10. What do the blue diamonds on the Zoom track of the Timeline do?

11. What are the red and green squares around the Playhead for?

12. What are the benefits of splitting a clip?

13. What do Transitions do?

14. What is the Storyboard used for?

Questionnaire
(The User Engagement Scale was delivered online after the participant took the declarative knowledge test and procedural knowledge test.)
1. I lost myself in this learning experience.
2. I was so involved in my watching the tutorial that I lost track of time.
3. I blocked out things around me when I was watching the tutorial.
4. When I was watching the tutorial, I lost track of the world around me.
5. The time I spent watching the tutorial just slipped away.
6. I was absorbed in watching the tutorial.
7. During this learning experience I let myself go.
8. I was really drawn into watching the tutorial.
9. I felt involved in watching the tutorial.
10. This learning experience was fun.
11. I continued to watch the tutorial out of curiosity.
12. The content of the tutorial incited my curiosity.
13. I felt interested in watching the tutorial.
14. Watching the tutorial was worthwhile.
15. I consider my learning experience a success.
16. Watching the tutorial did not work out the way I had planned.*
17. My learning experience was rewarding.
18. I would recommend watching the tutorial to my friends and family.
19. This tutorial is attractive.
20. This tutorial was aesthetically appealing.
21. I liked the graphics and images used in this tutorial.
22. This tutorial appealed to my visual senses.
23. The screen layout of this tutorial was visually pleasing.
24. I felt frustrated while watching the tutorial.*
25. I found this tutorial confusing to use.*
26. I felt annoyed while watching the tutorial.*
27. I felt discouraged while watching the tutorial.*
28. Using this tutorial was mentally taxing.*
29. This learning experience was demanding.*
30. I felt in control of my learning experience.
31. I could not do some of the things I needed to do in this tutorial.*

The questions were listed in a randomized order. The scale was administered using a five-point scale with “strongly disagree” and “strongly agree” at the respective endpoints. Items identified with an asterisk (*) indicate items that were reverse-coded.
Appendix F – *Online NASA-TLX used in the research study*

The following screens show the online NASA-TLX Workload Assessment that was used in the research study (Sharek, 2011; Hart & Staveland, 1988).
**INSTRUCTIONS:**

Please rate all six workload measures on the left by clicking a point on the scale that best represents your experience with the task you just completed.

Consider each scale individually and select your answer carefully. Mouse over the scale definitions for additional information.

Activities will play an important role in the success of this study, and your active participation is essential to the success of this experiment, and is greatly appreciated.

Click the Submit button when you have completed all six ratings.

Please note that the Performance scale goes from Poor on the left to Good on the right.

Submit
**Mental Demand**
How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

**Physical Demand**
How much physical activity was required (e.g., pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

**Temporal Demand**
How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

**Effort**
How hard did you have to work (mentally and physically) to accomplish your level of performance?

**Performance**
How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

**Frustration Level**
How insecure, discouraged, irritable, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?
Appendix G – Final Experiment Participant Questionnaire

Final Experiment Participant Questionnaire

1. Prior to viewing the tutorial, were you familiar with the software that was taught in the tutorial (Camtasia Studio)? (Circle One)  
   Yes  No

   If you answered Yes, do you have experience doing any of the tasks that you were asked to perform in the tests for this experiment? (Circle One)  
   Yes  No

   If Yes, which tasks have you done previously?

   ____________________________________________________________________

2. Do you have previous experience with a similar video editing software program? (Circle One)  
   Yes  No

   If Yes, which video editing software programs have you used that were similar?

   ____________________________________________________________________

3. Did you find the tutorial interesting?

   ____________________________________________________________________

4. What was interesting about the tutorial?

   ____________________________________________________________________

   ____________________________________________________________________

5. What would make the tutorial more interesting?

   ____________________________________________________________________

6. Did you find the tutorial easy to understand? ________________________________

7. Was anything in the tutorial confusing? If so, what?

   ____________________________________________________________________
Appendix H – *Debriefing Form*

**The Effectiveness of Interactivity in Multimedia Software Tutorials**

Lisa Whitman
North Carolina State University

**Debriefing**

Thank you for participating in this experiment, in which you viewed a software tutorial on Camtasia Studio. This study was designed to investigate whether interaction between a student and instructional multimedia software tutorials can be more effective in improving understanding of software programs and how to use them. Interactive, animated, and static versions of software tutorials were viewed by students, who were then tested on declarative knowledge and procedural knowledge to determine which presentation form greater impacts learning. This study also investigated if multimedia presentation type impacted user engagement or cognitive workload while students viewed the software tutorial.

It is expected that this study will reveal that interactive software tutorials will be more effective in teaching the same material than animated tutorials, while static tutorials are predicted to be the least effective for computer-based software instruction. It is also predicted that the instructional benefit of interactivity will be demonstrated with increased user engagement. Research on the effectiveness of computer learning environments and how to optimize their potential as a learning tool is important as computers are becoming an increasingly popular medium for instruction. For example, the results of this study can have implications in distance learning programs and for improving online software instruction provided by universities and companies to students, employees, and practitioners who need to learn software programs outside of a classroom environment.

Your participation in this study is greatly appreciated.
Appendix I – Scale for scoring the procedural knowledge test

Participants were given a point for each of the following tasks accomplished in the procedural knowledge test, for a maximum score of 25.

1. There is a title slide.
2. The title slide has a red background.
3. The title slide uses white font for the text.
4. The title slides says, “Let’s learn about and visit the NC State Belltower” as it appears in the sample video.
5. There is a Fade to Black transition between the title slide and a video capture of the NC State University website.
6. There is a screen recording of the NC State University website starting at the home page.
7. The mouse cursor points to the link, About NC State, with a slight zoom in effect as the cursor points to the link.
8. After the About NC State link is clicked, the mouse cursor is moved to the link labeled The Belltower on the About NC State webpage.
9. There is a zoom in effect on the mouse cursor as it hovers over the link labeled The Belltower and as the link is clicked.
10. There is a zoom out effect after the link is clicked to show the full webpage for The Memorial Belltower.
11. The mouse cursor is moved over to the Location section of the page and the image of the map is clicked.
12. The mouse cursor is moved to the link labeled The Belltower.
13. There is a zoom in effect on the mouse cursor as it hovers over and clicks the link.
14. After the link is clicked, there is a zoom out effect to show the full GooglePlus page for The Belltower.
15. The Directions link is clicked.
16. The mouse cursor is moved to click on the text box to enter a starting destination for directions.
17. There is a zoom in effect on the cursor.
18. The participant types DH Hill Library into the text box and selects it as a destination from the list that appears below the text box.
19. The Get Directions button is clicked.
20. There is a zoom out effect to show the full page, including the path from the library to the Belltower.
21. There is a Gradient Wipe transition from the Directions page to a title slide.
22. The title slide has a red background with white text.
23. The text on the title slide reads, “It’s a short walk to the Belltower from the Library!” as it appears in the sample video.
24. The video is produced to a Web format.
25. The produced video is saved.