ELICITING, ANALYZING, AND COMPARING MENTAL MODELS OF
COMPLEX AUDIO RECORDING SYSTEMS BETWEEN PROFESSIONAL AND
NOVICE RECORDING ENGINEERS

A Thesis in
Instructional Systems

by

Barry R. Hill

© 2006 Barry R. Hill

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Education

May 2006
The thesis of Barry R. Hill was reviewed and approved* by the following:

Susan M. Land  
Associate Professor of Education  
Thesis Advisor  
Chair of Committee  
In Charge of Graduate Programs in Instructional Systems

Brian Smith  
Associate Professor of Information Sciences and Technology  
Associate Professor of Instructional Systems

Peggy Van Meter  
Associate Professor of Education

Priya Sharma  
Assistant Professor of Instructional Systems

*Signatures are on file in the Graduate School.
ABSTRACT

One of the goals of a pre-professional college degree program is to help students acquire the knowledge and skills relevant to a particular discipline. For this to be effective and useful upon graduation, students must be able to function in a way similar to experts in that field—merely learning subject content is insufficient. Thus it is desirable to investigate how experts think and operate in the real world. How do professionals view content knowledge of the field? What meaningful connections do they perceive among this information? How do they apply this structural knowledge to actual procedures and problem solving? And how does an investigator determine this in order to inform the instructional design of an educational program?

To discover the mental process behind expert performance, cognitive and procedural techniques can be applied to produce mental models, which are internal conceptual and operational representations of a subject or system developed over time and experience. Cognitive maps can be generated that reveal how individuals structure their knowledge of a subject, showing meaningful connections among terms and concepts. These maps can then be compared to actual performance of a problem or task, analyzing how they approach a task and what they are thinking throughout the process. The literature supports the conclusion that experts think and behave quite differently from novices; therefore a great deal of study has been performed analyzing and comparing the mental models of experts and novices in a variety of disciplines. The purpose of this study was to compare the mental models of professional recording engineers in the music industry with seniors enrolled in a music recording degree program at a four-year college.
The results were intended primarily to inform the instructional design and implementation of the program as well as to provide recommendations that can be applied to other similar educational programs.

Eliciting structural knowledge of recording engineering concepts involved the application of the card sort procedure. Participants sorted a list of 20 terms and concepts into piles that indicated how they thought they related to each other. This data was then used to produce visual cognitive maps through multidimensional scaling statistical techniques. Procedural knowledge was analyzed by having participants think aloud as they completed tasks using Digidesign’s industry standard Pro Tools recording software. Operational and thought patterns were then derived through analysis and comparison from both sets of data.

As expected, the expert engineers possessed a more refined, highly structured knowledge of the field that enabled them to figure out various tasks even if they were inexperienced with the specific software used in the study. Differences between individual work experiences and personalities also led to variations in how they perceived recording systems and procedural situations. Student performance varied in terms of structural accuracy and ability to accomplish the tasks, but overall indicated a largely coherent mental model of how recording systems operate. One of the chief problems was the software itself; most often participants were clear on what they needed to do, but those with less experience with the Pro Tools system had difficulties understanding the fairly cryptic interface design. The few poor performers also suffered from inadequate mental models of how to solve the problem, compounding their lack of knowledge of the software. Recommendations are provided for designing similar instructional programs.
TABLE OF CONTENTS

List of Figures ........................................................................................................ viii
List of Tables ......................................................................................................... xi

Chapter 1. INTRODUCTION ........................................................................ 1
  Statement of the problem ..................................................................................... 1
  Background to the problem ............................................................................... 2
  Mental models ..................................................................................................... 7
  Expert-novice mental model comparison .......................................................... 8
    Expert-novice issues in the recording studio ............................................... 9
    Comparing analog and virtual systems .......................................................... 15
  Problem statement and research questions .................................................... 23
  Delimitations of the study ................................................................................. 23
  Definition of terms ............................................................................................. 26
    Recording system terms ............................................................................... 26
    Mental model terms ....................................................................................... 28

