5-2017

Situational Intuition: Hierarchical Modes of User Experience in Human-Computer Interaction

Simon Tarr

Situational Intuition:
Hierarchical Modes of User Experience in Human-Computer Interaction

by

Simon Tarr

Bachelor of Arts
The Pennsylvania State University, 1994

Master of Fine Arts
Ohio University, 1997

Submitted in Partial Fulfillment of the Requirements

For the Degree of Doctor of Philosophy in

Library and Information Science

College of Information and Communications

University of South Carolina

2017

Accepted by:

Samantha K. Hastings, Major Professor

Jennifer Arns, Committee Member

Sara Schneckloth, Committee Member

Paul Solomon, Committee Member

Cheryl L. Addy, Vice Provost and Dean of the Graduate School

DEDICATION

To Jen and Caspar, my deepest and brightest loves.

ACKNOWLEDGEMENTS

It takes a village to raise a researcher, and I am grateful for the kindness and wisdom of many people on this journey.

Thank you first to my advisor, Samantha K. Hastings, for your unfailing optimism and just the right insight into each problem along the way. Thank you also to the members of my committee, Jennifer Arns, Sara Schneckloth, and Paul Solomon—you were a dream team to work with as this project developed.

Thank you to supportive colleagues from the University of South Carolina and other institutions: Peter Chametzky, Heidi Rae Cooley, Mark Garrett Cooper, Mary Dalton, Laura Kissel, Ben Levin, C. Melinda Levin, Evan Meaney, and Jeffu Warmouth.

Thank you to my dear, dear friends: Kathleen Maltarich, Mark Maltarich, ZuZu Maltarich, Luke Maltarich, Jennifer Whetstone-Jackson, Grant Jackson, Ava Jackson, and Leila Jackson. Your camaraderie and love have been a source of joy and laughter that sustains me week after week.

Thank you to my parents, Criss and Diane Tarr. You made me into a pretty good student from the very beginning.

Thank you, Jen Tarr. This unusual endeavor took so much time and energy from both of us, and would never have been possible without your love and patience. You never showed any doubt in me or allowed a hint of it from myself.

And thank you, Caspar the Fox—my son and best buddy. You're the inspiration to make myself better in every way I can.

ABSTRACT

This dissertation examines the ways in which novice users teach themselves highly complex software with no prior experience, and how the user experience (UX) of such software affects the information-seeking processes of those who are actively trying to use it.

This qualitative study examined novice undergraduate students beginning to use complex digital non-linear video editing software as they sought to attain proficiency. Data collected in the form of interviews before and after the assigned task of the study, observation of the participants in process, and concurrent think-aloud narration by the participants, provide evidence for a newer extended definition of intuition. This study also proposes a new hierarchical model of UX that could account for this augmented definition.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Problems and Purpose

Media production technology has converged around the personal computer. As a result, many disparate processes in media production workflows such as screenwriting, cinematography, sound recording and mixing, animation, post-production, distribution—each of which used to have its own discrete, unique tool set—now are centered around the computer, and their skills are mapped onto the interfaces of pointer-and-screen (and in increasingly frequent cases, touch). Careers in the media industries have changed along with technology. Newcomers to media production fields are faced with a seemingly endless and ever-changing array of tools that they feel they need to master. Professionals already established in the field need to spend time and resources keeping up with changing trends, sometimes retraining completely.

Take, for example, film editing (I will use the terms "motion picture editing," "film editing," and "video editing" interchangeably, as is the custom in media production fields). To be clear, film editing is not about technical prowess. It is first and foremost a function of mind, but the technology cannot be ignored, particularly because the process is complex. This has always been true, ever since motion pictures were shot and edited on film. For the many decades of film editing, tools and techniques were so specialized and so expensive that a neophyte would typically gain access to them through years of guild apprenticeship. Access became slightly easier as media-making entered education

systems. Now anyone with even an entry-level computer (and more recently, mobile devices) can easily gain access to the primary tools to edit a film, but conceptually the intellectual process is just as complex and difficult as it ever was, regardless of fundamental changes in the user experience of the process.

This leads to the general questions underpinning this study. How do users approach and then learn a new and complex professional software tool? How does the user experience of such software affect that process? Drawing on my own professional background as a filmmaker and media arts teacher, I am particularly interested in how these general questions manifest themselves in the technologies of motion picture editing, a particularly complex and difficult process for new users. By examining this particular subset of novice users undergoing their initial introduction to a highly complex user experience designed for professionals, I hope to gain insight into sense-making aspects of information-seeking behavior, but specific to complex software user experiences.

Research Questions

In this study of undergraduate students beginning to use digital non-linear editing software, I will be looking into the information-seeking behaviors of novice users as they seek to attain proficiency in a complex software system, specifically using professional motion picture editing as the site to examine this phenomenon.

There are two research questions guiding this study.

RQ1: In what ways do users teach themselves how to use complex professional software with no prior experience?

RQ2: In what ways does the user experience (UX) of professional software affect the information-seeking processes of users who are actively seeking to learn how to use it?

Organization and Structure

Using professional motion picture editing software as a prototypical example of a highly complex software system, this dissertation is organized in the following way. Chapter 2 presents a survey of interconnected histories of motion picture post-production and user experience that informed the creation of these research questions. The research literature review then examines the psychological underpinnings of user experience and how human interface guidelines for software design evolved alongside strides in understanding human-computer interaction. An examination of affordances, expertise, schema and sense-making give a framework to situate the theories at play considering these research questions. After situating the role of the researcher, Chapter 3 explains the research methods and study design to describe a way to begin to answer the research questions, including processes for site and participant selection, and steps to ensure data reliability. Chapter 4 presents the results and analysis of the data collected in the study, concluding in Chapter 5 with a discussion of implications for research in UX and potential plans for future study in this and related areas.

CHAPTER 2

LITERATURE REVIEW

This literature review will first survey the history and evolution of motion picture editing as its technological basis shifted from physical/chemical to digital. Next, it will introduce the landscape of user experience as a superset of user interface (UI) and human-computer interaction (HCI). Finally, it will outline current theories of learning, memory, and information behavior (IB) germane to this research.

I have selected studies that are directly relevant, and will present them here, using Lecompte and Preissle's components of a literature review as a method in and of itself to build an argument and situate this study in a theoretical landscape (1993).

Survey and History of Digital Non-Linear Editing

Motion pictures were originally captured on film—long strips of plastic with perforations, coated in a photosensitive emulsion. This film is sent through a camera that takes twenty-four still images per second, chemically inscribing the images into the film's emulsion. Twenty-four images per second uses up quite a bit of film, between 36 and 90 feet per minute, depending on the width (or, “gauge”) of the film. Motion pictures are typically filmed out of sequence, and each sequence may be filmed multiple times (or "takes”) depending of the performance of an actor or success of the crew. Therefore, for the narrative of the given film to hew to the original script, those thousands of feet of film must be edited and put in the appropriate sequence. To be sure, the editing process is not merely chronological sequencing of elements, any more than writing a novel consists of

simply recounting plot events in order. As Soviet filmmaker and theorist Sergei Eisenstein explored in his classic films and then explained in his 1949 book *Film Form*, the concept of "montage" in its many forms, plays upon the juxtaposition of sequenced moving images to create further synthesized meaning, much in the way that a writer might take liberties with space and time in order to create a more complete and resonant story (242).

Since the inception of motion pictures, the method used to edit films consisted of making a copy of the footage to work with by chemically duplicating the strip of film, then literally cutting the duplicated strip of film apart at each different take, and finally splicing it back together with a tape or cement adhesive. This method was laborious and expensive. Laborious, because it usually took a professional editor and many assistants a significant amount of time to piece together a completed version of a film due to the enormous physical constraints of handling and organizing countless thousands of feet of film. Expensive, because highly skilled labor is costly, and because of the considerable expense of the film stock and chemistry to make workprints of the film to edit.

Magnetic videotape was a development of the television age to store and broadcast programs, and in the 1980s when videotape decks were affordable outside of television stations it became common for some motion picture editors, especially those with lower budgets, to have film labs transfer the original motion picture film to tape instead of printing a copy onto a film workprint. Using two videotape decks (one to play and one to record), the editor could copy the desired shots into sequence and build the film that way. It was much more space efficient to have a shelf of videotapes instead of

several dozen thousand feet of film, and it was a simple matter to make a new draft of a film: just insert a fresh tape and start over.

However, along with the advantages, there were also many disadvantages to this method. First, even the most expensive, highest quality videotape machines could not come close to matching the image quality of film. Second and most importantly, videotape is linear. This is to say, videotape must be recorded and played back in real time, and a program must be built in sequential order. If an editor wanted to add a five-minute shot to a scene on actual film, she could splice that shot onto the end of the film and her work on that idea is completed. On video she would need to copy that five-minute shot into the sequence, and it would take five minutes. Using film, if she edited a two-minute shot after that first five-minute shot, and then changed her mind to decide she actually wanted the two-minute shot to come first, she could simply un-tape the film and move it to the front of the first shot, and that action would take a few seconds. But on videotape she would have to start over, copy the two-minute shot first, and then copy the five-minute shot after it. One final frustration of editing on videotape came at the end of the entire production. To be able to use the edits from the video on the original film to create a theatrical release, editors had to make an "edit decision list" (EDL) of timecode numbers: a log that matched a number assigned to each frame of video back to a numbered frame from the film. Specialized technicians would then cut the original film negatives according to the numbers in the EDL database to create the final version of the motion picture. This process was cumbersome and fraught with error.

Motion picture technology impresario George Lucas, along with other *Star Wars* co-conspirators, invested significant research and development into computerizing the

film editing process. In 1984, Lucasfilm Ltd. unveiled EditDroid, an $87,000 terminal that could edit film using an array of laser discs containing a film's footage. Personal computing had still not reached critical mass in the marketplace, let alone ubiquity, and the concept of using a computer for something as specialized as film editing seemed so far-fetched that EditDroid even made ripples in popular media. Described in Discover magazine:

> Seated at its controls like Luke Skywalker piloting his spaceship, the film editor can rearrange footage almost as easily as a text editor at a word processor can juggle words and phrase—inserting and deleting frames, taking a sequence from one place and shifting it to another. (Gannes, 76)

EditDroid did not find its way into many film editing suites, possibly because it was not much of a speed or cost improvement, or maybe because of the abrasive tone that Lucas took when EditDroid was introduced. Describing traditional film editing, he said, "anybody who's worked with film realizes what a stupid nineteenth century idea it is" (77). A unionized guild of film editors might not jump at the opportunity to embrace an invention predicated on a wholesale dismissal of an industry's standard process. Nevertheless, the notion of modifying a mechanical/physical process into a digital one was introduced and the path forward for the industry was set.

It would be five years before Massachusetts startup Avid, Inc. released the first version of its Media Composer software for the Apple Macintosh IIx. The underlying technology was a significant improvement from what was available in 1984. The Macintosh computer was already a popular choice for graphics professionals and writers,

and flexible storage such as hard disks and rewritable optical discs, though expensive, were available.

Of all the features that Avid foregrounded in its promotional materials, one stands out: the user experience for established editors. In the first promotional video Avid released in 1990, several professional editors were shown working with the Media Composer, and then they described the experience in terms that other editors would understand. In the video, editor Tom Schachte explains, "you don't have to learn to type on the keyboard, you don't have to pay attention to timecode numbers. Just like editing film, your reactions can be based purely on reacting to the picture" (1990). Basil Pappas continues this comparison with traditional film-based editing equipment by testifying that "the idea of the timeline graphically [lays] it out just like a synchronizer" (1990). A “synchronizer” was a sprocketed device that allowed multiple strips of picture and sound to stay in sync, with the added benefit of giving the editor the ability to stand over the film being edited and see the large sections of the film laid out in a glance. The “timeline” has been a nearly universal interface element since the beginning of digital editing, and it also allows the user to glance at large stretches of time at once. This key conceptual element to editing film quickly and efficiently, which was not available in linear video editing, was built into Media Composer from the beginning and was inherited by software that followed.

Avid's early strategy of targeting film editors was not just a marketing strategy. The entire user experience paradigm of the Media Composer application was designed around entrenched workflows of the film editing community. The command to add video to a sequence is "splice," named for the act of taping two cut strips of film together. A

segment of video is called a "clip" or a "subclip," named for the clipped ends of film, and they are arranged in Media Composer within folders named "bins," named after large metal bins used to hang film sequences before they are spliced into sequence. Locking multiple tracks together in the timeline was referred to as "ganging," which is how editors referred to locking tracks together in a synchronizer, and making fine adjustments to an already-edited clip in a sequence activated a mode called "trimming," which is exactly what one did with a blade when using film. Taken as a whole, this process became known in opposition to the linear real-time videotape editing as "digital non-linear editing" (DNLE), or sometimes just called "non-linear editing," even though the original method of cutting and splicing film was already non-linear.

The decision to create a user experience specific to film editors worked. By the mid 1990s, Media Composer was the de facto DNLE standard, making its way into post-production facilities and even film schools around the world despite its high price. User experience design, by signaling the long-standing tools of film editors, had smoothed the transition and had facilitated an entire industry's shift to a new platform.

In 1998, Apple Computer, Inc. purchased the assets of an unreleased application called Final Cut from Macromedia, Inc. Apple finished the application's development, and released it as Final Cut Pro 1.0 in 1999 to stabilize their position in the professional video market. With dwindling market share in personal computing and limited cash flow, Apple's future at the time was very much in question. Avid had just announced that it would be releasing Media Composer for Microsoft Windows, with certain advanced features available for Windows exclusively. However, just as film editors were loath to abandon film for a new technology until it was proven to be a smooth transition, digital

non-linear editors proved largely unwilling to migrate to the Windows platform, and Apple's new editing software found a receptive audience.

The user experience for Final Cut Pro from its introduction through version 7 was similar to the one pioneered by Avid with Media Composer. Bins, clips, and trimming were all present; it even had icons with tiny pixel-based sprocket holes representing footage, just as Media Composer did. It was simple for editors who were used to Avid software to make the switch to Final Cut Pro, and it was made even more attractive with a significantly lower price, and the ability to run on computers without special added hardware. Now instead of an editing system that cost as much as a house, a filmmaker could set up a digital editing system for a few hundred dollars, using their current computer. By 2011, Final Cut Pro had captured 54% of the video editing market share according to research film SCRI, with Avid and several other competitors fighting over the remaining users (Pogue, 2011).

The user interface for Media Composer has remained largely unchanged since its original version, offering slight revisions to update the graphics over the years. This strategy to maintain familiarity and security for industry professionals is in keeping with the larger user experience strategy that initially lured film editors to digital platforms. Apple appeared to be following the same path with Final Cut Pro, leaving both the user interface and the user experience consistent through its first seven versions.

However, in 2011, Apple unveiled a major revision to Final Cut Pro—such a change that they skipped two version numbers and labeled it version 10, using the Roman numeral "X" as they did in the Macintosh operating system. Final Cut Pro X was rewritten from the ground up with significant technical improvements and a sleeker

graphic interface. But packaged with the improvements was a completely new user experience. Apple promoted this new user experience as "revolutionary." This is not a new claim for a technology company, but Apple went further to describe the software as "a whole new production" that "breaks free from the restrictions of old-fashioned timeline tracks." In promotional videos, Apple described Media Composer as "traditional," and "built on old organization concepts" (2011). Final Cut Pro X had no "bins" but instead had "events" (a concept borrowed from Apple's own consumer applications iPhoto and iMovie). There were no nods to film images or terminology—no ornamental sprocket holes, no "tracks" in the timeline to evoke the concept of discrete channels of image or sound. Apple's claim was clear: future editors would work in a new way, and current editors should get used to it.

The response was dramatic. While many editors agreed that there were many positive aspects to the new program, the industry moved away from Final Cut Pro X, partially because many functions still necessary for professional work were removed, but also because of the user experience. One editor and educator noted that Final Cut Pro X was a "brilliant program, provided the user can essentially forget everything they've learned" (Cheng, 2012). Apple has since released several updates to Final Cut Pro X that replaced functions that had been removed, but the damage had been done. Professionals and educators, unsure that the new software would be worth the time to learn and leery of a company that would try to rewrite an industry's workflows in a single software release, took refuge in other software.

These differences in user experience, and their relation to widespread adoption are important to highlight, as this dissertation will examine the relationships between user

experience, sense-making, and learning as they relate to professional media software and processes.

Psychology, Human Interface Guidelines, and User Experience

User experience (UX) is often used as a synonym for user interaction (UI) or human-computer interaction (HCI). However, Bargas-Avila and Hornbæk (2011) specify further and describe UX as a separate and emerging phase of HCI that focuses on affective, emotional dimensions (2689). This evolution of how UX is considered has been proceeding, and Law, et. al. (2009) conducted surveys on industry and academic perceptions of UX in the hopes of refining its scope toward a more comprehensive definition. Hassenzahl and Tractinsky (2006) mapped out what they saw as a research agenda for the entire discipline to flesh out UX as a multi-faceted framework to consider HCI from experiential, emotional, and holistic points of view (95).