Chapter 2. REVIEW OF THE LITERATURE ............................................. 32
  Mental model overview ...................................................................................... 32
  Mental model development ............................................................................. 36
  Eliciting and representing mental models ....................................................... 39
    Cognitive analysis elicitation methods ......................................................... 42
    Representation methods .............................................................................. 44
    Cognitive analysis via qualitative techniques .......................................... 48
    Procedural analysis techniques ................................................................. 51
  General applications of mental model research ........................................... 53
  Mental model research in education and training .......................................... 55
  Expert-novice mental model comparisons .................................................... 59
  Implications for the current study ................................................................. 64

Chapter 3. METHODOLOGY ................................................................. 65
  Participants ....................................................................................................... 65
  Context of the study ......................................................................................... 67
  Research design ............................................................................................... 71
Chapter 4. RESULTS AND DISCUSSION ................................................................. 101
Card sort .............................................................................................................. 101
Experts .................................................................................................................. 101
Students ............................................................................................................. 120
Think-Aloud Protocol ......................................................................................... 127
The experts .......................................................................................................... 129
Summary of results for research questions 1 and 2 ......................................... 136
The students ........................................................................................................ 140
Summary of results for research questions 3 and 4 ......................................... 150
Experts and students compared ......................................................................... 153

Chapter 5. GENERAL DISCUSSION ..................................................................... 160
Factors involved in mental model development ................................................. 160
Importance of experience for model development ........................................... 160
Accurate models depend on appropriate system models during instruction ... 161
Mental models are unstable ............................................................................... 163
Accurate mental models require active thought and attention ....................... 163
Personal interest fosters active mental model development .......................... 164

Issues with the Pro Tools system ...................................................................... 165

Implications for instructional design ............................................................... 169
Design effective system models for instruction ................................................. 169
Learning tasks must be authentic and meaningful .......................................... 170
Instruction must provide extensive, guided practice ...................................... 171
Instructional experience must provide variety and novelty ............................ 171
Applying mental model analysis in the classroom ......................................... 172
The use of the Pro Tools recording system for instructional applications .... 173

Implications for future research ...................................................................... 173

Limitations of the study ..................................................................................... 175
References ...................................................................................................................................... 177
Appendix A. IRB Approval ........................................................................................................... 190
Appendix B. Card Sort Similarity Data Matrix ........................................................................... 192
Appendix C. Think-Aloud Exercise Instructions ......................................................................... 193
Appendix D. Investigator Think-Aloud Checklist ....................................................................... 195
Appendix E. Card Sort Concept Names And Brief Definitions ................................................ 196
Appendix F. Sample Transcription .............................................................................................. 198
Appendix G. Code List With Definitions ..................................................................................... 205
Appendix H. Sample Event Lists ............................................................................................... 208
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Instructions for setting microphone recording levels on the “easier” console.</td>
<td>12</td>
</tr>
<tr>
<td>1.2</td>
<td>Instructions for setting microphone recording levels on the more complex console. Note the greater number of controls on this console to see how this particular device can more easily distract novices.</td>
<td>13</td>
</tr>
<tr>
<td>1.3</td>
<td>A simple analog mixer.</td>
<td>16</td>
</tr>
<tr>
<td>1.4</td>
<td>Analog patchbay connecting the console with external signal processors.</td>
<td>17</td>
</tr>
<tr>
<td>1.5</td>
<td>Signal flow diagram showing multiple routing operations.</td>
<td>18</td>
</tr>
<tr>
<td>1.6</td>
<td>An example of a digital mixer featuring only a few physical controls per channel. Most functions must be accessed via soft buttons and scrolling through menus.</td>
<td>19</td>
</tr>
<tr>
<td>1.7</td>
<td>Pro Tools software-based mixer showing “real” controls.</td>
<td>20</td>
</tr>
<tr>
<td>1.8</td>
<td>Assignment matrix on an analog recording console.</td>
<td>21</td>
</tr>
<tr>
<td>1.9</td>
<td>Pro Tools output selector, which routes signals to various processing buses and hardware outputs. It does not replicate the traditional assignment matrix function, and it is difficult to grasp exactly what all these options are at first glance.</td>
<td>22</td>
</tr>
<tr>
<td>2.1</td>
<td>Cognitive map of physical therapy concepts generated by multidimensional scaling (from Jonassen et al., 1993, p. 64).</td>
<td>46</td>
</tr>
<tr>
<td>2.2</td>
<td>Pathfinder net of sports concepts (from Jonassen et al., 1993, p. 74).</td>
<td>48</td>
</tr>
<tr>
<td>3.1</td>
<td>The Trident Vector 432, a large format analog recording console.</td>
<td>69</td>
</tr>
<tr>
<td>3.2</td>
<td>The left diagram is the equalizer (tone control) section from the Sony console, the right from the Trident. Note how more complex the Trident controls are, including more knobs and switches. This is representative of how the entire console compares to the Sony.</td>
<td>70</td>
</tr>
<tr>
<td>3.3</td>
<td>Flowchart of research procedures.</td>
<td>73</td>
</tr>
<tr>
<td>3.4</td>
<td>TPL-KATS Card Sort screen.</td>
<td>78</td>
</tr>
<tr>
<td>3.5</td>
<td>Multidimensional scaling plot.</td>
<td>83</td>
</tr>
<tr>
<td>3.6</td>
<td>MDS stress and squared correlation output.</td>
<td>85</td>
</tr>
</tbody>
</table>
Figure 3.7. MDS weighted subject plot ................................................................. 87
Figure 3.8. MDS map showing outlier cards that were placed in different stacks, shown being “pulled” between both clusters. Note that the shaded clusters were manually added and labeled during analysis and not generated by the MDS process ............... 89
Figure 3.9. Analog patchbay routing signals between console and external processor. By connecting cables between both devices, engineers can easily see where the signal goes .......................................................... 94
Figure 3.10. The same signal routing in Pro Tools has no cable connections, requiring the user to figure out how to complete the virtual routing. It is actually more difficult than it appears in this example .................................................. 95
Figure 4.1. Expert MDS map ................................................................. 102
Figure 4.2. Expert 2 and 3 MDS map ................................................................. 106
Figure 4.3. Expert 3 MDS map ................................................................. 110
Figure 4.4. Investigator sort #1 MDS map ................................................................. 115
Figure 4.5. Investigator sort #2 MDS map ................................................................. 116
Figure 4.6. Investigator sort #1 ................................................................. 117
Figure 4.7. Investigator sort #2 ................................................................. 119
Figure 4.8. Student MDS map ................................................................. 121
Figure 4.9. Students 2, 4, and 5 MDS map ................................................................. 125
Figure 4.10. Expert 3 event list ................................................................. 130
Figure 4.11. Expert 2 event list ................................................................. 132
Figure 4.12. Expert 1 processing flowchart for task 1 ................................................................. 133
Figure 4.13. Expert 1 results. Bar heights represent the number of occurrences present for each type of coded event ................................................................. 135
Figure 4.14. Experts 3 and 2 integration of experience and task comparison ................................................................. 139
Figure 4.15. Quality of student models evidenced from protocol data ................................................................. 140
Figure 4.16. Student results: Correct, incorrect, & incorrect but with correct model concepts ................................................................. 141
Figure 4.17. Student 5 model quality compared to results ................................................................. 142
Figure 4.18. Student 5 correct/incorrect procedures compared to results ................................................................. 143
Figure 4.19. Student 1 model quality compared to results ................................................................. 145
Figure 4.20. Student 3 results. .......................................................... 147
Figure 4.21. Student 7 results. .......................................................... 148
Figure 4.22. Student 7 event list. ....................................................... 149
Figure 4.23. Expert & student model qualities compared. .................... 154
Figure 4.24. Expert 1 event list curve showing decreasing difficulty. .......... 156
Figure 4.25. Student 7 event list curve showing increasing difficulty .......... 157
Figure 4.26. Expert 1 and student 7 overall performance compared .......... 158
Figure 4.27. Combined expert and student model qualities .................... 159
Figure 4.28. Combined expert and student results .............................. 159
Figure 5.1. Factors involved in personal model development .................. 165
Figure 5.2. Overall model quality compared to results ......................... 168
LIST OF TABLES