MacDonald and Atwood (2013) extended a historical analysis by Kaye and Sengers (2007) of how human-computer interaction is evaluated and they describe five distinct phases that emerged since the 1940s: the System Reliability Phase, from the 1940s to 1950s, the System Performance Phase in the 1950s and 1960s, the User Performance Phase in the 1960s-1970s, and the Usability Phase from the 1980s to the 2000s, and the User Experience Phase from the 2000s to the present. These phases described the limiting factors of what would have been considered successful technological use for the respective periods. For example, during the reliability phase of HCI the computers would break with startling regularity, so the reliability of the machines themselves was the restricting element in determining success of the computing experience. Later, the restricting factors were how well the computer could perform now

that they weren't breaking regularly (system performance phase), how efficiently a human could perform now that the user no longer has to wait around for the computer (user performance phase), how readily a user could learn to use the system without extensive training (usability phase), and ultimately how users feel as they are engaging with the computer (user experience phase). MacDonald and Atwood presented these phases as part of a call for better, more holistic ways of considering and evaluating user experience, as the current state of examining and evaluating HCI is too limited.

For those who develop software, companies often publish Human Interface Guidelines, often referred to as a "HIG." This practice was started by Apple, which published the *Macintosh Human Interface Guidelines* in 1992, updating it regularly (in 2016 it was called the *OS X Human Interface Guidelines*) and even spinning off a separate HIG for its mobile operating system iOS, a distinctly different experience from the desktop UX. In an attempt to unify the look-and-feel of applications from various developers, other companies began publishing HIGs. As of this writing, there are HIGs for Microsoft Windows, Oracle Java, and different Linux interface environments such as KDE, GNOME, and Ubuntu. In March 2015, Apple published a HIG for the Apple Watch, codifying how designers should consider "glances" at a tiny wrist-based screen (11).

One Apple UX example that remains unchanged over twenty years is the use of "metaphors." The Apple HIG begins by enjoining software developers to consider end users' prior knowledge of the world around them into their designs. For example, users "organize their hard disks in a way that is analogous to the way they organize file cabinets" (p. 4 in 1992, p. 27 in 2012).

A newer addition to the Apple HIG is the concept of the “mental model.” Apple specifies that users always have a preconceived notion of the job they are trying to accomplish with a piece of software, and that the user experience should support that.

> Be aware of the model’s inherent metaphors, which represent conceptual components of the task. In the letter-writing example, the metaphors include letters, mailboxes, and envelopes. In the mental model of a task related to photography, the metaphors include photographs, cameras, and albums. Strive to reflect the user’s expectations of task components, organization, and workflow in your window layout, menu and toolbar organization, and use of panels. (28)

Given the examples in the evolution of Final Cut Pro to Final Cut Pro X, it would seem that Apple violated a fundamental principle of their own HIG. But this is not all that surprising, as prominent technology blogger John Gruber writes: "the HIG is dead. It died long ago. And it was Apple that killed it" (2011). Gruber responds to complaints about a proliferation of non-HIG-compliant applications that accompanied the recent popularity of Apple products by pointing out that Apple itself always felt free to experiment with interfaces, particularly in their major applications, for example Final Cut Pro.

Another notable evolution in the HIG is a greater emphasis on the importance of intuition in software design. This stands to reason, since if manuals are seldom used and the primary tool to learn a program is the software itself, the means to use the software should be apparent. Apple again relies on the more recent concept of the mental model to admonish software designers to create intuitive interfaces, and in this context they mean visual interface elements that comport with what users might expect based on their mental models (2012). But where does this concept of intuitive software come from?

What does intuition really mean when it comes to how we think about a new situation, and how can it apply to user experiences with software?

In psychology, Gibson developed the concept of "affordances" to help explain visual perception; an affordance is a specific relationship between a being and its environment that presents an opportunity for a specific action (1979). Norman adapted the concept of affordances in his first edition of *The Psychology of Everyday Things* (1988), later released in paperback as *The Design of Everyday Things* (1990), and applied it to describe the relationship between a person and a physical object that enumerates the possible actions by the person on that object. For example, a mug affords drinking, due to its ability to hold liquid, but its weight also creates an affordance for keeping papers on a desk from blowing away. The affordance to drink only exists, however, when the relationship with a person allows it: if a person lacks the motor control to use an open-topped container, or to hold the mug by its handle, the affordance to drink no longer exists because that relationship precludes it.

Norman's concept of affordances was introduced when interaction design was pre-digital, and it caught on in nascent graphic design and software development communities, but without real understanding of the concept (1999). Designers might add a button or a digital interface element, and refer to it as an "affordance," and would misuse the term to refer to an interface element that had been designed to be intuitive. But Norman insisted that user interfaces on a screen could not be affordances, they are merely signals that indicate possible perceived affordances (1999). Significant revision was necessary in his 2013 edition of *The Design of Everyday Things* to correct this, and he added the new concept of "signifiers," signals to the existence of possible affordances

(p. 28). A digital example would be a common sliding pane on a smartphone, invisible until the user swipes down, containing a summary of the day's appointments, weather, and other notifications.

The information pane that slides down affords users views of that information, but the small graphical element that communicates that there is something to do at that edge of the screen is a signifier that there is an available action. This pairing of signifier and affordance may indeed be considered to be intuitive elements for the user to interact with, in the sense that Apple means: the interface elements comport with users' mental models of how an interaction *should* occur. However, the conflation of affordance with intuition breaks down quickly. Consider the prior example of a sliding pane on a smartphone, to be intuitively swiped by the user. A signifier may point to an affordance in the design that is perfectly understood by the user, but any hope of intuition in that design is predicated on users' prior knowledge of smartphones. While so ubiquitous at the time of this writing as to seem nearly obvious, fifteen years ago there were no examples of featureless glass screens for our fingers to skitter across. Herbert Simon was blunt about intuition from a point of view of problem-solving and expertise when he wrote that "intuition is nothing more and nothing less than recognition" (1992).

If intuition is a function of relative levels of expertise and familiarity, then what happens when disconnects occur, for example when completely new and untrained user encounters a complex problem such how to use a professional software program that assumes a high level of expertise? This interaction between users and software, affordances aided by signifiers, is the primary site of examination for this study of novices approaching a new tool, and the examination of how users identify signifiers that

communicate affordances will come into play. HIG documents give some clues as to how users are intended to use software, but only general use software that consumers might need for everyday activities. But software developers design applications for professional and highly specialized use, where mental models may not readily exist and where an intuitive course of action or affordance might not be available to users. One of the aims of this dissertation will be to try and discover what aspects of UX influence new users' entry into more complex professional applications, and how they learn to wield them as tools when typical affordances and signifiers are missing, thereby diminishing the possibility of a so-called intuitive experience.

Schema, Artifacts, and Sense-making

Learning how to use professional software is difficult. Understanding how users approach this difficulty represents complex territory in HCI and learning because professional software applications present two learning problems at the same time. First, the user needs to use a process to solve a problem ("I must turn these hours of video into a short, compelling program"), and second, the user needs to learn to use an artifact ("I must learn this software in order to solve this problem"). There are some learning theories that apply to the first problem solving process and others that apply to the artifact learning process. This section looks at both categories.

Many aspects of learning are related to memory, and applicable theories derive from cognitive psychology. In 1932, Bartlett published his seminal work on remembering, which introduced schema theory, proposing that certain patterns form in the mind, and those schemata inform what we remember. This theory is still important to understanding the mind and has informed and expanded into more specific theories of

memory, and also to theories of teaching and learning. When schema theory is applied to learning, researchers apply the concepts of memory to describe both the formation of new schemata in the learner as well as how new information is incorporated into a learner's pre-existing schemata (Swanson, 2012).

Implicit learning is a learning theory that extends schema theory. Introduced by Reber in 1989, implicit learning describes knowledge acquisition processes that occur outside of independently concerted and deliberate attempts to acquire such knowledge. This process does not rely on conscious recollection; in other words, giving an exam on such material would not reveal the learner's understanding. However, this mode of learning is critical for language acquisition (Conway, et al, 2010). Research in this area may be important to testing how users react to a given UX to solve a problem.

In addition to problem-solving and process-oriented theories of learning, theories that apply to artifacts are also applicable to this project. Activity theory in particular, as introduced by Bødker in 1989, proposes that the study of the use of an artifact is the only way to understand that artifact. Bødker's body of work is of particular interest to this study, as it focuses more on professional users than casual ones. Bødker later singles out software as useful to study using activity theory, since, though the software is an artifact, the user tends to focus on the activity being accomplished through the artifact, rather than on the artifact itself (1991).

Ryan found that users changed their patterns of use when learning artifacts through exploration alone, molding their expectations to their understanding of what the artifact could do (2011). This is also important to understand when looking at learning

artifacts longitudinally, but does not reveal information about initial discovery and how this is facilitated through the user experience.

Kurtz claims that perceived usability, specifically the aesthetics of a user interface, had no effect on learning (2010). However, those results are not relevant to this study, since those experimental variables had significant shortcomings. The variable user interfaces that were presented to test subjects were limited to only surface aspects of user interface and aesthetic appearance rather than UX or usability. The study examined only very basic aspects of layout and design, and did not take into account the multivariate nuances of UX. Using any HIG, the user interface examples presented in Kurtz's study as "high beauty," "medium beauty," or "low beauty" would all be considered aesthetically unpleasing examples.

Peterson, Madsen and Kjær demonstrated that becoming accustomed to the use of a technology is an evolutionary process (2002). Performing a three-month study of two families with new television sets with video recorders, Peterson, et al. interviewed the participating families about their use of the new home theater technology, and also observed the families during hands-on sessions using the television systems. This study used a functionalist theoretical framework, looking at a social system in action and seeking to describe it in order to understand and analyze how the families in the study learned to use the new television system. The methodology of Peterson, et al.'s study was ethnographic and interpretive, using observations over an extended period to draw inductive conclusions. Peterson, et al.'s study is important from the perspective of learning, specifically the long-term evolutionary nature of human-computer interaction (HCI), but it does not tell us anything about how the user experience (UX) in particular

relates to the process of system learning related to the user experience, or how (or even if) it contributes to success in learning the system.

Law, et al. demonstrated that despite attempts to create international standards for and definitions of components of user experience, UX remains an elusive, variable field to pin down (2009). Using a survey of 275 experts in UX from across academia and industry, Law, et al. asked questions about definitions of aspects of UX, and about participants' feelings about those definitions. This study had a functionalist framework, attempting to analyze a complex system, with an interpretive methodology, inductively determining a population's opinions about a subject by analyzing many survey results where individuals registered many different opinions. The results indicated that user experience is highly subjective and particularized, making general definitions of the field difficult. The particular focus on inconsistencies in how UX terminology is applied helps clarify and taxonomize the field, but does not help to understand how it is used.

Wiedenbeck and Zila (1997) demonstrated that combining a directed exercise with users' exploration of a new software application increased the users' success in learning that software, regardless of background experience. One hundred and two students with varying levels of computer expertise learned a new piece of software and were then tested on the use of the software after being given specific tasks to accomplish. This particular study had a behaviorist theoretical underpinning, seeking to perform a controlled test to reliably explain and predict a phenomenon. It points to a positivist methodology, using a precise experiment to describe human action and behaviors. Wiedenbeck and Zila's study takes a different approach to explain behaviors that can

affect successful learning as it relates to HCI, but it does not address UX at all, let alone how it relates to the process or success.

In 1980, Dreyfus and Dreyfus proposed a multi-stage model of skill acquisition, which describes levels of mental activity as learners progress from the ranks of novice to master (p. 15). The Dreyfus model, and the studies that underpin it, specifically pertains to the development of training procedures. For example, the Dreyfus study was designed to improve aircraft piloting in the US Air Force, generalized to developmental training (p. 16). However, the development of expertise as a user acquires skills as a novice may be an important byproduct when considering what role UX plays in the process.

Indeed, piloting aircraft may be an apt comparison to the kinds of complex processes that comprise this study. Sarter and Woods (1991), examined a phenomenon of expertise commonly used in military and aviation circles as "situational awareness," a contextually-aware, time-sensitive ability to gather and filter information in performance-critical and information-dense domains such as an airplane cockpit or combat scenario. They identify a robust feedback mechanism as a critical prerequisite to allow situational awareness to emerge (Sarter & Woods 1991). As an example in the context of a flight deck, such a feedback mechanism would certainly be, at least in part, the interface with the aircraft. Related industries have approaches to designing interfaces with situational awareness in mind (Endsley 1988, Endsley & Jones 2004) to guide aeronautics industries, military applications, power grid infrastructure and medical instrumentation. However, the concept of situational awareness might share some similarities with complex user processes where lives are not on the line, such as the professional video editing processes of this study.

Dervin's body of research on sense-making was an initial impetus for this study, providing a compelling model to consider information-seeking behavior as a journey to traverse one or more "gaps" in the seeker's understanding (1983). The sense-making model developed by Dervin encompasses much of the entire process of living and learning: the information-seeker's past and present, the contours of the knowledge to be sought and known-unknowns, the complications caused by unknown-unknowns, the list stretches on (1999). This concept of a gap in knowledge that is affected by ontology is important for this study, in terms of attempting to define the depth and complexity of the unknown as well as the means by which a user might attempt to traverse such a gap.

However, sense-making as a model can be unwieldy simply because it attempts to account for virtually everything in the information-seeking process. In a previous study that I conducted, which served structurally as somewhat of a pilot study for this one, a small cohort of undergraduate users who were similarly inexperienced were introduced to DNLE software by editing a segment from a 1958 television program. The results of that study suggest that users' past experiences act as a kind of "cultural filter" that dramatically affect how they approach new complex tools, and that modeling specific filters that can account for privilege in information behavior might allow researchers to consider privilege more effectively than by generically ascribing it to "past experiences" in an overly complex model (Tarr 2015).

That previous study raised a number of issues that I have integrated into the design of this dissertation. It was a pilot study in the sense that I tested processes of site and participant selection, pre and post-interviews, and the mechanics of assigning the complex task of video editing to new users. However, the research goals of that study

were very different. The previous study focused on the results of the users' tasks, and examined them in light of their backgrounds, specifically looking for connections to their pasts that might indicate that current models of information-seeking behavior should be updated. Since that study focused on the end results of the editing task, and participants recalled experience of the process of using the software, but did not include direct observation of the participants' activity while interacting with the UX. This dissertation will gather much more observational data during the editing process. While the end results of the participants' tasks will certainly be examined in this study, the task that participants are set to is simpler and less content-based, since this study is concerned with studying aspects of user experience, rather than examining the content produced.

Gaps and Conceptual/Theoretical Perspective

In addition to the several gaps in the literature that I have presented here, there are no studies that relate specifically to professional media production software as it relates to UX. The study that I am proposing will fill these gaps by looking at a group of new users as they begin to use an unfamiliar piece of professional software, and by observing their user experience as it relates to their endeavor to make sense of it and gain proficiency.

I suspect that there is a complicated relationship between highly complex tasks and complex tools used to perform those tasks, and that relationship is informed and negotiated by the user experience of the tool. To put this in terms of this study: the act of editing a motion picture is complex and difficult, and the software used to edit motion pictures is complex and difficult to use. While the act and the tool to perform the act are related, they are two separate things. I speculate that the design of such a software tool,

along with its user experience, plays a significant part in the success of a new practitioner in both learning the mechanics of the software, and learning the craft that is enabled by the software. This literature influenced me to conduct this study in such a way that I will record and observe participants as they are involved in a set of basic tasks that can only be accomplished with a given piece of software.

CHAPTER 3

METHODS

Situated Knowledge and Related Assumptions

I come to this research project with many personal and professional investments. I got my first filmmaking degree from Penn State in 1994, where I was taught to edit movies first by taping strips of film together, and then by copying bits of videotape from one machine to another. I found both of these processes frustrating; they took much of the joy out of the process of editing for me. Later, when I went to Ohio University to work on my Master of Fine Arts degree in film, I was particularly interested in the Avid Media Composer, which at the time was starting to take the motion picture post-production industry over completely. In fact, the progression from editing film to linear video to digital non-linear process as described in the past chapter is an evolution that I personally experienced my own career.

For my first job in 1997, I returned to Penn State to administer a lab, where I maintained and repaired two Avid Media Composers that had been donated by wealthy alumni. These two machines were shared by scores of students who would do most of their editing using the older methods, only using the Media Composer for their final polished version. The students had to learn the creative concept of editing one way, and then learn how to use the Avid software by way of the most common method to teach software at the time—by having an expert sit with the student alone or in a small group. The expert would show them around the interface, taking turns clicking on the prescribed

buttons, with a few thousand pages of manuals sitting in a handsome box on the desk nearby for orientation and reference.

Over the years, those manuals got smaller. Soon they disappeared completely, replaced by on-demand help systems. The way that people looked for information to learn how to use the software was changing. However, I noticed that the methods of teaching didn't change. Computers got much faster in the mid-2000's, and professional software got much cheaper. Instead of buying a film editing system for the cost of a house, people could get it for the cost of a monthly cell phone contract.

When I was hired by the University of South Carolina in 2006, there were hundreds of students coming into the media arts program, but there was a curricular bottleneck—everyone had to take a class called Digital Media Art Fundamentals, but there weren't enough sections offered. The art department was spending a fortune hiring temporary faculty to teach a flotilla of sections of 15 people each. Even then students often couldn't get into the class, as a result they would become frustrated and switch to another major. My first year at USC, I taught four sections of this entry-level course. I'd get squashed into a little computer lab with 15 or 20 machines, and teach what was expected. By the end of the course, the students weren't particularly skilled technically, and I hadn't been able to spend nearly enough class time on important subjects like aesthetics or content.

I devised a method to drastically change how students gained technical proficiency. I made short, targeted online videos that students could follow at their own pace outside of class to help guide them through the process of learning the software. The philosophy behind the videos changed the conventional teaching of software, working to

get students to be "just good enough to be dangerous." The key is that the videos didn't actually teach specific software very much. Instead they taught how to learn how to figure out any software. We were then able to spend the time in class working on high-level concepts. I changed the course to seat one hundred undergraduates, and quickly everything got better.

Based on my experiences in academic fields and the literature on both active learning (Bonwell and Eison 1991) and implicit learning (Reber 2004), I have developed an assumption about the process of learning how to use software: that it is generally better to learn those processes through relevant directed tasks outside of a classroom environment. Based on my experience as a professional filmmaker and the literature on user experience (Peterson 2002) and human interface guidelines (Apple 1992), my presumption is that the user experience of a piece of software is designed purposefully to be learned and used in a way that is gradual and cumulative. These led to my interest in this study to investigate how UX facilitates that accumulation of knowledge, and to apply that new understanding to a better integration of that process into a comprehensive understanding of how we develop expertise.

Role of the Researcher

With respect to this study, its site and participants, I bring several subjective personae to the project that must be considered in its design and execution. First and foremost, I have status issues that are right out in front of me all the time. It is critical to acknowledge and deal with up front is the fact that I am a professor, and that the people in this study are students. No amount of good rapport or kind, casual collegiality can take away that fact that I am in a power position over them; I assign grades, I may eventually

write them letters of recommendation. I must take care, as the ironic weakness of my position is the fact that I wield a great deal of power.

However, this subjectivity as a professor and filmmaker puts me in a strong position to pay attention to many details that might escape researchers who are not experts in this craft, who do not also have to be concerned about newcomers to the craft, or who may not approach software and design with the critical eye of a user. If I attend to and keep this power that I wield observed and in check, it can be a source of strength and creation of knowledge. My position as an expert, and the opportunity to help shape newcomers to the film and media industries, means that I can look critically at the design of the tools used in that industry, as well as how students are brought into the field, in a trustworthy way.

Methodological Approach

I used an action research approach to this study, selecting one class that I teach as the site of this study. Describing the specific details of this site will make clear why it was the right choice for this particular study. Action research serves three primary purposes: to improve a practitioner's understanding of their specific practice, to improve the conditions that inform that practice, and through these to improve that practitioner's approach to the practice itself (Kemmis 2009). This approach raises a complicated question. If the focus of the study is the user experience of NLE software, and if the students in question are going to be teaching themselves how to use this software, then why should this be an action research study?

The course that serves as the site of the study, Digital Media Art Fundamentals, is a course in art making where students create projects to solve specific problems using a

variety of tools. For example, in one project, students were assigned a location in the surrounding neighborhoods and presented visualizations of that space twenty years in the future. However, students were not told how to accomplish this. A major part of the project is to imagine the content, decide the best way to create those static and moving images, and then gain the skills in the specific tools necessary to accomplish the project.

As stated earlier, it is clear from the literature that learning a technological skill is an evolutionary process of accustomization (Peterson, et al 2002), that this process works better when active rather than abstract learning is involved (Bonwell and Eison 1991), and when the process is outside a traditional classroom setting (Reber 2004). In the course that serves as the site of this study, the specific mechanics of learning any given piece of software are self-directed by the students. Thus, the students are indeed at the helm of their own learning of the tools. But every aspect of how that process integrates into and impacts the larger course design is critical to the success of the project, the class, and ultimately each students' development of expertise.

My approach to this study comes from my background and profession as an educator, and while I hope the findings of this study may be put to educational use, the study itself is not pedagogical. This research is intended to examine the experiences of the novice user subjected to a complex user experience, and is therefore designed to examine new users before they have learned the complex professional software they intend to use. The focus of this study is on how the design of the user experience adds to the understanding of that process. This self-directed learning process is inextricably entwined with the course I teach where the novice students undertake that process. I hypothesize that the user experience of the software in question is likewise inextricably

entwined with self-directed learning by novices. For these reasons, action research is the best methodological approach for this study.

Employing a battery of qualitative methods to collect data, described in detail below, this study is not intended to produce generalizable findings that can be applied to large populations. Rather, it is designed to intensively understand the novice experience by deeply examining new users in a human-computer interaction designed for professional use.

Site Selection

There are several criteria that a site would need to have to be suitable for this study, action research notwithstanding. By applying Patton's selection strategy of criterion sampling, which prioritizes satisfaction of specific criteria, this site would fulfill the most important needs (2007).

Since this study hinges on digital non-linear editing as a process, the software to be tested should be an application that is commonly accepted as a professional industry standard. There are three pieces of software that fit that bill: Avid Media Composer, Apple Final Cut Pro, and Adobe Premiere Pro.

Media Composer was the original NLE that met widespread use, but still retains many aspects of its original intent to lure editors from traditional film-based cutting. As a result, the system is extremely technical and relies heavily on archaic celluloid film editing jargon—not an ideal candidate for newcomers. Final Cut Pro usurped Media Composer as the dominant software in the early 21st century. However, in 2011, Apple unveiled a major revision to Final Cut Pro—and while many editors agreed that there were many positive aspects to the new program, the industry moved away from Final Cut

Pro X, and older versions of Final Cut Pro are no longer not legally available for purchase. This leaves Adobe Premiere Pro (sometimes referred to simply as "Premiere"), which has been growing in popularity both in the industry and in educational settings.

There needs to be a sufficient number of possible participants with no prior experience in using Premiere (or any professional digital video editing software), in order to focus only on participants' initial responses to the UX. Those participants would also need to have a specific desire to learn to use Premiere either to go into filmmaking fields or to accomplish specific professional video editing tasks, rather than just to pass a test for the class or add a line to their resume. This will ensure that there is a significant inherent drive in the participants to learn the software and retain that knowledge. Since the intersection of these two groups of potential participants in such a course might be small, the course will need to be relatively large ($n>75$).

Finally, the participants and I would also need access to a computer or computers that have Premiere installed (any such machines would have to be sufficiently powerful to adequately run the software).

My class at the University of South Carolina, Digital Media Art Fundamentals, is a 100-person class where novice students gain proficiency in professional media production tools (including Premiere) and then apply those tools to media art projects. To perform these tasks in the class, students are able to use many different computer labs on campus that have machines with Premiere installed, including a small lab which I have access to administer. Even if this study were not based in action research, this course would be an excellent candidate site.

Participant Selection

Turning again to the criteria that lead to Digital Media Art Fundamentals as the choice of situation for the study, I again apply Patton's (2007) selection strategies. I first used criterion sampling to reduce the pool of potential participants, and then purposeful random sampling to arrive at "reasonable coverage" of the phenomenon, which I plan to address in this study (p. 246).

Because undergraduate students come from varying educational backgrounds, some of their high schools will have provided them opportunities to become familiar with professional media software, either through a class or a workshop, or even a school television station. Therefore, it was necessary to reduce the pool of possible participants from students in the Digital Media Art Fundamentals class down to people who have not previously used any professional video editing software, not only Premiere as a specific application. Participants need to have had no prior professional video editing software experience because several common interface idioms are shared across several high-end video editing applications. If a participant was familiar with a different program, they might know what sort of structures to look for in advance, and that would make this study's data less useful.

Students that take Digital Media Art Fundamentals come from different parts of the University—some are looking to be filmmakers, others might be hoping to become web designers, graphic artists, sound designers. Still others might be computer science, business, or philosophy majors taking the course to fulfill a general education requirement in art. While all of them will need to demonstrate a certain level of proficiency in professional video editing, only some in the class will have a specific

desire to gain that skill independently of taking the course. Another way of putting it: some students take this course in spite of one of the skill proficiencies acquired, including video editing, and someone with a disinclination or ambivalence to the specific skill would be a less ideal participant in this study.

Past experience teaching this course suggests that an estimated 70-80% of the students will not have experience with professional video editing software. Separately, about half of a given class is more interested in filmmaking skills than other skills. These sets of possible participants might intersect to varying degrees, but typically represent a significant number of students from the class. If there are one hundred students in a class, then the pool of possible participants might be as many as twenty or thirty. Of that set, my initial estimate was that ten to fifteen people would be willing to participate in the study, in which case I would invite all those who indicate willingness to participate. However, in the case that the pool of potential and willing participants is over twenty, there could have been some data management problems stemming from having such a large pool, and then I would take a purposeful random sample of ten to fifteen. As Patton (2007) indicates, a qualitative study such as this focuses more on the depth of analysis of a smaller but more intensively studied sample, in which case a purposeful random sample of potential participants would be appropriate to reasonably describe the phenomenon being explored, while still maintaining confidence in the data.

In order to first determine the interest level in media art in general and video editing specifically, I observed a discussion thread in the class Facebook group where students would introduce themselves to the other students, talk about their major, their interests, and reasons why they decided to take the course. As expected, responses varied

widely from students across many different majors. Observing these self-introductions served as a first opportunity to identify experienced users who would not fit the criteria for this study. Comments such as "I have always really enjoyed shooting and editing videos for projects" or even as forthcoming as "I have worked a little in the field using filmmaking programs such as Adobe Premiere" would flag someone as having too much advanced knowledge for this study. Similarly, people who identified themselves as having a strong primary interest in a field that is not related to or does not involve professional video editing were also considered as less suitable for this study. For example, I considered someone who indicated that they "would like to work in the finance industry" or "my dream is to design sports apparel" to be less suitable for this study due to a generally lower level of motivation to gain the particular set of skills at the heart of the study. This left students who expressed a general desire to either move into filmmaking and media careers, or a desire to augment their other fields in ways that specifically use media and filmmaking skills. This resulted in an initial pool of 31 potential participants.

This first pool of participants then received a preliminary questionnaire (after receiving initial consent forms) with two questions to further narrow the participant pool: "have you ever edited motion pictures before" and "have you ever used Adobe Premiere Pro before?" There was a follow-up question that asked, "if you have edited motion pictures before, but have not used Adobe Premiere Pro, what programs have you used" in order to determine if potential participants had used another professional program such as Final Cut Pro, or a basic consumer program like iMovie or Windows Movie Maker. For the purposes of this study, only potential participants who self-identified as having no

prior experience editing motion pictures with a professional video editing program were invited to participate: a total of 21 people. In addition, there were two people who described particularly strong desires to learn how to learn editing skills, but had disclosed limited experience with basic consumer programs. They were invited to participate, but declined. Eight other invited participants also declined due to concerns about time constraints or disinterest in participating. This resulted in a total of thirteen participants who met the criteria of showing specific interest in learning how to use professional video editing software, while reporting no prior experience in using such tools.

There are some important limitations and assumptions to declare when considering these participants. First, every possible participant has applied and been accepted to university study, and has the means to attend. This is a massive privilege that cannot be ignored. Second, I am assuming that all potential participants have basic facility with modern computing devices, and can perform the basics of mouse/pointer use, understand interface items like windows and menus, and can find their way around a computer desktop. At the time of this writing, this might be a reasonable assumption, especially given the previous caveat of all participants being university students. However, future versions of this study might consider people of an age variance where this is not a reasonable assumption.

Data Collection

I used multiple methods to collect data from my participants, because as Greene explains (2007), different aspects of data complement each other when different methods are applied in a single study. This study combined interviews, observation, and a concurrently analyzed think-aloud protocol.

After the participants were selected, I conducted interviews with each of them. The interviews were loosely structured, with the goal of gaining access to each participant's unobservable past (Weiss, 1994). The interviews confirmed the participants' lack of experience with professional video editing software, and also gathered their stories about their relationship with computers and software, their comfort level with technology, and what kinds of prior activities they have had with other computer technology and software in their lives. As the conversations evolved in the interviews, there were further follow-up questions and discussions about other activities and hobbies that the participants might have had.

Concerning the validity of interview data, Weiss (1994) indicates that researchers can reasonably assume that we will hear the truth in interviews, but not necessarily the entire truth or the exact truth, and that absent corroboration with other evidence the quality of the interview process itself is a primary safeguard of the validity of the acquired data. Since participants in this study have a reasonable assurance of confidentiality, and the nature of the questions being asked are not likely to incriminate or embarrass the participants or their loved ones, there is high confidence that the collected interview data is reliable. Nevertheless, there is an assumption in this research that the self-reported data in interview are implicitly valid.

After the interview phase of the study, participants began the main phase of the study: a set of tasks to perform on their first time encountering a professional video editing application (the exact design of these tasks, as well of how they evolved through expert validation, is in the following section of this dissertation). As participants performed the tasks, I used several methods to gather data. First, the participants

described what they were attempting to do as they did it, using a concurrently analyzed think-aloud protocol to gain information about the users' thought processes as they discover how the software works (Ericsson and Simon, 1980). The participants' actions on the computer were recorded using a piece of software called Screenflow, which was designed for recording screencasts for tutorials. Since Screenflow can record all action on screen, I have completely synchronized data of exactly what the users were doing, as well as what they were saying to themselves while they did it. Third, while the set of tasks were taking place, I observed the participants nearby, taking extensive field notes of the larger setting that would not have been captured in the recordings. The observation field notes are a central, overarching set of data that serve to connect all of the other information in an attempt to tie it together and make sense of it (Bodgan and Biklen, 2007). Finally, I conducted a second interview with the participants, allowing them to reflect on their experience and how they felt about their performance of the task.

This made a total of thirteen recorded task sessions and observations, along with thirteen interviews, to create a corpus of data to analyze.

Study Design, Expert Validation, and Reliability

The task observation section of the study took place in a lab that was set aside for only this use during the period of the study. It was equipped with an Apple Macintosh computer with 64-bit quad-core processors, 8 gigabytes of RAM, high-speed internal and external storage drives, and a high-definition monitor. The computer used in the study had ten sample video clips in a folder on the desktop. These sample clips, which I designed and animated, had file names such as "one.mov," "two.mov," "three.mov," etc., through "ten.mov." The content of each clip was one minute of an animated numeral,

where the numeral corresponds to the name of the file (so, “one.mov” is a video of a “1,” and so forth) (Figure 3.1).

Figure 3.1 Three sample frames from the assigned video editing task.

In the initial design, participants were given instructions (with no time limit) to first find the sample video clips on the desktop and import them into the video editor; second, edit the video clips into order, where the numbers are in sequence, and each number in the sequence lasts for as many seconds in duration as the number in that clip indicates; and finally, export the resulting video as a file that can play in another application. Though it would be possible to assign video editing tasks using any kind of film footage, either captured specifically for the purposes of this study, or footage from an old television show used to practice editing techniques, this study uses clips of numbers bouncing around the screen to remove any effects of narrative storytelling from the task given to participants. For example, the study could have used footage from a scene of a fight from a T.V. western, as this is common practice for new editors. However, the act of building narrative temporal and spatial continuity in a scene is difficult regardless of the technology being used. Since the participants’ abilities to construct narrative flow is not relevant to the study, the additional irrelevant difficulty would confound the data since it would be difficult to tell whether a participant’s actions or responses were due to the user experience of the software or the challenge of the

narrative. Since counting to ten is certainly within the skill set of college students, using a series of animated numbers isolates the variables that need to be observed.

To maximize the validity of this study, I used expert validation to determine the effectiveness of the specific testing exercises that I would be using. This expert validation was inspired in part by Linstone and Turoff's Delphi method, a process to gather information from experts in a field to reach consensus on a subject (1975). In July 2014, I identified experts in motion picture production and education, presented them with the set of tasks to have been set before the participants in the study, and asked them a set of questions to get their opinions on the exercises.

Since the University Film and Video Association (UFVA) is the professional organization for educators in motion picture fields, its annual conference was an opportunity to meet with experts and conduct expert validation interviews face-to-face. I arranged to interview: C. Melinda Levin, Director of Graduate Studies of the University of North Texas Department of Radio, Television and Film and past-president of UFVA; Jeffrey Warmouth, Professor of Interactive Media and Game Design at Fitchburg State University; and Mary Dalton, Professor of Communication, Film Studies, and Women's and Gender Studies at Wake Forest University. The feedback from that panel of experts indicated that the set of tasks I was planning to use with participants was generally sound and in keeping with the goals of the study, but there were some specific notes to address, including some that I used to refine the design of the study.