Table 4.1. Expert sort results. ................................................................. 102
Table 4.2. Expert 2 and 3 sort results...................................................... 105
Table 4.3. Expert 3 results. .................................................................. 109
Table 4.4. Student sort results................................................................. 121
Table 4.5. Students 2, 4, and 5 sort results. .............................................. 125
CHAPTER 1
INTRODUCTION

Statement of the problem

The music recording industry is currently undergoing a major shift in how music is produced in the recording studio. Until recently nearly all production was performed on analog equipment, requiring significant investment in numerous types of hardware devices that filled the room. With advances in computer technology, however, much of the same capability is now available with software-based recording systems such as Digidesign’s Pro Tools. Essentially the recording studio has been replicated virtually in software, providing options for the engineer never before known. The issue for audio education, however, is that these systems are not easy to learn. Novices have a difficult time understanding conceptually how the system works and do not have the background and experience professionals have developed over the years. Professional engineers possess a knowledge structure that has been refined over time, resulting in a generic mental model of how recording systems operate. These conceptual models do not contain all the specific details, but rather provide the template that can be quickly applied to new situations. By examining how professionals approach a virtual recording system and eliciting their mental models, we can draw comparisons to the developing models students possess. Identifying patterns and major concepts from the experts can then be applied to improving audio education, an academic area that has suffered from a notable lack of targeted research and attention in instructional matters. Specifically, the purpose of this particular study is to elicit and compare mental models of recording systems, and Pro Tools specifically, between professional recording engineers and students who are
about to graduate from a college degree program in recording technology. This comparison presumably will reveal structural and procedural knowledge gaps and inaccuracies that students possess, providing direction as to how the curriculum could be modified to diminish this gap.

**Background to the problem**

The basic audio recording process is essentially a straightforward, relatively simple sequence of events consisting of capturing acoustic performances and sound events through microphones and other devices, routing those electronic signals to a recorder of some type, and manipulating them in certain ways to produce a final recording. With analog recording systems this requires the use of a *recording console*, a complex hardware device that inputs microphone signals and routes them to a separate recording device called a *multitrack recorder*, usually detouring them to external *signal processing* devices along the way. Multitrack recorders are capable of storing many separate musical parts, each of which can be recorded at different times. The console is the central device that routes and processes these signals and is the main interface between the engineer and audio signals. In the audio recording industry the term for how these signals are controlled and routed is called *signal flow*.

Fundamental recording procedures and techniques using such equipment have not changed much over the years—analog recording systems primarily differ in terms of scale and features, not operational concepts. It was therefore fairly straightforward for novices to gradually develop an appropriate concept of how such systems operated. Traditional analog systems are physical devices, providing tactile and visual representations of operational procedures; novice engineers can thus more easily “see”
the underlying schemes of operations. Once they learned one particular recording console they could generally transfer that knowledge to another system. The basic operational controls would always be present, differing only in location, complexity, and perhaps terminology. Fundamental signal routing options, though they may be accomplished in slightly different ways, were always available in order to perform standard procedures. An engineer with an appropriate knowledge of how recording systems operated would know what to look for when confronted with a new system, providing efficient transfer of knowledge and a much-reduced learning curve.