One common question that arose among the experts was: how are the participants, when faced with a new software tool and a set of instructions, going to learn how to perform the tasks? This is an unsurprising concern coming from media professors,

particularly professors who (like me) learned how to use our tools in an environment when we would receive tutelage directly from a teacher with a set of manuals. However, as I explained earlier in the methodology section, this study is not directly about education (or rather, training). In this study, the educational environment is backdrop against which this study will be observing effects of UX on sense-making and information-seeking behavior.

Another concern to consider when developing specifics for the study is the participants' familiarity with the software platform. For example, the lab facility for this study will use computers using the Mac OS X operating system. However, many undergraduate students will likely be primarily familiar with the Windows operating system, and this disparity might confound some results. Some interaction with the operating system is unavoidable, but since the software being examined, Adobe Premiere Pro, is a cross-platform application, its design is identical regardless of operating system. This should control for most interactions that might be attributed to unfamiliarity with the operating system outside the environment being studied. However, as this is a study concerned with the experience of novice users approaching complex interfaces for the first time, the experience of a participant attempting to perform a small task with an unfamiliar operating system is not a confounding variable to this study. Nevertheless, the design of the study ensures that processes that could occur outside of Premiere (such as looking for video clips to import) would be limited to tasks that would be generally the same across operating systems.

Given the difficulty of the process being introduced to participants, two of the experts pointed out that attrition might be an issue, and that I should be prepared for one

or more drop-outs from the study. This is an astute observation, and it calls into question what "success" for the participants would look like. If a participant were to give up during the task, that would not be considered a "failure" in the case of this study since it would still yield potentially useful data, particularly about what was happening at the moment of quitting. For this study, while "success" for the participants would ideally be to complete the tasks, giving up is a completely viable outcome. "Success" in terms of the study would be if the participant remains in the study until the end regardless of outcome. In the event of a participant not completing a task, it would be important to communicate with them about the events in the study and explain to them that their participation is still valued. This in fact occurred three times in the study, as will be described in following sections.

In the largest concern brought up during expert validation, two of the experts suggested that I take particular care with using leading language in the instructions given during the study. For example, I had originally intended to instruct the participants to "import the video clips." However, one expert pointed out that "that's an industry term. You're making an assumption" and that phrasing the instruction in that way would only make sense to participants with certain software experience. Referring to the same instruction, another expert pointed out that the term "'import' is a trigger, a specialized knowledge word" that would telegraph to the participants exactly what the process would be for them to bring the video into the software in order to work. In fact, a term such as "export," which might seem like a generic enough term, has domain specificity in this software and has changed over the years. In the first early versions of Adobe Premiere, if a user wanted to take their editing work and create a new video from it, they would

choose a command in the menu bar called “print to video,” in an effort to make it clear that what the user wanted was just like printing out a paper in a word processor. The “print to video” command disappeared in subsequent versions, along with the decline of paper-centric writing workflows. Every command is laden with meaning, so instructions must be scrubbed of domain specificity or hints.

Taking this advice into account, I examined the language I had intended to use to instruct the participants and found that it had other specialized knowledge words that precisely signaled the process that the participants should use, including “export” and “cut.” It also seemed possible that these specialty words might confuse the participants with language, forcing them to spend cognitive energy deciphering the meaning of my instructions rather than executing them with the software.

With this advice from the expert consultants, I set out to rewrite the participant instructions in such a way that would neither excessively steer participants toward workflows or solutions that this study is designed to observe, nor confuse the participants at the moment that they are receiving the instructions. When trying to simplify the language as much as possible and remove technical jargon, I was inspired by web cartoonist Randall Munroe’s comic “Up Goer Five,” which presents a detailed diagram of the Saturn V moon rocket that is richly annotated only using the one thousand most common words in the English language (2013). This comic’s popularity led to the publication of Munroe’s 2015 book, *Thing Explainer*, which expands this concept of hyper-simplified language to articulate complex systems such as human cells (“tiny bags of water you’re made of”) and nuclear power plants (“heavy metal power building”). Though *Thing Explainer* and “Up Goer Five” are works of humor, Munroe effectively

simplifies and communicates the essence of these systems using only the most frequently used "ten hundred" words. My goal, then, was to rewrite the participants' instructions for this study in such a way that abided by the "Up Goer Five" restriction of using only the thousand most common English words (plus one: the word "Premiere," which was necessary to be able to direct participants to use the software central to the study). Using a piece of software called the "Up-Goer Five Text Editor" (Sanderson 2013), I was able to identify the offending words in the instructions, which included all specialty words as well as many other words and phrases that also needed to be recomposed. For example, "import the video clips into Premiere" became "bring the movies into Premiere," "edit the video clips" became "put all of the movies together," and "export the video you have created as a file that can play in another application" became "turn the movies you put together into one longer movie that can play without Premiere."

As a further precaution of reliability, I performed code reliability checks during the data analysis process. It was not practical for the results of this study to be sent to multiple coders, then to confirm a Cohen's (1960) kappa-value to across the array of coders. Barbour (2001) points out that a team of coders is not necessary to ensure rigor in qualitative data, so long as a methodical, systematic process is in place to make the analysis transparent (p. 1116). For this reliability check, I sent samples of coded data to the same set of experts who analyzed and gave critique on my original study design. These samples contained interview transcripts from users at varying levels of accomplishment in the study (to be analyzed in subsequent chapters) with codes that marked evidence such as different levels of past experience with computing and video, and methods of figuring out problems with software. Each member of the expert

validation team indicated that both the coding terms and where in the data they were applied seemed accurate and justified.

As a further safeguard to ensure that these data and results are reliable, I have designed the study to triangulate the data collected by conducting two interviews, recording think-aloud sessions, and conducting simultaneous observations. As noted by Fielding and Fielding, data triangulation is another useful way to increase the validity of research findings, though triangulation cannot guarantee a study's validity (1986). However, by combining triangulation with expert validation, and code reliability tests, it is more likely that the data gathered will be trustworthy.

Risks and Benefits

Any potential risks of this study are far outweighed by potential benefits. One important direct benefit to the participants of the study may be an early self-appraisal of their own skill level, which may serve as encouragement to engage in regular practice. Given that the participants will be selected for both their interest in the field and their lack of prior experience, it would be particularly important for those participants to practice their skills regularly regardless of their performance in the study. Another direct benefit would be the opportunity to develop rapport with the researcher who happens to be their teacher. It is well-established that high-quality interviews need a rapport between the researcher and participants (Denzin & Lincoln 2011, and Glesne 1999), and it is also known that in higher education a developed rapport between student and teacher is beneficial to a successful student learning experience (Benson et al 2005; Grantiz et al 2009). The long-term indirect benefits of the study will hopefully be better software user

experience design in future versions of media software, and better incorporation of self-directed software skill acquisition into classroom instruction.

I anticipate a potential risk of the study, related to a potential ethical issue noted below. Because the study is action research, and the participants were in a large class, confidentiality may be somewhat porous as participants might talk about their experiences with their classmates, and classmates may be able to then figure out details about the participants. While it may not seem particularly important on the surface that one student has no experience in an area, or that they have interests in a certain career, or that they are even participating in a study, this might put a participant in an awkward situation in the class or in the major. Non-participant students may wonder why participants are getting attention outside of class or developing a rapport with the teacher. These situations are speculative, but nevertheless it is critically important to monitor and maintain confidentiality to protect the privacy of the participants who are reporting aspects of their lives and their pasts. During the duration of the study and for the remainder of the semester in which the study took place I took care not to refer to the study to other students or in class, while regularly encouraging all students to make appointments with me and visit my office hours to get additional help or advice, and to give additional opportunities to cultivate rapport.

Ethical Issues

When considering ethical issues of this study, I turn to questions posed by Tuhiwai Smith that point to blind spots which privilege can create (1999). In particular, for this study I refer to her question: "for whom is this relevant? Who says so?" (p. 173). I hold the perspective that this study is relevant for all users and designers of professional

software, as well as all those who use such software in classroom instruction both as teachers and learners. However, this perspective doesn't give much to current students in the class. Participants who put up with the intrusion of the study received additional focused attention and structured practice (expanded below in the "Risks and Benefits" section, but the students in the course who were not participants were left out. They did not get the possible benefit of greater interaction and rapport with the instructor that the participants received, nor will they reap the long-term rewards of possible improvements in future course designs. This is not a treatable issue, as the nature of the study requires exclusion of most of the students who are enrolled in the course. Further, since any benefits from this study will hopefully be applicable to the entire community of users of complex software systems, the ethical choice of giving a few participants a slight possible benefit in the short term weighed against possible benefits for a large number of people in the future seems to be a clear one.

Limitations

Since this study is designed to look at one specific subset of students in one specific class as they begin to use one specific software package, there are limitations to this study. Because I am only studying people who do not have prior experience with professional video editing software, the results do not speak to those who may have partial or cursory knowledge of these skills, or people who have basic skills and are trying to improve them. The study also only looks as people who have self-declared interest in going into motion picture career fields, so results could not be extrapolated to those who are picking up the skill incidentally, or only as a course requirement. The participants are all undergraduate students, so we will not know how the results might

apply to graduate students or students in middle and high school. All of those, however, may be candidates for further study.

Study Significance

My particular interest in this study is rooted in two interrelated desires. By examining the relationships between UX and learning as they relate to professional media software and processes, I hope that this study contributes to new understanding of user experience, specifically to improve how these interfaces are designed, and how future software can be developed. Parallel to that, new understanding about how users approach self-directed learning of software as a function of user experience might improve how we incorporate that technology into professional media courses and curricula.

CHAPTER 4

ANALYSIS

This chapter analyzes the qualitative data collected in this study. After an overview of the participants, I will summarize themes and patterns that emerged from the interviews and tasks, and describe categories of performance, which several subgroups of participants shared.

Participants and Process

Thirteen participants were selected for this study and chose pseudonyms for themselves (or were given pseudonyms if they opted not to, or if their chosen pseudonym made them identifiable), and those pseudonyms will be used to indicate different participants in this dissertation. The interviews were coded using a thematic analysis to inductively identify themes and patterns that were prevalent in the data (Braun & Clarke, 2006).

The majority of the participants (10; 78%) identified themselves as female and 3 participants (22%) identified themselves as male. Most participants were white (11; 85%), 2 participants (5% were African-American. One participant (7%) was over 50 years of age, the rest were between 18 and 25 years old. All participants confirmed that they had never before used professional motion picture editing software (this was a fundamental criterion for selection). At the time of selection, all participants had reported that they had not edited motion pictures before, and did not identify any software that they might have had experience using.

For the first part of the study, I interviewed the participants with open ended questions to gather their stories about their prior experience and comfort level with computers and software, following up with discussions about other activities, interests, and hobbies in their lives.

While all the participants indicated a specific interest in using motion picture editing in the future, only six participants (46%) are studying in a media or media-related field. Two participants (15%) are majoring in computer science, and five participants (39%) are studying other non-media-related fields such as biology, finance, English, or fine arts.

During the interviews, four participants (31%) indicated that they had in fact used a consumer based video editing program such as Apple iMovie or Windows Movie Maker, even though they had not reported doing so prior to participating. All of these participants, three women and one African-American man, had actually made videos in their past and used basic consumer-based editing programs, but in the interviews minimized the validity or even realness of their past experience. When asked if he had ever made videos of any kind, a participant pseudo-named Witzel said, "not quite… I had used Windows Movie Maker, really just throwing a stupid filter on it… so I've never really edited for real." After asking me "what do you mean by making a video," participant Mary conceded that by recording herself cheerleading, using basic consumer editing program to add basic transitions, and sending an athletic tryout video to get a scholarship might have been "a little bit" of making a video. Participant Inga posted videos of her dog having puppies to YouTube, and later made some animated videos by taking many individual images of clay creatures and putting them together with Windows

Movie Maker, but insisted that "to me, that's not really making videos." Despite the initial intended criterion of only inviting participants with no video experience at all, these participants were retained in the study for the purposes of observing the possible relationship of this experience to their performance in the study. Several trends arose during the interviews that did not become clear until after the main task section of the study was completed and analyzed. Those patterns are discussed later in this chapter.

After all the initial interviews were completed, participants began the task phase of the study. This took place in a lab at the University of South Carolina that was set up with a customized workstation that was set up to conduct the study, set apart from other labs to avoid distraction. Participants underwent this part of the study one person at a time, with no time limit. Participants had been briefed during the interview that they would be asked to do a set of video editing tasks using Adobe Premiere Pro, but were not told what the task would be. Upon starting the task, each participant was given printed instructions that said:

> There are ten movies on the computer, they are named "one.mov" through "ten.mov". Each movie shows a number that is the same as the name of the movie.
>
> 1. Bring the movies into Premiere.
> 2. Using Premiere, put all of the movies together in a way where: (a) the numbers in the movies are in order (one through ten), and (b) each number lasts for as many seconds as the number shown (one.mov is one second, two.mov is two seconds, etc.).

3. Finally, turn the movies you put together into one longer movie that can play without Premiere.

Participants were instructed to use a think-aloud protocol and to describe what they were doing as they did it, while they were simultaneously observed and their screen actions were recorded along with their voices.

Three participants did decide to give up on the exercise during the study, providing valuable data about what might have been going on leading up to the decision to stop. In those cases, I communicated with the participants to determine that they were willing to remain in the study even without completing the task, both to allow the use of data gathered up to that point, and to continue with the follow-up interview part of the process.

These three participants who were not able to fully complete the assigned tasks comprise one of three distinct groups that emerged from the study, and this analysis will next describe the groups as well as the traits and behaviors that unify and separate them, leading to discussions of what these clusters of behavior might imply. The three groups that emerged in the study are: the Aware Group, the Average Group, and Abandon Group.

The following three sections detail the characteristics shared by participants of these respective groups. The participants in the Aware and Average Groups all completed the assigned tasks; the two groups share many characteristics and in many cases only differ by degrees. I will first analyze the trends exhibited in the Aware Group along with that group's unique characteristic, then highlight the similarities and incremental differences of the Average Group. Participants in the smaller Abandon Group did not

complete the task, and exhibited a distinct and different set of characteristics, and I will analyze those last.

The Aware Group

Five participants (38.5%) were classified into the Aware Group. Several traits emerged among these users as they worked in the lab on the assigned task that led to this classification.

The most obvious marker for participants in this group was overall speed. The participants in this group took between 15 and 30 minutes to complete the entire video editing task, generally faster than the other users, even though there was no time limit or stated advantage or reward for performing the tasks more quickly. Speed of completion at first might seem like an independent criterion to consider the groups of participants separately, but it appears to be an accumulation of several other interconnecting factors that participants exhibited as strategies to make this new software work, and to leverage the software to improve their own performance. These accumulated factors have an effect on the overall time it takes for the user to complete the set of tasks. Not all of the participants in the Aware group exhibited all of these factors, but each of them displayed most of these traits, examples of which appear here.

Aware Group participants were consistently *iterative*; they tried many more different approaches to the subtasks than did the other participants. Even when those attempts were not successful, Aware Group members would jettison the failed attempts quickly, revise their approach, and move onto the next iterative attempt to solve the problem. There are several redundant methods users can use to change the duration of clips, and when Leafia was met with difficulty when resizing them by dragging the length

of the video with her mouse, after two imprecise tries, she changed methods and found a way to enter durations in by typing time values. Inga completed the entire sequence, then went back and changed the clips in the sequence to more accurately reflect what the task list instructed (Figure 4.1).

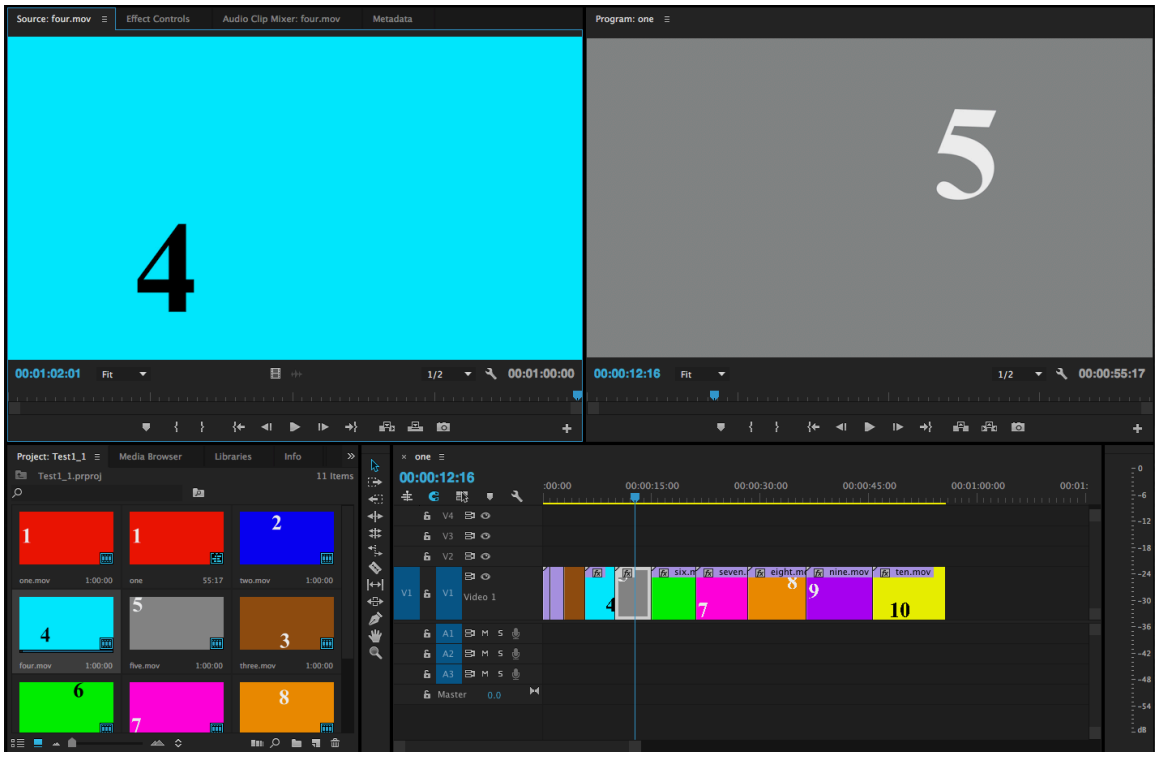

Figure 4.1 An example of a complete task from a participant in the Aware Group. Adobe® product screenshot reprinted with permission from Adobe Systems Incorporated.

Witzel completed the entire sequence, only to find that when reading the time measurements, he had been mistaking minutes for seconds and had in advertently created a fifty-five minute video of animated numbers by stretching the duration of the videos to an extreme length. Catching his error, he went back and was able to quickly redo the sequence to the proper duration. Stephen similarly found that while putting together the entire sequence, he realized that he had mistakenly thought that time in video was

measured in decimal fractions of a second, rather than in frames (increments of 1/24th of a second), so he went back and made adjustments in his timeline durations, making his sequence more precise.

Building upon the iterative approaches, participants in this group were *exploratory*, making incremental changes in their techniques as they worked. After establishing that they understood the process, Keegan began looking for keyboard shortcuts to streamline her work, speeding up a slow, mouse-only process with a fast set of keypresses. Stephen quickly adopted a two-hand navigation process, using one hand on the keyboard and one hand on the mouse to operate the software with more efficiency than primarily using a mouse to do all the operations. Inga found that she could modify the interface and zoom in to the timeline to be able to manipulate it more precisely. Since the task involved building a sequence of ten video clips, many operations in the task had to be completed ten times, so after viewing the pattern of operations, Leafia looked for ways to condense steps and perform operations in all the remaining simultaneously. To be sure, not all of the attempts at improvement were successful, but participants in this group shared a tendency to look for these improvements soon after beginning work, even though there was no reward or advantage to completing the task quickly.

Aware Group participants reasoned about their process deductively, but honed that process by *filtering* out information that they deemed to be less important in the moment, prioritizing moving forward imperfectly over perfecting details at the outset. When facing difficulty in trimming clips to the needed length, Keegan said, “I’ll figure that out later,” and moved on, returning to that sub-task after getting more comfortable with the software. In nearly identical events, Inga said “is that supposed to happen?

Maybe? I'm just going to let it happen," while Stephen said "I don't know what that means," but they both would move on to what they planned as their next steps without slowing down to figure out something that seemed irrelevant to the task at hand.

Participants in the Aware Group were successful at *contextualizing* the feedback they received from the software's interface, even when those changes might have been small. This contextualizing might have been as simple as seeing a message in plain view when it became clear that the message applied to the subtask at hand, as Keegan did when she identified that the Premiere interface itself directed users to "drag clips to timeline to begin." Noticing small changes seems to indicate a sensitivity to changes in the state of the interface and thus the application, such as when Inga quickly noticed that the shape of the mouse cursor changed when it was in proximity to different elements of the interface, indicating that a new operation was available, allowing her to manipulate the durations of the clips in the timeline more simply than by entering numerical values. Contextualization also happened when users recognized a situation that was familiar in their experience, such as when Leafia observed and noted a similarity to a consumer-level application they had used.

Only one of the participants in this group sought external help, even though all were informed at the outset that there was a high-speed internet connection and web browser available on the testing machine. When Keegan had difficulty changing the duration of clips with sufficient precision she proclaimed, "I'm tempted to click around here, but I'm not going to do that because 'help' [the help menu] is my friend." But after seeing no applicable results from her search, in about thirty seconds she said "that helped

less than I thought it would," and went on to find an effective method to alter clip duration within a few minutes.

All of the participants classified into the Aware Group shared a behavior during that task that was not present in any participant in the other groups. Taking the advice of the expert validation in Chapter 3, all specialized domain-specific language was expunged from the instructions of this task. Simple, but loaded terms such as "cut," "edit," were removed. The experts were specifically concerned about the use of even more leading language, "import" and "export," terms used in professional parlance that are also used explicitly as direct commands in Premiere, so those triggers were also replaced with neutrally descriptive, non-domain specific terms such as "bring" and "put."

Each of the five participants in the Aware Group immediately deduced the domain-specific operation "export media" during the final stage of the task.

There is no other way around the "export media" command; there are no shortcuts, no tricks to substitute, no common computing trope that would have the same effect. Other potentially domain-specific concepts like "edit" or "import" can be accomplished in other ways; users can "edit" or "cut" by simply dragging items close to each other or by using a tool with a razor blade icon, the concept of "import" can be successfully circumvented by simply dragging clips directly into a window, an action that is shared with the basic techniques of using a modern computer operating system.

But the only way to complete the full task is to use the "export media" command. While this step at first eluded every other participant in predictable ways (to be described in detail in subsequent sections), these five participants, who had never used professional video editing software before, successfully used this domain-specific command on the

first try in way that resembled intuition. This possible intuition is examined in more detail later in this chapter and into the next.

Investigating details that emerged during interviews, other trends emerged with users in this group. One common thread that emerged among every member of this group was that they each demonstrated intensive self-directed practice at a specific independent pursuit or hobby. Participant Inga has been playing World of Warcraft since she discovered it at the age of eleven, eventually founding and managing a guild of over five hundred players, and learning to build her own computer so it would be powerful enough for her needs as a leader in the international role playing game. Stephen taught himself to program the C++ language in his early teens, then developed an obsession for motorcycles, doing all of his own maintenance and racing his bikes regularly. Keegan focuses a significant amount of energy on finding new work; she regularly searches for jobs, even though as an early-year undergraduate she does not yet have enough work under her belts to get that work. Leafia has been writing competitive fan-fiction set in the Harry Potter and Pokémon universes since she was ten years old. While a biology major and chemistry minor, Witzel is an accomplished percussionist, playing in everything from school orchestras to rock bands since he was in elementary school.

During the interviews prior to the software task, participants were asked to describe their processes of working through frustration when using computers, and to retell an account of when they successfully worked through a problem they might have had while using a computer. Participants in the Advance Group consistently described these events as solitary attempts to figure out the problem they were solving by trying to work with the software at hand. None reported searching for online tutorials and internet

forms, getting help from a teacher or peer, or any other external assistance as part of their regular approach. This manner of advanced trial-and-error to deduce solutions within the confines of the interface at hand is somewhat in keeping with the behavior these participants exhibited during the study where participants might quickly jump to an external resource and then return to their regular process, but external resources are not central to the users' primary problem solving process.

The Average Group

Five participants (38.5%) were classified into the Average Group. All of the participants in this group successfully completed the entire video editing task. They took between 25 and 50 minutes to complete the entire video editing task, generally longer than the Aware Group.

Participants in the Average Group exhibited some similar factors (iteration, exploration, filtering, contextualizing) to the Aware Group, but with variations that contributed to them being less effective than their counterparts in the Aware Group, or they might not apply these factors as consistently or comprehensively as Aware Group members. These participants might only exhibit one of the characteristics, rather than three or four in the Aware Group. Sometimes gains in effectiveness or efficiency in one area was mitigated by deficiencies in another.

Ninjabear would explore to try and find a batch operation to speed up her work, but would also repeat unnecessary operations such as editing clips into sequence, deleting them, and then re-adding the same ones several times (a kind of anti-iteration). Maria would filter out details for expediency, saying "let's just leave it that way," but would not return to that detail to make it more precise later, saying "I think that's about as good as

it's going to get." Rey observed a contextual change in the interface when the focus of the active application shifted and said "it used to say Premiere, and now it doesn't," stopping work at that point to avoid causing a problem with her sequence, but she didn't notice the differences in the icons that would have signaled to her that she left out one of the clips in the sequence.

While all of the participants in the Aware Group intuited the domain-specific "export media" command on the first try, none of the participants in the Average Group did. During the final stage of the video editing task, each of the participants in this group first tried another method to turn their sequence into a self-contained video, most commonly to try and use the "save" or the "save as" command, like one would expect to with a word processor to create a new version of a written document. However, "saving" in the context of a professional video editing program simply creates a new Premiere file, not a new video. Video editing applications do not directly manipulate the media at hand like a word processor or a graphics editor. Since video files are so large and cumbersome, video editing software works "non-destructively," acting as a database that keeps track of a list of changes to display. So, a Premiere file does not actually contain any video, it contains a list of information that tells Premiere what parts of which videos to play at a given time.

In interviews, trends emerged that seemed to parallel those in the previous (Aware) group. Most of these participants reported having interests and hobbies, as one might expect any normal person. All of these interests involved organized sports (these participants were lacrosse players, swimmers, and cheerleaders), or school groups and clubs (chorus and art clubs). However, none reported anything approaching the kind of

strong self-directed pursuits that were the hallmark of conversations with members of the Aware Group.

Prior to undergoing the task, the participants described their typical work process in the face of an obstacle while using computers. All but one of the participants in this group identified external resources—such as internet searches, tutorial videos, and getting help from more experiences friends—as an important method they use to solve their problems. However, only two of the participants used that process in the video editing task. One of the participants, Zoe, searched for several minutes for YouTube videos that matched her problem before finding one that "looks exactly like what my problem is supposed to be," but she had not self-identified that as part of her usual process.

The Abandon Group

Three participants (23%) the subgroup classified as the Abandon Group, all of whom gave up the task after 60-90 minutes. This section will examine the cases of those three participants—Mackenzie, Mona, and Mack—in greater detail.

Mackenzie is a twenty-five year old African-American woman from the San Francisco bay area who changed her focus from journalism to the arts after taking time away from college to assist in family caregiving to take care of her baby daughter and sick mother. Her parents, now successful, never went to college and didn't want their children's success to be limited by anything, including their geography, so they "kind of required" Mackenzie and her five siblings to go to college away from home. They support her decision to pursue the media arts as a program of study, but they insist that she have a concrete plan and pursue it diligently. The structure that her parents encourage extends into extracurricular pursuits as well—Mackenzie played piano and clarinet, and

participated in clarinet and cheerleading. Her parents required her and her siblings to play an instrument and a sport, and she changed her instrument and sport one time each. “We kept busy. Nothing was ever by force,” she explained, “they let us get the feel of things and go with it, but at a certain point we had to stick with it. You can’t just flop around forever.”

While Mackenzie might technically be a returning adult student, she is not that much older than her traditional-aged peers, and experience with computers growing up and in school is not very different from slightly younger students. There was a shared computer in her family den that she would use to play some educational games, but “not for anything super drastic.” There were also computer labs in her middle and high school, which she mostly used for typing papers for classes rather than writing them by hand, and using some basic research. But there was no advantage for her to use the computers at school, since she “was using computers at school and at home the same way.” Once working in the media arts in college, she had one team based project involving the creation of a video, but their method of dividing the work entailed assigning work to the team members who already felt comfortable with the task. Mackenzie felt more comfortable with the camera, and another teammate handled everything to do with post-production: all the computer-based work.

When Mackenzie was engaged in the assigned task for this study, she started out without incident. It only took her three minutes to create a new project and import the clips into Premiere. By the time she was twelve minutes into the task, she had built the sequence of clips in the required order, and each clip was within a few frames of the durations indicated in the instructions. At this point, she began to struggle with the final

instruction: “turn the movies you put together into one longer movie that can play without Premiere.”

The solution was to choose the command under the “file” menu that indicated “export,” and select the submenu item “media” while the item to be exported was selected (Figure 4.2). After that, a window appears with more technical options to choose, any of which would successfully complete the task if selected, because in so choosing the software would create a new self-contained video file of the users’ newly created sequence that would play without Premiere, once the user finally committed to the act by pressing the button in the lower right hand corner marked “export.”

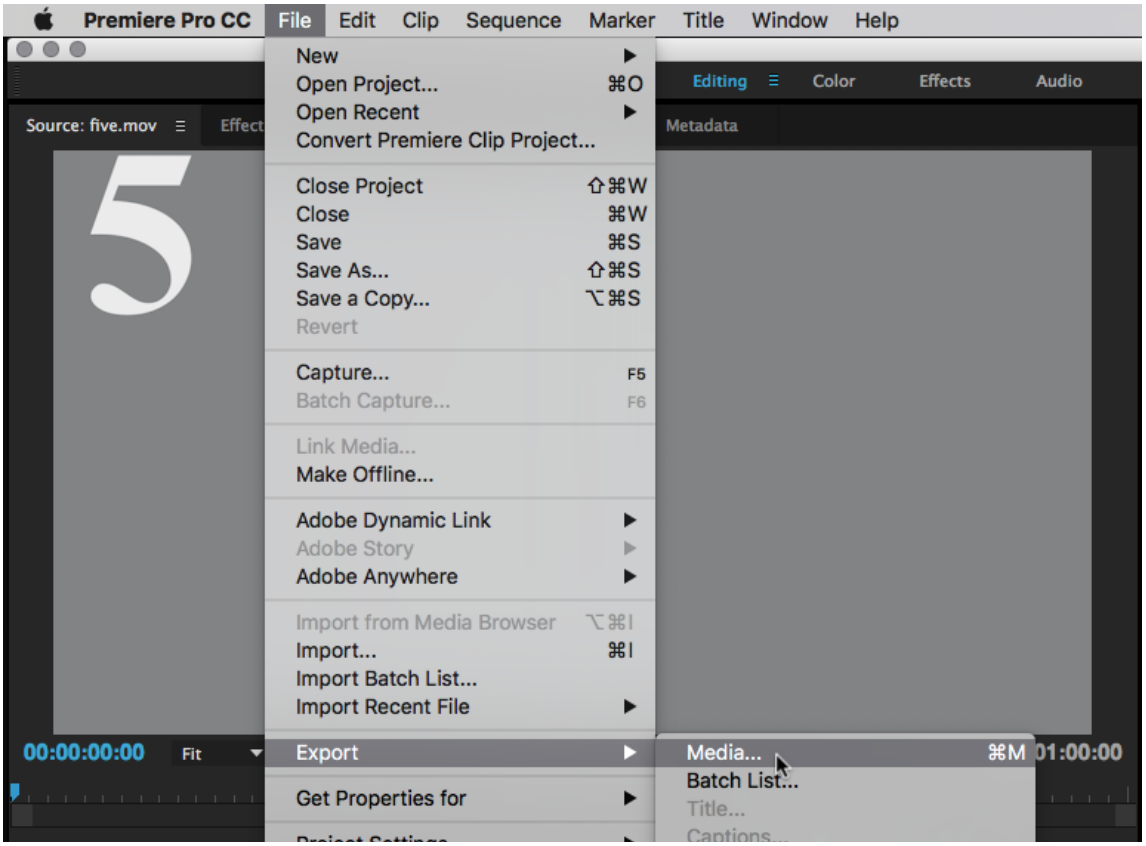

Figure 4.2 The “export media” command. Adobe® product screenshot reprinted with permission from Adobe Systems Incorporated.

For the next hour Mackenzie tried to accomplish this part of the task. In keeping with the actions of members of the Average Group, she tried using the commands "save" and "save as," common in nearly every software application, including the business computing that she had used in high school, to create a new file.

The first twenty minutes of her struggle entailed trying to find a way to play this file, an abstraction which contained no video, with another program. She said "I think this is the part that has me stumped because I'm not sure where to play it," indicating that she believed at this point that she had completed the task correctly, only couldn't find a way to play the file. During her search, she double-clicked on the animated source clips, which opened them and played them back in a utility program called "Quicktime Player." This could have served as a hint that this is indeed the program she should use to play back a video outside of Premiere, and that the file she had created was therefore not a playable item.