The difficulty in recent years stems from the proliferation of digital and software-based recording systems. Such systems essentially take all of the various processing and routing equipment in the recording studio and duplicate their functions within software. This “studio-in-a-box” concept has tremendous appeal and offers powerful production capabilities, but the result is a very complex product design based on a virtual interface where core functions are accessed via numerous menu options, windows, and “soft” buttons. There is no way to “see” exactly what is happening to the audio signals as is possible on a hardware device where every function is physically controlled by knobs, switches, and faders organized in a logical, ergonomic layout. Experienced engineers can apply their mental knowledge and understanding of how recording systems work and eventually figure out newer digital devices. Novices, however, have no such background to draw upon and must learn procedures by rote for each system they use. Development of recurring patterns comes much slower, if at all. The wide variation in product design does not foster and support user awareness of such core operational patterns, making it
even more difficult for novices to develop standardized structural concepts in the field of audio recording.

However, in spite of these design differences all recording systems do in fact provide core operational patterns and concepts for routing and processing audio signals during the recording process. Student engineers need to learn these fundamental operating patterns of signal flow in order to transfer their skills to the wide variety of recording systems and devices on the market. The goal is for young engineers who find themselves in front of a different system to not have to start from scratch, but to draw on prior experience and knowledge about how such systems work underneath all the various controls and options. In other words, they need to develop an accurate mental model of recording systems. A mental model represents an individual’s conceptualization of how a system operates, based on experience and/or instruction. Borgman (1986) explains that mental models “describe a cognitive mechanism for representing and making inferences about a system or problem which the user builds as he or she interacts with and learns about the system” (p. 48). A model of a device is “formed largely by interpreting its perceived actions and its visible structure” (Norman, 2002, p. 17). For a simple example, take the volume control on stereo units. Most often it is a rotary knob, and users readily expect that by turning it to the right (clockwise) will increase the volume. Why? Because this action has become “standard” in our culture, and people have learned this through experience with similar knobs. Now, take a light switch that in the United States turns on a light when switched to the up position. When we see such a switch on any device, we predict it will activate some function. When the individual is confronted with a device designed in the U.K., however, this mental model is violated by the fact that flipping the
switch up will actually deactivate its function. Another example is the automobile. Most of the basic controls function the same regardless of manufacturer or model. If an experienced driver sits in a new vehicle they have never seen before, they can readily predict that depressing the right foot pedal will cause the car to increase speed and that depressing the pedal immediately to the left will decrease speed. Turning the steering wheel to the right will veer the car to the right. Now, imagine an automobile where the turning mechanism is a horizontal slider mounted on the dash. This violates the driver’s mental model of steering, but perhaps if the control were labeled appropriately accommodation might not be too difficult. However, if the slider were mounted vertically, the driver’s conceptual model of steering would be violated more severely for two reasons: people are used to turning a wheel to steer a car, and traditional design dictates that a vertical control adjusts something up and down, not left and right. In the recording industry, long-established standards of analog equipment and procedures helped develop relatively consistent mental models among professional recording engineers. This is rapidly coming “unglued” for two reasons: 1) digital and software-based systems by their very nature violate, or at least obscure, the analog paradigm to some degree and traditional methods of recording audio, and 2) manufacturers seem determined to develop products purposefully different from their competitors, preventing consistency among systems that would help foster some semblance of a common conceptual model.

Specifically for the purposes of this study, the primary issue in audio education today is the increasingly widespread use of a software-based recording system called Pro Tools, developed by Digidesign, Inc. Pro Tools has become the de facto standard in
today’s recording and film industries, and so educational programs in recording engineering are selecting this particular product so students will be familiar with the system once they enter the industry. This makes logical sense on the surface, but the limitations of such an approach are many-fold: 1) students have difficulty developing accurate mental models of such a system since functions and operations are virtual and are rarely transparent to the user, 2) despite the fact that Pro Tools is a “standard”, there are numerous other software- and hardware-based recording systems in use today. They represent a wide array of design concepts and operational patterns, and so the issue is again the development of accurate system models that allow a user to effectively and efficiently transfer their generic model to any new system, 3) students learning without appropriate system models do two things: they learn functions by rote (which are therefore not generally transferable to novel situations), and they develop their own conceptual models, which are much more likely to be inaccurate and inappropriate to help them in future situations (Norman, 2002; Parush, 2004).