Over the next fifteen minutes, she began trying unrelated commands. In Premiere, she would scan through the menus and choose items seemingly at random (like "create Photoshop file"), look at the resulting dialog box, and the cancel the command. On the desktop of the computer's operating system, she would right-click on her newly created file and choose options from that menu like "compress," which would make a new copy of the file that was in a ZIP format that might be more suitable for emailing, but then she would try to play that file, unsuccessfully since it is still not a video. She started choosing more obscure commands in Premiere, such as "render and replace," which took each individual video clip in the sequence, created a new video copy of it, and placed it in her sequence. These new copies of the clips showed up on the desktop, and could have given

a hint about the kind of file she needed to make. But instead, she deleted the newly created files from the desktop, which made Premiere think that her videos were missing, and her sequence then displayed nothing but a dramatic warning message: “media offline.” Ironically, it is at this point where Mackenzie finally tried the “export” command, and actually created a new video and opened it in Quicktime Player, but it was only a video of the “media offline” warning (Figure 4.3).

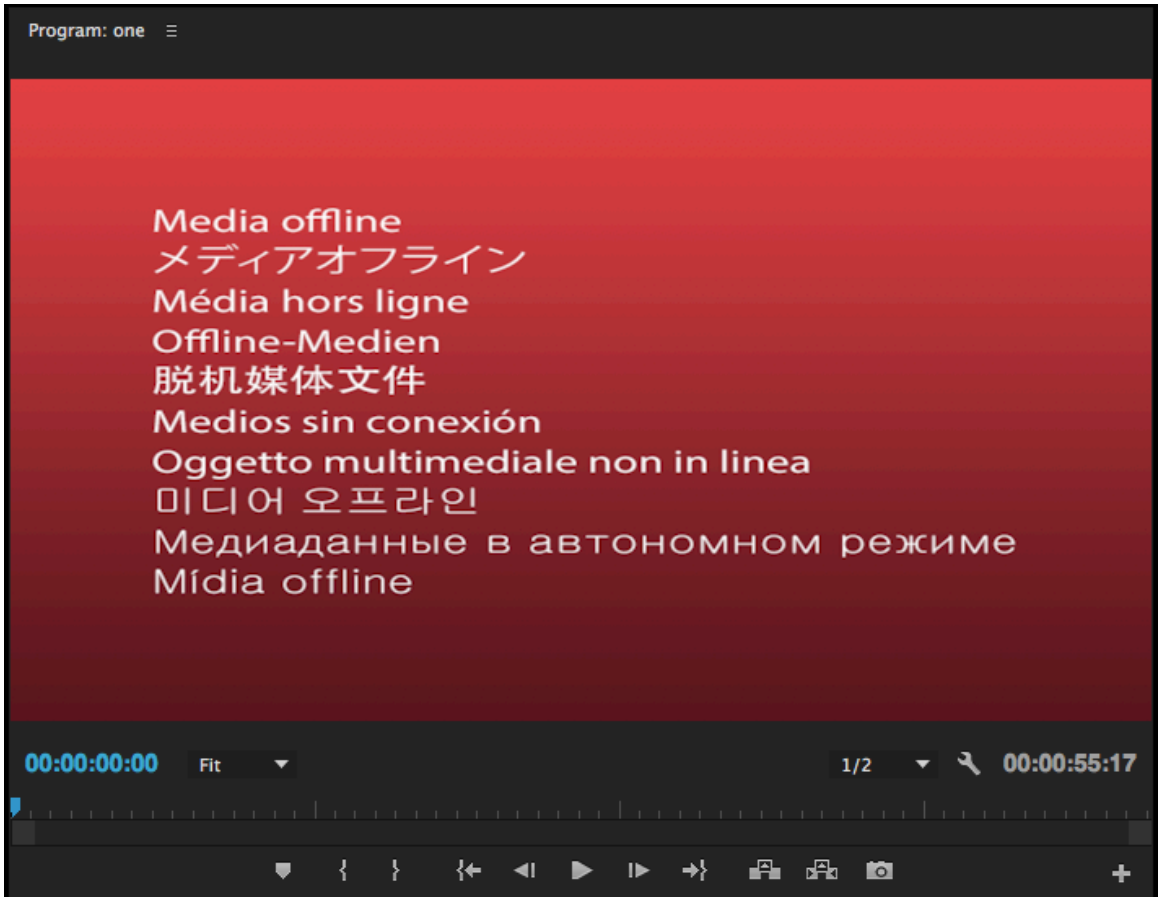

Figure 4.3 The “media offline” warning. Adobe® product screenshot reprinted with permission from Adobe Systems Incorporated.

Fifty-five minutes into the task, Mackenzie unsuccessfully tried to revert to her older version of her edited sequence, but did eventually manage to reconstitute her

correct sequence. At this point she wondered, “Maybe before I save it as something I have to make it something else. But how?” She began attempting another battery of seemingly random attempts: dragging different items in the program onto different windows and then choosing the “undo” command when it damaged her previous work. She chose the same “save as” command several more time, but then cancelled before committing to the original error she’d made. She compressed files on the desktop again, she moved things into folders and then moved them back.

At this point, when she returned to Premiere, she had selected her completed sequence, chose the “export media” command, and looked at the window that would have completed the task. But rather than choosing the button that said “export,” she chose the button a few dozen pixels to the left that said “cancel.” Then after a few more random actions, she declared that she was stumped and quit the task, a few minutes after she had unknowingly almost succeeded.

During the debriefing interview, Mackenzie described the experience as “almost like reading another language.” When asked why she never used the help menu or looked online for assistance when she was stuck, she said “I thought about it… I felt like it would be a copout.”

Mona is a fifty-six-year-old Caucasian South Carolina native who is returning to college to get a second bachelor’s degree now that she and her husband have successfully put their three children through college. She received a degree in English in the early 1980s, and spent the intervening years in administrative assistant jobs while she and her husband raised their children. While she was in college the first time, computing was a new and novel endeavor, and she recalls a professor recommending that students “get in

on the ground floor… but I was like 'I don't want to do that, but it's interesting.'" As Mona moved through the work force, computers arrived as general purpose machines and while she thought they "made no sense" she could "read the directions" to do the kinds of office computing that the other participants in the study might have encountered beginning in elementary school.

As Mona's children grew, her son in particular became her technological go-to person, setting up the music on her iPod and generally giving technical support. When facing computer-based problems on her own, she describes herself as "very patient," reporting that she had stayed up all night recently to try and complete a project, though it had been unsuccessful.

She has "one million hobbies," including travel, making trips to Europe several times a year, often to visit one of her daughters studying in Paris. Her husband, as physician, got her interested in cycling when they were dating in college, and they would cycle around the European countryside and watch the Tour de France.

Mona's attempt at the assigned task is perhaps most remarkable in the sense that over the course of nearly an hour, she was only able to complete the first item on the list—importing the video files on the desktop into Premiere—but without knowing she had done this. During the first twelve minutes of her work, she did not interact with Premiere at all, opening and examining the videos on the desktop of the computer instead. After she opened and played the first two videos, she performed a few operations in the Quicktime Player seemingly chosen at random, such as saving copies of the files on the desktop and then deleting most of them, and making a "new movie recording," which had she not cancelled the operation would have recorded her screen in a basic way

similar to what I recorded for this study. Eventually, she moved all the video files around on the virtual desktop, placing them in a graphic physical order there according to the numerical sequence. This act, though it fundamentally had nothing whatsoever to do with the task at hand, might have been the closest that she came to accomplishing any part of the task.

Eventually, Mona performed a file search to find Premiere and launched it, then closed the main application window, hiding it from view. Each time she wanted to bring the application to the foreground, she would perform this same file search and double click on the icon she found as though it had never been running, even though it was already active. For the next fifteen minutes, she clicked on every different visible tab in every window in Premiere, without lingering on it long enough to observe what was in the window or different from what was previously visible before clicking on the next tab. Periodically, she would close the main application window, movie files around on the desktop, put them into a new folder, or make another copy of them, then return to Premiere and continue clicking on tabs.

Twenty-seven minutes into the task, she began looking through menu items, and found the "import" command as one of the options. She used the following dialog box to select the folder of clips that she had made on the desktop and successfully imported them into Premiere. However, because she had clicked on so many different tabs in the interface, she had hidden the tabbed window from view that would have shown her that the files were successfully imported (this tabbed window is open by default when the program starts, and when it is empty it displays the message "import media to start.)"

A few minutes later, Mona clicked on enough window tabs to nearly return Premiere's interface to how it would appear when it begins, and she was able to see successfully important footage. In the tabbed window next to that footage was the empty workspace where users place video clips in sequence, and in that empty workspace displayed the message "drop media here to create sequence." She never did follow that instruction, but rather closed the project and selected not to save the project before closing, losing the fact that she had imported the media.

Mona started again, this time importing the clips one by one into a folder in Premiere over the next ten minutes. Next, she used the "new sequence" command in the menu to create several new sequences containing one clip per sequence. Rather than have a sequence with all the clips in it, she now had many sequences, each with a single clip in it, effectively making another copy of the footage. She repeated this several times until the forty-seven-minute mark, when she closed the project, and again specifically cancelled the option to save before doing so. She started again, this time trying to use the "new sequence" command, but without having imported the footage beforehand. This continued for another seven minutes before she said "I did nothing," and stepped away from the workstation.

During the second interview, she explained that she felt like she knew what she had to do, but only couldn't figure out how to do it. She went into great detail that she thought she was trying to re-orient all the windows in the software in order to make everything visible at once. Knowing that this was not a path to a solution, I continued asking what her idea was to continue if she'd been able to accomplish that, and she replied that she thought that if her son had been there he could have done it in minutes,

but instead that she thought "if I do it that way then if a eureka moment comes at least I'll be ready." When asked why she had not used Premiere's help menu or other resources that she could have accessed online, she replied "I usually don't find what I'm looking for, I don't find answers."

Mack is a twenty-four year old Caucasian man from South Carolina who returned to school after a two-year break in his studies. He was originally a criminal justice major, but was placed on academic probation due to grade deficiency. After his hiatus he returned to studies, but his grades were so low that he was restricted from most majors. Since the media arts program does not have a GPA threshold to be permitted to take courses in the program, it was one of the few programs where Mack was able to take classes. He found the subject matter interesting, and would like to go into broadcasting, but those majors are also restricted, so he is taking courses in the art program in hopes of bolstering his GPA to be able to switch to a more restrictive major.

Sport is the most important hobby, or maybe the most important thing in Mack's life. Both he and his brother went to a private school with a tuition several times that of the university he attends, but he left private school in eleventh grade for public high school to be able to play basketball more competitively. Because most of his private credits fulfilled high school requirements, he didn't have to take many classes in public school and spent most of his time playing basketball. When he eventually went to college he had a chance to play for the university team as a walk-on to show his potential, but quit after a week when faced with waking up at 5 a.m. every day to practice.

He rejects the use of computers as much as possible, he "never liked to be inside," or even watch any TV unless it's sports. He took a typing class in middle school, and that

was the first time he recalls ever using one. “I didn’t really use a computer that much, and all of our papers at the time we would just hand write,” he said, adding that he had never used a camera before the current semester.

Mack’s attempt at the task was the longest of all the participants, at just under an hour and a half. Right away, he tried to “open” rather than “import” the video clips on the desktop of the computer with Premiere. This does not open or import the clips, and the software actually gives no feedback whatsoever. He did not open Premiere or begin importing videos for the first seventeen minutes of his time. That time was spent attempting to manipulate the clips on the desktop of the computer, first dragging the clips into the computer’s menu bar (which has no effect), then right-clicking on each individual video and choosing “compress,” creating ten individual ZIP files that would not be usable. He then moved the un-zipped video clips into a folder deep in the hierarchical directory of the hard drive, in a folder that is only used by Premiere to keep track of preference files that users never directly use. He tried again to open the files with Premiere, with the files in a new location, but the result was similarly unsuccessful. One notable behavior at this point is that once an attempt at something is unsuccessful, Mack always attempted the exact same operation in the exact same way at least four or five more times, to the identical effect.

After seventeen minutes, Mack selected the “new project” command and was greeted by the blank main application window, including the stark messages centered in the two otherwise empty panes that take up the lower third of the screen: “import media to start,” and “drop media here to create sequence.” Rather than use this, he chose another tab and managed to navigate to the obscure folder where he placed the video files. From

here, he was able to work with the files within Premiere, and over the next twenty minutes was able to piece together a sequence according to the instructions.

For the following forty-eight minutes, Mack tried to complete the last sub-task: to make the sequence playable outside of Premiere. He continued the pattern of making an attempt, seeing that it did not work, and then repeating that same set of operations several times. For his first set of attempts, he clicked on a button to save an individual photo from out of the sequence, then viewed that image in a photo viewer only to realize that it was a single still image. Next, he tried the "save as" command to make a new Premiere file, only to find time and again that this file was not a standalone video, then he went back to making still images another eight times.

At this point, Premiere became unstable and began to crash due to "unknown errors," possibly because both the saved project file and all the footage were kept in a settings folder deep in the drive hierarchy that is only used by Premiere to keep track of application settings. Around the one-hour mark, after Premiere crashed several times, Mack found a new copy of his project (again, in the new location deep in settings folder) that Premiere had automatically saved just before one of the many crashes. He opened it, and finding it to be more stable he began to try and figure the problem out again.

Eventually around the one-hour-and-fifteen-minute mark, Mack found the "export media" command. However, each time he attempted to export the sequence, a new error appeared as a dull red window in the corner of the screen that said "a low-level exception occurred." In programming, an "exception" is an error that occurs while a program is running and encounters an unforeseen problem that it does not have a plan to deal with (Spraul 2012). Therefore, a low-level exception is bad news because a rare and

unexpected occurrence is preventing something on the computer from happening at a fundamental level. It isn't provable without proprietary tools that this occurred directly because Mack's actions or how he placed his files, but this was the only time this occurred during this study. After two more identical attempts, Mack said, "I give up. Am I close?"

Like the other participants in this group, Mack did not make any attempts to use the help menu visible in every application, or to search any online resources to try and find a solution to a problem that he faced. He remarked how different this experience was to past times that he used computers, saying "normally on computers I'm just writing papers, I don't really get on the computer that much," then he echoed the exact sentiment that Mackenzie expressed: "it's like a foreign language to me."

During all three of the exit interviews for participants in the Abandon Group, users shared a sentiment that there were points in the task where things felt familiar, like they knew what they were doing. These feelings of familiarity were centered around the earlier parts of the task, dragging and manipulating the clips to put them into Premiere, putting them in order and arranging them. Mackenzie even went as far as to identify intuition at play, describing the initial importing of files into Premiere:

> It wasn't too bad, that felt kind of intuitive, just because I work with MacBooks on my own, so I know that you can just drag things into whatever you need to do. That part felt pretty right, and then once I started seeing it play in the—I don't know what it's called—started seeing it lined up, then it kind of started to make sense. Once I realized that you could shorten [the clips] there, that part kind of flowed and felt a little natural. Kind of using, I don't know if "common sense" is

the word, because that's kind of hard to say in technology, but it was kind of doing what I thought would be most appropriate and make the most sense.

Here Mackenzie identifies the kind of intuition that Herbert Simon attributed to mere recognition (1992), as applied to the parts of the process that she and the other Abandon Group participants were familiar with, namely the aspects of the process that shared at least some similarities to what they already know how to do with computers.

Shared Perceptions

Not only did participants in the Abandon Group identify this sense of familiarity with early aspects of the task that involved moving and manipulating the media clips directly with the mouse, every participant in the Average Group and the Aware Group also specifically identified that part of the task as not being a problem. Four participants independently said that it "made sense" or "made perfect sense." Other participants described that part of the process as "straightforward" or "reasonable." When faced with a problem that resembled what they already knew, even when some of the participants might have stumbled or gotten stuck, they reported that it made sense to them.

The perception that the most difficult part of the task—exporting media—was somehow different was likewise shared by participants regardless of group classification. As reported above, the refrain from the Abandon Group was that it was like encountering a foreign language. The Average Group participants, who each were rebuffed by the software on their first attempts before eventually succeeding, reported frustration as well. Rey said, "It stopped me for a minute," before she went on to complete the task. Even though she completed the task, Mary confessed, "I don't even know what I did right to make that work." Members of the Aware Group reported some frustration as well.

Stephen reported that he "had no idea what was going on" with that part of the process.

When faced with the completely unfamiliar, domain-specific aspect of the task, participants across all groups were frustrated to varying degrees. But participants in one group abandoned the task altogether, another group completed the task after unsuccessfully attempting to solve the problem using ill-fitting techniques that they were familiar with, and one group navigated that part of the task quickly on the first try.

CHAPTER 5

DISCUSSION AND IMPLICATIONS

This chapter will conclude this dissertation, continuing the analysis of the data in relation to the research questions that guided this study, reflecting on the findings in light of the originally stated intent. After presenting models that describe the structure of the collected data, this chapter will discuss potential limitations of the study. Finally, these findings are examined further in the context of considering implications for future research and practice.

The research questions underpinning this study are:

RQ1: In what ways do users teach themselves how to use complex professional software with no prior experience?

RQ2: In what ways does the user experience (UX) of professional software affect the information-seeking processes of users who are actively seeking to learn how to use it?

Expertise, Intuition, and Complex Systems

The first research question tries to gain a better understanding of the novice experience, and how it informs the ways in which new users make sense of highly complex human-computer interactions. This study uses professional video editing software as a prototypical site to examine this question, because of the unique history of such software as an outgrowth of the specialized trade of film editing, and the software's

subsequent evolution into the market for more general use, to the point of being introduced in large general-education art courses at public universities.

This study recruited participants who both have a desire to gain expertise in this domain and also have no prior experience using this type of software. This afforded a unique opportunity to collect and observe data about what happens when completely new users encounter highly-specialized software designed for professional use and what techniques they might have to try and use such software for the first time.

The data from this study showed multifaceted results with respect to this research question. When participants were faced with aspects of professional video editing that resembled computing tasks with which they already had expertise, they were able to perform those tasks with relative ease despite their unfamiliarity with the new software environment. However, when participants were faced with a specialized, domain-specific aspect of a professional video editing task, they responded in three distinct ways.

Participants who exhibited the least experience using computers in any way were not able to complete the new software task. While this is not particularly surprising, another trend emerged with these participants. Whether they described events in their pasts such as leaving uncomfortable work to other team members, or family members directing significant life choices, or generally drifting with no reported reason, those participants were not able to do the work to make new sense of unfamiliar processes. These participants who did not finish the task model information seeking behaviors that prioritize reliance on others for basic facility. They also model information-seeking behaviors that seem to prioritize avoidance strategies or systematic reliance on other people at key expertise-building moments; it is not clear from these data whether they are

specifically avoiding software and digital technologies or if they are more generally averse to unfamiliar experiences.

Some participants were able to complete the domain-specific part of the task, but with some difficulty. They would first try to solve the problem by leveraging techniques they already knew, but calling on familiar techniques was not sufficient to complete the task. Spending more time exploring the software and trying different approaches, these participants were eventually able to discover the process necessary to complete the specialized part of the work. These participants were generally more reactive to the user experience they encountered during the task, as opposed to the more proactive approach exhibited by the more successful Aware group. When sharing information about their pasts, participants in this group also shared a less proactive, less independent style of participating in independent personal pursuits than the higher-achieving participants.

Participants in the third, higher-achieving tier successfully solved the specialized domain-specific aspect of the task immediately on the first try, as though they knew how to do it already, even though none of them had ever faced the problem before. Indeed, one person in this group had an exceptionally high degree of computing experience, but this was not shared among this tier of participants. In fact, most of these participants had the same or less computing experience than participants in the Average Group. One unique characteristic that these participants shared was a high degree of intensive, self-directed, independent practice at a personal pursuit.

These findings raise questions about the nature of expertise and intuition. The members of the Aware Group exhibited behavior that resembled intuition and even situational awareness in a task that required a degree of domain-specific expertise, for

which no specialized signifiers existed to communicate such affordances to the participants. If intuition is, as Herbert Simon indicates, "nothing more and nothing less than recognition" (1994), and Norman's signifiers and affordances (1999) were not in play, then how did such behavior consistently arise? If it were simply a matter of language skill or intelligence, and the ability to deduce the domain specific meaning of the phrase "export media," then why did no other participants leverage their own language skills and intelligence to exhibit the same behavior?

An experienced pilot develops such a level of expertise and familiarity with an aircraft's cockpit instrumentation that they are exceptionally sensitive to changes in that environment. That level of sensitivity and expertise becomes the elusive state of situational awareness. I would extend Herbert Simon's definition of intuition-as-recognition, by way of revisiting Sarter and Woods' classification of situational awareness as a contextually-sensitive awareness and filtering in an information-dense environment (1991), to postulate that situational awareness as currently understood is a kind of highly tuned intuition in a given domain. In this study, participants in a contextually-sensitive, highly information-dense environment exhibited intuition-like behavior without the commensurate recognition that should be a precursor to such behavior. The participants seemed to be situationally aware, even though this contradicts currently accepted definitions of intuition and situational awareness. These data suggest that there is evidence of an extended definition of intuition. The phenomenon of intuition does not seem to depend solely on recognition, as Herbert Simon suggests, but also on accumulated experience in parallel situations in past experience, which may lead to

heightened sensitivity to contextual cues independent of expertise, a kind of "situational intuition."

The participants in this study who quickly solved the domain specific problem, though it was the first time they used this complex software system, could be said to be situationally intuitive. While participants who were eventually successful but less attuned to the contextual cues of the environment and the feedback of the interface would have a lower level of situational intuition, and the participants who could not complete the task would not be situationally intuitive.

Hierarchies of User Experience

The second research question is concerned with how the design of a complex professional software system might itself affect the information-seeking behaviors of those attempting to use it for the first time. By observing new users as they face such a system for the first time, this study showed evidence that different types of users responded differently to an identical software UX design.

One set of participants interacted with the software quickly and smoothly, exhibiting a kind of intuition in the process described earlier in this chapter as "situational awareness." Another set of participants showed facility in the part of the software that was familiar to their experience, and struggled when faced with a new domain-specific paradigm, but were able to discover how to use the interface of the software to complete the task at hand. And a third set of participants exhibited similar facility when doing familiar tasks, but when faced with the new domain-specific work were not able to leverage the software's interface to complete the task. These different levels of ability to

interact with the same software in different ways shows that user experience is complex, contextually sensitive, and varies depending on the user having the experience.

When MacDonald and Atwood described the five phases of human-computer interaction, they described historical periods in which HCI designers were evaluating the limiting factors of what could be considered a successful interaction with a computer: the System Reliability Phase, from the 1940s to 1950s, the System Performance Phase in the 1950s and 1960s, the User Performance Phase in the 1960s-1970s, and the Usability Phase from the 1980s to the 2000s, and the User Experience Phase from the 2000s to the present (2013). These historical phases were conceived from the hypothetical point of view of a designer of an HCI experience. For example, a designer in the System Reliability Phase will be concerned primarily with the computer not breaking, a designer in the System Performance Phase's priority is to consider how the user will use the software, a designer in the User Performance Phase considers the software's effects on how the user can perform better, a designer in the Usability Phase considers how the software can be used by anyone without extensive training, and a designer in the User Experience Phase considers how the software makes the user feel when they are using it.

However, a designer in the current User Experience Phase of HCI would not ignore the concerns that were paramount to designers in earlier phases. If software designed in the 21st century breaks the machine on which it runs, then that software has failed the primary criterion of the system reliability, even though the System Reliability Phase of HCI ended fifty years earlier. In this sense, a new historical phase does not replace the prior phase, but rather adds to it in a way that the concerns of each previous phase remain relevant as new phases emerge. Therefore we might consider these

historical phases as less of a timeline and more of a hierarchy where a designer must consider the concerns of the current phase, but must also ensure that the concerns of all prior phases are also considered, somewhat like Maslow's 1943 hierarchy of needs that describe increasingly complex stages of more complicated needs that can be satisfied once the more basic ones are met: food and water appear as foundational needs, then psychological needs like love and accomplishment, and eventually self-actualization (Figure 5.1).

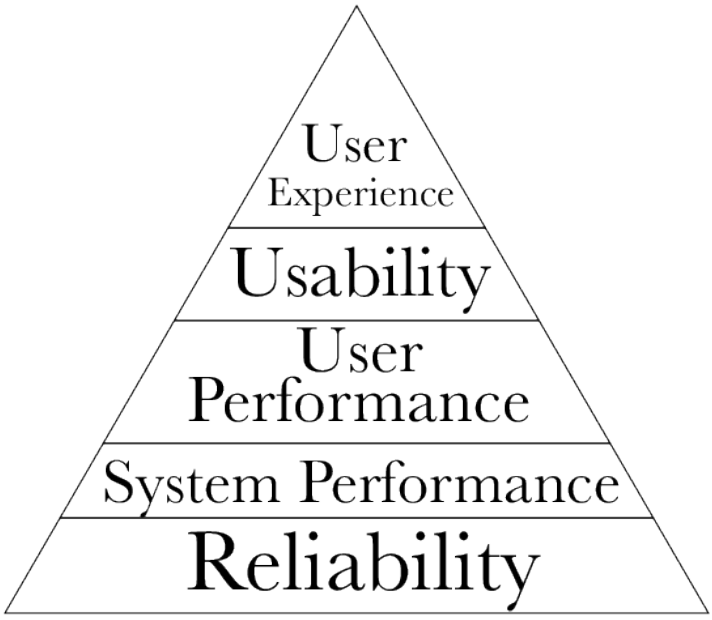

Figure 5.1 MacDonald and Atwood's historical phases, visualized as a hierarchy.

While MacDonald and Atwood's historical phases of HCI describe a progression from more primitive to more nuanced concerns, these phases refer to the practitioners who are building human-computer interactions, and I have suggested that this timeline is more accurately represented as a hierarchy. The participants in this study exhibited interactions with the tested software that varied based on their ability to participate in

interaction patterns of varying familiarity, suggesting that there may be a similar hierarchy that takes users' perspective in HCI into account.

I propose a new model, inspired in part by MacDonald and Atwood's historical phases, that maps out a hierarchy of user experience that takes into account both the design architecture of a given human-computer interaction and users capacity to make sense of and use that architecture. This hierarchy asks progressively more nuanced questions as interactions become more complex, and these questions can be used to evaluate the success of a given user experience in a multidimensional way, taking into account both the system's capacity to present an experience to a user, and a user's capacity to have a given experience at all (Figure 5.2).

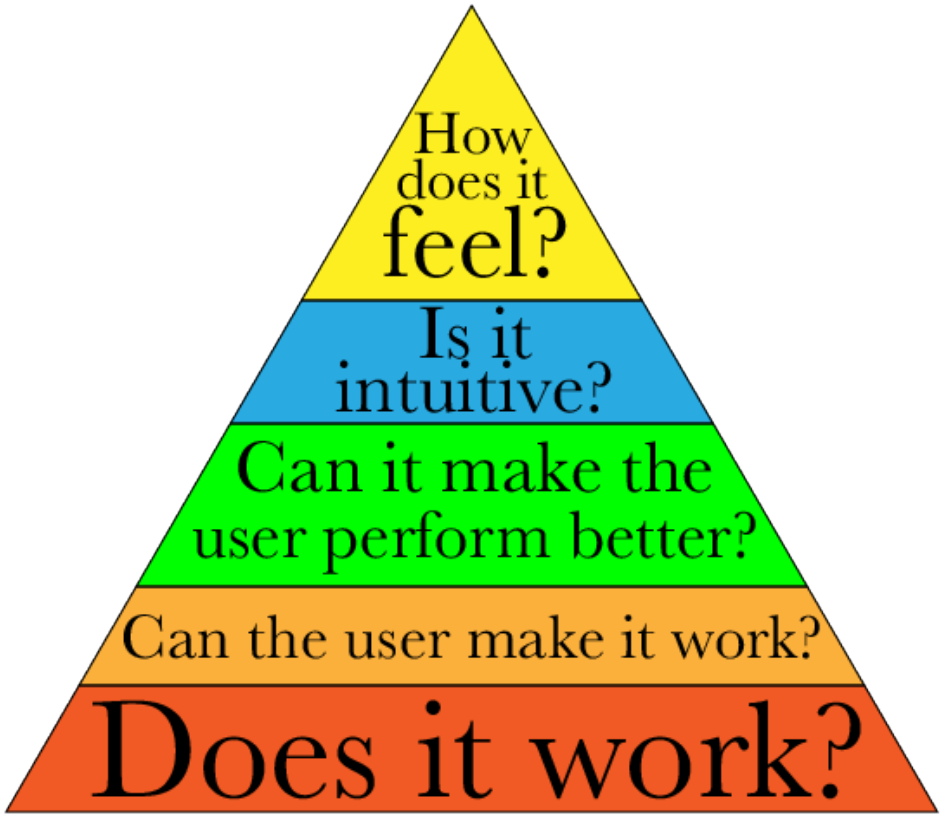

Figure 5.2 The User Experience Hierarchy.

First, as the fundamental baseline for all considered interactions, the model asks the question "does it work?" Any potential user experience relies first on the basic functioning of the system. Without this, no other interactions are possible, and thus no possible experience on the part of the user.

Second, the model asks "can the user make it work?" This represents the most basic level, simply making the interaction work. This level of interaction occurred in the study when users were able to manipulate video objects in Premiere, change their durations and place them in order. No further levels of interaction or experience are possible unless the software is able to support it, or the user is able to meet the software at this level.

Next, we turn to the question of "can it make the user perform better?" This is a more advanced variation of "can the user make it work?" At this level, the human-computer interaction has the ability to shape the workflow and behavior of the user, and the user has the capacity to improve their own performance with targeted use of the interaction. An example of this kind of interaction was evident in the study when users were able to change their behavior and condense steps to work more quickly and efficiently during the video editing task.

Fourth, this hierarchy asks "is it intuitive?" This is analogous to the historical Usability Phase, which was concerned with designers trying to create systems that would be available to wide audiences without specialized training. In this new model, this level of user experience interrogates a system's capacity for situationally intuitive interactions, which themselves depend on a given user's availability for those interactions. These

kinds of experiences occurred in this study when users in the Aware Group quickly succeeded in domain-specific tasks for which they had no prior training.

Finally, this model addresses the affective dimension by asking "how does it feel?" The apex of this hierarchy investigates both an interaction system's ability to leverage an emotional dimension to make a more effective connection with the user, and the user's ability to have that kind of an effective emotional interaction with a system. Because this study concerned novice users in a new and complex software system, no participants in this study exhibited this kind of user experience at this level of the model.

Users who satisfy lower levels of the user experience hierarchy have more increasingly nuanced levels available to them, but if they do not satisfy a more foundational level, the more complex levels are not available. For example, the users in this study who were not able to successfully use the software to complete a part of the task stalled on the second level of the hierarchy, so were not able to have a user experience that would improve their performance or be situationally intuitive. Meanwhile, the participants in this study who did have the elusive moments of situational intuition had already satisfied the lower levels of the hierarchy, making the higher-level experience available.

Reflection

This dissertation set out to explore relationships between user experience, sense-making, and learning as they relate to complex software processes. While exploring these relationships, the study also set out to discover what aspects of UX influence novice users' entry into a complex arena such as professional non-linear video editing, and how

those novice users use these tools without the usual signifiers and affordances of intensive training, and therefore with a diminished possibility of an intuitive experience.

The findings of this study showed that some users made sense of the new process in a way that looks like intuitive behavior. According to currently accepted definitions, these behaviors would not be considered intuitive per se, but this study reveals evidence that the threshold for considering an interaction to be intuition should be revised, as the relationship between a novice and a new experience might be more complex than previously thought. This phenomenon, here called "situational intuition," describes a new classification of intuitive behavior, marked by a heightened sensitivity to contextual clues.

This study also showed a complex interplay between users and user experience in a complex professional software system, suggesting a hierarchical model that could be used to evaluate UX in relation to a system's ability to render an experience for a user, as well as a user's ability to have progressively more complex user experiences.

Limitations

There are limitations to this study that must be acknowledged. First, as a qualitative study, the findings of this dissertation are not intended to be generalizable to a large population, so there is no guarantee that conclusions drawn from these findings will hold true in all cases. However, precautions were taken to ensure that the results were reliable and valid, allowing readers of this dissertation to interpret the meanings of the data from the rich descriptions herein, and determine the relevance and applicability of the study's findings to their own situations and experiences, or to serve as a platform for further research.

Because this study was specifically about the experience of novices, there were no participants who demonstrated sufficient expertise to reach the apex of the proposed hierarchical model of user experience ("how does it feel?"), so while that highest point in the model is conceptually supported and seems to be a logical extension of the presentation of historical modes of UX, it was not directly observed in this study.

As noted in the limitations section of Chapter 3, all of the participants in this study had were only people who had no prior experience with professional video editing software, and therefore the results cannot speak to users with passing knowledge of these skills, or users who might have more significant experience and who may wish to improve their skills in this area. Similarly, all the participants had declared interest in going into motion picture career fields, so the results here cannot reliably speak to incidental or casual users.

While the experiences of the participants in this study varied widely, all of the participants shared at least one point of relative privilege: all had been accepted to and had the means to attend a university as undergraduate students. Therefore, these results are not transferrable to other groups such as post-graduates, those who never attended or would never attend college, or those who are too young to be in college. The ages were somewhat uniform as well, only one participant was an age outlier, so these results could not reliably speak to experiences of users of widely varying ages such as children or senior citizens.

As a more practical matter, as this study was in process a methodological limitation emerged that could be improved in the future. Since the task required intense focus in the part of the participants, the self-talk process of the think-aloud protocol was

sometimes forgotten by the users; in those cases I would need to remind them to remember to think aloud about their process. This was especially true of participants who were having a more difficult time completing the task.

These expected and unexpected limitations may be addressed in both future studies stemming from this work, and in modifications to future methods of similar studies.

Implications and Future Research

The results of this dissertation, and the gaps left by limitations in the study, signal opportunities for future directions in research.