From personal experience, we have found at our institution that students, who have learned recording techniques on analog equipment first before moving on to digital systems, have not seemed to do well in transferring the knowledge gained from the analog experience to the software-based Pro Tools system. Pro Tools uses the typical analog recording console and operational patterns as its design metaphor (though rather loosely), but as it is a virtual representation many of those functions and procedures are not immediately visible, nor are they straightforward, often lost in a maze of menus, windows, and new terminology. The result is that many students do not seem to apply standard analog procedures and struggle to find ways to accomplish recording tasks—in
effect throwing out the conceptual models we have attempted to help them develop up to that point. Students who have performed adequately on analog consoles, which are fairly straightforward to understand, have difficulty functioning effectively within the virtual software environment.

With this in mind, it would be informative for audio educators to find out what mental models professional recording engineers possess about recording systems and how they apply them to various tasks specifically using Pro Tools systems. Using this as a benchmark, we can compare with students who have largely completed a degree program in recording engineering and determine what models they possess for such a system. The differences can provide clues that could inform instructional strategies and practices throughout the curriculum, including decisions regarding selection of system types (hardware and software) for installation in school recording studio laboratories.

**Mental models**

How do human beings organize and structure the knowledge they accumulate over time? How do we approach a novel situation and attempt to predict potential solutions and outcomes? Why do individuals carry so many varying, and often conflicting, notions about even simple concepts and principles? Research into mental models has helped psychologists, scientists, design engineers, teachers, and many others understand and work with how humans structure knowledge. We want to know how people understand a specific content domain, whether that be the law of gravity, the major concepts in organic chemistry, or the operational workings of a nuclear power plant. Educational psychology research has shown that individuals take with them widely varying understandings of concepts even after undergoing the same instructional
experience (Perkins & Unger, 1999; Perkins, 1992). For instance, studies have revealed that a majority of high school graduates do not possess accurate understandings of even the most basic principles in science and math, such as the distinction between heat and temperature (Perkins, 1991). These are not the low achievers, but the “bright” academically motivated students who are college-bound to major in these areas.

Engineers in a nuclear power plant can be suddenly faced with flashing and buzzing warning indicators, necessitating immediate diagnosis as to the probable cause. Unfortunately this becomes a far more complicated exercise than is implied in training; there are a multitude of differing possibilities, requiring the operators to have an absolutely accurate mental model of how the entire system works under various normal and problematic conditions, not all of which are easily predictable for training purposes (Norman, 2002). A more common example of simple mental model usage is how people believe a home heating thermostat operates. Upon entering their chilly home, people will typically turn the temperature up much higher than they really desire under the impression that it will warm the house faster. This is incorrect, but it stems from an inaccurate conceptual image of how the system works (Norman, 2002). Identifying how people perceive systems and content domains is tremendously useful, and mental models provide a theoretical framework in which to accomplish this.

**Expert-novice mental model comparison**

One of the most common applications for applying mental model research is the comparison between experts and novices in a given content domain (Jonassen, Beissner, & Yacci, 1993; Gentner & Stevens, 1983; Carley & Palmquist, 1992; Villachica et al., 2001). For instance, professionals approach a task or problem with not only a vast
knowledge base from which to draw upon, but also the ability to utilize that information in ways that come only from long experience (Johnson, 1988; Jonassen & Henning, 1999). It is much easier to package knowledge base content for instructional purposes than it is to help beginners think and operate in ways that are as efficient and effective as that exhibited by professionals; in other words, it is one thing to deliver content during instruction, but quite another to help the learner structure that information in meaningful ways that can then be applied in real situations. The experience gained by experts over the years of performing tasks and confronting new problems develops specific, purposeful relationships between the items stored in their memory. This allows them to approach a novel situation, identify key components relevant to the immediate task, and develop a solution in a more streamlined manner than a beginner is able to (Gentner & Stevens, 1983; Johnson, 1988). In artificial intelligence this concept is called “heuristics”, where the knowledge tree relationships are such that a diagnosis can be determined by focusing on only those aspects that are relevant, ignoring other information along the way. A beginner, on the other hand, must review each line of data and available courses of action, hoping to eventually discover the correct combination. By examining and comparing the mental knowledge structures of both groups, performance gaps can be identified and results employed to improve instructional strategies and practice (Jonassen et al., 1993; Jonassen, 1995).