Perhaps the clearest example for the need for further study comes from the dimensions of the participant pool. Future studies could extrapolate this research design to examine different populations. Including participants with domain expertise in professional video editing, but who have only used other competing brands of DNLE systems might be able to test for more conclusive evidence of the higher levels of the user experience hierarchy model. Testing participants with expertise in the system being used in a future study could describe these phenomena in relation to users intending to improve performance rather than trying to gain a new skill. Studying more experienced users could give more experimental support to the "how does it feel?" section of the proposed hierarchical model.

As noted in the limitations section, participants who were focusing deeply during the performed task sometimes had to be reminded to continue the think-aloud protocol, and during periods of particularly intense focus, some users had little to no self-talk. A future study with a similar design might instead use a retrospective think-aloud protocol,

rather than the more common concurrent think-aloud protocol. As another insight into participants' processes, a future study might also employ eye-tracking software to cross reference where users are looking on screen in relation to the actions that they take in the course of a task.

Investigating children with less experience, or senior citizens who have largely avoided computer use could provide more details about the user experience at the lowest levels of that model. Similar variations could be studied by testing with participants who have not and will not attend college, or users who are not particularly interested in entering the professional field represented by the software being tested.

The results of this study could be applied to extend theories of information behaviors. As mentioned in Chapter 2, the emergence of human interface guidelines in software design encouraged developers to consider mental models when designing user experiences in software, however this study suggests that these mental models vary dramatically. While participants in the Aware group were exhibiting high situational intuition, participants in the Abandon group reported some information-seeking behaviors that tend to rely on other people for basic facility, or their mental models may altogether avoid either unfamiliar software or unfamiliar experiences in general. If software developers are relying on mental models to design systems, a better understanding of who uses which kinds of mental models under which circumstances could allow for future software designs that create different signal and different affordances for users with mental models that are outside of the typical scope of the software's design.

Additionally, this study appears to confirm aspects of schema and implicit learning theories. Participants in this study night not have had direct experience in the

tasks they performed, but some of them were able to leverage existing schema in their memories to the task at hand. If we consider implicit learning and schema theory in relation to the results of this study, participants who were successful in acquiring the new knowledge to complete the assigned task, without a concerted effort to acquire that knowledge, should have those new schemas more firmly entrenched in their minds than someone who would have attempted more deliberately to acquire those skills by a traditional method. A future variant of this study could examine this phenomenon by observing participants in a control group who learned the skills traditionally, and then over time compare retention of these skills across the two groups. A similar longitudinal variant of this study could also examine its relation to artifact theory and activity theory, to try and gain a better understanding of the long-term evolutionary nature of how users develop durable relationships to complex software processes through user experience.

In the areas of multi-stage skill acquisition and situational awareness, this study could be adapted to domains of avionic, military, or space exploration systems that typically require high situational awareness to succeed, to determine what kinds of parallel expertise or skills, other than high levels of domain expertise, might contribute to situational awareness, or how UX design could impact such systems.

The level of intensive independent pursuit of projects and hobbies by the participants in the Aware group is in keeping with results of a prior study where participants with past intensive independent pursuits tended to have subtler and more complex outcomes when editing the same video with provided archival footage (Tarr 2015). This emerging trend from multiple studies is ripe for deeper exploration. A future study might specifically look for measurable examples of this phenomenon that suggests

that self-directed pursuits, independent of curriculum and tangential to academics, might play an important role in abilities to learn and perform in the future.

Since this study looks at untrained users as they are first introduced to a professional video editing system, it could inform future versions of software. Though typically such software is used by people who are already highly trained to edit motion pictures, the users who come to be experts in using those tools were all novices at one point. Software designers who understand the behaviors of different kinds of newcomers as they approach their systems for the first time might be able to build systems that are conversant with new users that have very different starting points, thereby potentially increasing their future user base.

For educators in the media fields, this study sheds some light on how different kinds of new users approach the software tools that are necessary to become practicing professionals. Rather than assuming the same starting point for every student or driving all students through the same learning process modified only by speed of progression through instruction, this study provides evidence that shows that even when no prior experience is in place people approach these systems very differently, not simply varying by speed. One can surmise that this would require modified approaches to help different kinds of users achieve their fullest potential. Future research in how new users progress to levels of expertise from the novice state could help direct this more precisely.

Conclusion

This study suggests that there may be another type of intuition, "situational intuition," that is related to sensitivity to contextual cues, extending current accepted definitions of intuition that are primarily related to recognition and expertise. Further, the

data support the existence of a hierarchy of user experience, extending current definitions of UX to include a spectrum of interactions from the perspective of system designers and potential users. As a result of the outcomes of these data, this study proposes a new hierarchical model of user experience that could account for this augmented definition. This dissertation also suggests implications that could guide future research toward user experience aspects of human-computer interaction, expertise and skill acquisition, intuition and situational awareness, and media education.

REFERENCES

Apple Inc. (1992). *Macintosh human interface guidelines*. Addison-Wesley Publishing Company.

Apple Inc. (2009). Final Cut Pro v. 7. Cupertino, CA.

Apple Inc. (2011a). *A new way to organize*. Cupertino, CA.

Apple Inc. (2011b). *Complete file-based workflow*. Cupertino. CA.

Apple Inc. (2011c). Final Cut Pro X. Cupertino, CA.

Apple Inc. (2012a). *iOS human interface guidelines* (pp. 1–219).

Apple Inc. (2012b). *OS X human interface guidelines* (pp. 1–333).

Apple Inc. (2013). Final Cut Pro X product page. Retrieved May 4, 2013, from http://www.apple.com/finalcutpro/

Apple Inc. (2013). Final Cut Pro X - What is Final Cut? *www.apple.com*. Retrieved 2013, from http://www.apple.com/finalcutpro/what-is/

Apple Inc. (2015). *Apple Watch human interface guidelines* (pp. 1–58).

Avid, Inc. (1989). Media Composer v. 1. Burlington, MA: Avid, Inc.

Avid, Inc. (1990). *Avid Media Composer promo*. Retrieved from http://www.youtube.com/watch?v=js_eS6bFwdc

Avid, Inc. (2012). Media Composer v. 6. Burlington, MA: Avid, Inc.

Bargas-Avila, J., & Hornbæk, K. (2011). Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience (pp. 2689–2698).

Presented at the CHI 2011, Vancouver, BC, Canada: ACM. doi:10.1145/1978942.1979336

Bartlett, F. C. (1932). *Remembering: a study in experimental and social psychology*. New York: Macmillan Co.

Benson, Trisha A., Andrew L. Cohen, William Buskist, Regan A. R. Gurung, Arnie Cann, Pam Marek, Andrew N. Christopher, Cynthia S. Koenig, and David F. Reinhart. 2005. Faculty forum. *Teaching of Psychology* 32(4): 237–270.

Grantiz, Neil A., Stephen K. Koernig, and Katrin R. Harich. 2009. Now it's personal: antecedents and outcomes of rapport between business faculty and their students. *Journal of Marketing Education* 31(1): 52–65.

Bødker, S. (1989). A human activity approach to user interfaces. *Human-Computer Interaction*, *4*(3), 171–195. doi:10.1207/s15327051hci0403_1

Bødker, S. (1991). *Through the interface—a human activity approach to user interface design* (pp. 1–193). Hillsdale, NJ: Lawrence Erlbaum Associates.

Bonwell, C. C., & Eison, J. A. (1991). *Active learning: creating excitement in the classroom*. ERIC Clearinghouse on Higher Education. Retrieved from http://eric.ed.gov/?id=ED336049

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77–101. doi:10.1191/1478088706qp063oa

Cheng, J. (2012, January 15). Why the video pros are moving away from Apple. *arstechnica.com*. Retrieved May 4, 2013, from http://arstechnica.com/apple/2012/01/video-pros-apple-needs-to-acknowledge-the-pro-industry-and-fast/

Cohen, J. (1960). "A coefficient of agreement for nominal scales". *Educational and Psychological Measurement*. 20 (1): 37–46. doi:10.1177/001316446002000104

Conway, C. M., Bauernschmidt, A., Huang, S. S., & Pisoni, D. B. (2010). Implicit statistical learning in language processing: Word predictability is the key. *Cognition*, *114*(3), 356–371. doi:10.1016/j.cognition.2009.10.009

Denzin, N. K., & Lincoln, Y. S. (2011). *The SAGE Handbook of Qualitative Research (4 ed.)*. Thousand Oaks, CA: SAGE.

Dervin, B. (1983). An Overview of Sense-Making Research. Presented at the International Communication Association Annual Meeting, Dallas, TX.

Dervin, B., & Frenette, M. (2001). Sense-making methodology: Communicating communicatively with campaign audiences. In R.E. Rice & C.K. Atkin (Eds.), *Public communication campaigns* (3rd ed., pp. 69–87). Thousand Oaks, CA: Sage.

Dreyfus, S. E., & Dreyfus, H. L. (1980). A five-stage model of the mental activities involved in directed skill acquisition. *DTIC Document*. University of California, Berkeley.

Eisenstein, S. (1949). *Film form*. Orlando, FL: Harcourt.

Endsley, M. R. (1988). Design and Evaluation for Situation Awareness Enhancement. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 32(2), 97–101. http://doi.org/10.1177/154193128803200221

Endsley, M. R., & Jones, D. G. (2004). *Designing for situation awareness*. Boca Raton, FL: CRC Press.

Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review*, 87(3), 215–251.

Fielding, N. & Fielding, J. (1986). *Linking data.* Beverly Hills, CA: Sage.

Gannes, S. (1984, August). Lights, Cameras… Computers. *Discover*, *5*, 76–79.

Gibson, J. J. (1979). The theory of affordances. In *The ecological approach to visual perception*. Hillsdale, NJ.

Glesne, C. (1999). *Becoming qualitative researchers*. White Plains, NY: Longman.

Greene, J. L. (2007). *Mixed methods in social inquiry.* San Francisco, CA: Jossey-Bass.

Gruber, J. (2011, January). Uniformity vs. Individuality in Mac UI Design. *daringfireball.net*. Retrieved May 4, 2013, from http://daringfireball.net/2011/01/uniformity_vs_individuality_in_mac_ui_design

Hassenzahl, M. (2004). The interplay of beauty, goodness, and usability in interactive products. *Human-Computer Interaction*, *19*(4), 319–349. doi: 10.1207/s15327051hci1904_2

Hassenzahl, M., & Tractinsky, N. (2006). User experience - a research agenda. *Behaviour & Information Technology*, *25*(2), 91–97. doi: 10.1080/01449290500330331

Holloway, I., & Todres, L. (2003). The status of method: Flexibility, consistency and coherence. *Qualitative Research*, *3*, 345–57.

Kaye, J., & Sengers, P. (2007). The evolution of evaluation. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. http://doi.org/10.1145/1240624.2180962

Kemmis, S., & McTaggart, R. (2008). Participatory action research. In N. K. Denzin & Y. S. Lincoln (Eds.), *Strategies of qualitative inquiry* (pp. 271–330). SAGE Publications.

Kurtz, A. J. (2010). *The influence of aesthetics on the learnability and memorability of website interfaces*. (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses, UMI Number 3409102.

Law, E. L.-C., Roto, V., Hassenzahl, M., Vermeeren, A. P. O. S., & Kort, J. (2009). Understanding, scoping and defining user experience: a survey approach, 719–728. doi:10.1145/1518701.1518813

LeCompte, M. D., & Preissle, J. (2003). The role of theory in the research process. In *Ethnography and qualitative design in educational research* (2nd ed., 116–157). San Diego: Elsevier.

Lincoln, Y. S. & Guba, E. G. (1985). *Naturalistic inquiry*. Beverly Hills: Sage.

Linstone, H. A., & Turoff, M. (Eds.). (1975). *The Delphi method: techniques and applications*. Reading, MA: Addison-Wesley Publishing Company.

MacDonald, C. M., & Atwood, M. E. (2013). Changing perspectives on evaluation in HCI: past, present, and future. *CHI '13 extended abstracts on human factors in computing systems* (pp. 1969–1978). New York, New York, USA: ACM. http://doi.org/10.1145/2468356.2468714

Maslow, A. H. (1943). A theory of human motivation. *Psychological Review*, *50*(4), 370–396. doi:10.1037/h0054346

Microsoft, Inc. (2010). *Windows User Experience Interaction Guidelines*. Redmond, WA: Microsoft, Inc.

Munroe, R. (2013). Up goer five. *Xkcd* [web comic]. Retrieved from http://xkcd.com/1133/

Munroe, R. (2015). *Thing explainer: complicated stuff in simple words*. New York, NY: Haughton Mifflin Harcourt.

Norman, D. A. (1988). *The psychology of everyday things*. New York: Basic Books.

Norman, D. A. (1990). *The design of everyday things*. New York: Doubleday.

Norman, D. A. (1999). Affordance, conventions, and design. *Interactions*, *6*(3), 38–43. http://doi.org/10.1145/301153.301168

Norman, D. A. (2013). *The design of everyday things (revised and expanded edition)*. New York: Basic Books.

Patton, M. Q. (2007). Purposeful sampling. In *Qualitative Research & Evaluation Methods* (3rd ed., pp. 230–246). Thousand Oaks, CA: SAGE Publications.

Pogue, D. (2011, June 22). Apple's final cut is dead. Long live final cut. *The New York Times*. Retrieved May 3, 2013, from http://www.nytimes.com/2011/06/23/technology/personaltech/23pogue.html?_r=0

Reber, A. S. (2004). Implicit learning and tacit knowledge. *Journal of Experimental Psychology*, *118*(3), 219–235.

Ryan, W. N. (2011). *Learning-in-use with interactive artifacts*. (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses, UMI Number 3488323.

Sanderson, T. (2013). The up-goer five text editor. Retrieved from http://splasho.com/upgoer5/

Sarter, N. B., & Woods, D. D. (1991). Situation awareness: a critical but ill-defined phenomenon. *The International Journal of Aviation Psychology*, 1(1), 45–57. http://doi.org/10.1207/s15327108ijap0101_4

Simon, H. A. (1992). What is an "explanation" of behavior? *Psychological Science*, 3(3), 150–161. http://doi.org/10.1111/j.1467-9280.1992.tb00017.x

Spraul, V.A. (2012). *Think like a programmer*. No Starch Press: San Francisco, CA.

Swanson, K. G. (2012). *Learnable interfaces - leveraging navigation by design*. (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses, UMI Number 3521565.

Tarr, S (2015). Technicolor shades: considering privilege in modeling of information behavior. *Annual Review of Cultural Heritage Informatics: 2014*, 39-56

Telestream Inc. (2015). Screenflow, version 5. Nevada City, NV.

Tuhiwai Smith, L. (1999). *Decolonizing methodologies: research and indigenous peoples*. New York: Zed Books.

Wiedenbeck, S., & Zila, P. L. (1997). Hands-on practice in learning to use software: a comparison of exercise, exploration, and combined formats. *ACM Transactions on Computer-Human Interaction*, *4*(2), 169–196. doi:10.1145/254945.254967

Weiss, R. S. (1994). *Learning from strangers: the art and method of qualitative interviewing*. New York, NY: Free Press.

APPENDIX A—OPENING INTERVIEW QUESTIONS

These open-ended questions are prompts for the opening interviews. Follow-up questions then examine details revealed by the participants.

1. Please tell me your name and major.
2. Where did you grow up?
3. What do the people you grew up with do?
4. Tell me about the earliest time that you can remember using computers.
5. Tell me about a recent time when you figured something out using a computer.
6. Tell me about a recent time when you were frustrated by using a computer.

Depending on responses to 2, 3, and 4, ask follow-up questions about:

7. How did you use software growing up?
8. Describe what technology was where you went to school.
9. Tell me about using computers where you lived growing up.

Depending on responses to 2, 3, and 4, ask follow up-questions about:

10. Tell me about hobbies that you have now.
11. Describe hobbies that you had while you were growing up.

As the interview ends ask:

12. Other unpredicted follow-up questions based on prior responses.
13. Your name will not be used. Is there a pseudonym you'd like to use instead?

APPENDIX B—CLOSING INTERVIEW QUESTIONS

These open-ended questions are prompts for the closing interviews. Follow-up questions then examine details revealed by the participants.

1. How did the process feel?
2. Describe the parts of the editing process that made sense to you.
3. What kinds of things did it feel easy for you to do?
4. Describe the moments that didn't make sense to you.
5. What kinds of things felt difficult?
6. Other unpredicted follow-up questions based on prior responses.

APPENDIX C—EXPERT VERIFICATION INTERVIEW QUESTIONS

These open-ended questions are prompts for the expert verification interviews. Follow-up questions then examine details revealed by the participants.

1. Please read the set of basic tasks that will be given to a beginning user who has never user Premiere.
2. Please talk about your initial impressions of this set of tasks.
3. What do you think about the difficulty of the tasks (keeping in mind that the user has never used professional digital non-linear editors before)?
4. Keeping in mind that the goal of the tasks is to allow us to observe how the software's user experience affects how a new user approaches software, how do you think this set of tasks accomplishes this?
5. Other unpredicted follow-up questions based on prior responses.