*Expert-novice issues in the recording studio*

Professional recording engineers have traditionally worked in a complex workspace that involves the interconnection and interaction of many different pieces of equipment. Knowing how everything is connected, understanding how they relate to each
other, and being able to perform the task of recording audio signals amongst such a complex interweaving of devices takes time and experience. Whereas the novice sees only a vast multitude of knobs, switches, and wires and can only operate the equipment by rote steps, the expert can “see” overall patterns of functionality. They can immediately zero in on one specific control amidst the hundreds that are typically found on a major recording console. If something goes wrong, they can quickly estimate where the fault might lie and take corrective action, often with little thought. In some cases the engineer may have to determine a work-around, finding a different way to accomplish a task than is normal. This requires a thorough understanding of the entire system from which to consider possible options and finalize a course of action. Additionally, this ability then allows them to quickly understand and operate any different recording system with which they have no prior experience. Since the underlying patterns and concepts of operation remain the same, they do not need to start from scratch, but rather can simply find how certain functions are implemented on that particular system.

This is the core issue when instructing novice engineers. It is a simple matter to train individuals on a particular piece of equipment or software. Procedural tasks are easily laid out in step-by-step instructions and reinforced with job aids. Over time the novice will become faster in performing these tasks, eventually completing them with little conscious thought. However, when faced with a different type of recording system, their performance falters as they typically see the new system as a completely different problem to figure out. To illustrate this point, the following two graphics, taken from job aids developed for recording students, indicate how to perform identical tasks on two different recording consoles. The objective is the same—route an incoming microphone
signal to an external multitrack recorder. However, the number of controls available on each console is different, the second being more complex. For a beginner, this means that there are more distractions as they attempt to figure out which controls to operate. For an expert, the core controls needed to route microphone signals in this situation are the same on both, the task remaining only to determine where they are located on each device. The first example (see Figure 1.1) represents an analog console that students at the author’s college are exposed to during their first recording class. The second (see Figure 1.2) is the more complex console they then learn to use the following semester, attempting to apply their developing mental models of fundamental operating patterns.
Getting a mic signal to the multitrack

1. Select output bus in the assignment matrix. Use DIRECT if going to same # bus. Pan Odd/Even if assigning stereo tracks.

2. Turn on phantom power if using a condenser mic. Do NOT turn up the mic preamplifier yet.

3. Turn up output bus level to tape. Start at zero, which is about one o’clock on the pot.

4. Put multitrack into input or sync (overdub) mode. Arm tracks for recording.

5. Lastly, slowly turn up the mic pre while musician is playing into the microphone. Make sure you watch meters—don’t listen for level. If you need more gain, don’t turn the pre up all the way. Turn it down first, then push the GAIN button.

Figure 1.1. Instructions for setting microphone recording levels on the “easier” console.
Getting a mic signal to the multitrack

1. Select output bus in the assignment matrix. Use DIRECT if going to same # bus. Turn on MTK Pan and pan odd/even if assigning stereo tracks.

2. Turn on phantom power if using a condenser mic. Do NOT turn up the mic preamplifier yet.

3. Turn up output bus level to tape. Small fader is channel path if Fader Reverse is selected in master section.

4. Arm the multitrack recorder track that corresponds to the bus output # selected in assignment matrix.

5. Last, slowly turn up the mic pre while musician is playing into the microphone. Watch meters—don’t use your ears.

Figure 1.2. Instructions for setting microphone recording levels on the more complex console. Note the greater number of controls on this console to see how this particular device can more easily distract novices.
Although the second console is more complex, it is quite similar to the first in functionality. However, even though instruction focuses on basic operational patterns, many students struggle to make these connections between the two. Presumably the major cause for this is the limited time-on-task they have had at that point, having only completed a one-semester course using the first console. Mental models require time and experience to adequately develop. Even considering this, from a pedagogical perspective the ability for a novice to learn on a relatively simple analog console, then continue their exposure on a more complex analog console, is a powerful approach that helps them begin generating a basic model for recording systems. This is far more than many educational programs are able to provide due to the high cost of this type of equipment and expertise required for operation and maintenance.

The real issue, especially considering the fact that many schools are attempting to train engineers on digital and software-based systems, is the cognitive leap from fundamental signal flow, more easily seen and understood on analog equipment, to the vastly complex operations of systems available today, including Pro Tools. How do most professional engineers employ Pro Tools, considering that these virtual recording systems are less transparent to figure out, require learning various menu commands and keyboard shortcuts, and offer a multitude of ways to get lost? They bring with them long years of experience with analog equipment. This is the paradigm they are accustomed to and which has influenced the mental models they have developed. It is also the design paradigm on which Pro Tools and most other software-based recording systems are founded. This works best, of course, when the engineers using such systems also possess that paradigm as the core of their mental models, something novices do not possess
unless given the opportunity during their education. Therefore, this study seeks to
determine how much of this is in fact what is happening by eliciting the knowledge
structures professional engineers possess and how they approach various recording tasks
using a Pro Tools system. Doing the same with novices (students) should clarify some of
the differences with how each of these groups conceives of recording systems.

Comparing analog and virtual systems

To better clarify why the difference between analog and digital recording
equipment is so profound, an in-depth explanation of these systems is necessary. Analog-
based equipment consists of separate pieces of hardware that function to route signals,
process them in some way, and provide storage (recording). These devices feature
dedicated controls, meaning individual switches, knobs, or faders (sliders) for each
function. The control surfaces of analog units are visually organized in a manner that is
usually straightforward to understand and operate. Digital-based systems feature either a
hardware device that performs the routing and processing functions (and sometimes
recording as well) or software-based systems in which all operations are represented and
performed virtually on a computer. Typically many, or even most, available functions are
accessed via a limited number of “soft” buttons and multiple menus. Each “soft” button
may control many different functions depending on which “mode” it is in, allowing the
manufacturer to reduce the number of physical controls on the unit. Most typical
recording functions and signal routing is accessed through scrolling menus that the user
must learn over time. Though these digital systems provide an enormous range of
productivity options, they are much less transparent to figure out, especially for the
novice.
Most analog systems look and function similarly; while the operational functions and layout vary somewhat between different models, once an engineer learns “the basics” they should be easily able to transfer that knowledge to a different analog system. For example, consider the primary device that is at the heart of the recording process, the recording console. This device features identical, vertical columns of switches and knobs (termed *rotary potentiometers*, or more briefly *pots*); each column controls one channel, or signal (see Figure 1.3).

![Figure 1.3. A simple analog mixer.](image)

When the engineer wants to adjust a particular signal, they simply reach for the appropriate switch or knob on that particular channel. If they want to connect a particular external device, such as a reverberation processor, they take a cable and physically connect the inputs and outputs of that device to the appropriate plugs on the console, much like the process used for early telephone switchboards (see Figure 1.4).
Figure 1.4. Analog patchbay connecting the console with external signal processors.

The basic process for recording a microphone signal is to plug it into one channel of the console (mic input), increase the amplitude level of the signal (mic preamplifier pot), perhaps process it in some way such as through an equalizer (tone control), then route it to a particular output bus that determines which number track it will be recorded on the multitrack recorder. Recording systems feature a number of tracks, each storing one component of a song such as a vocal or piano part. Once each signal arrives at the recorder, it is simultaneously returned back into the console so the engineer can monitor the results. These returning signals are also used for setting up a separate submix of all the song’s parts (tracks) that are then routed to the headphones worn by musicians in the studio. This allows them to hear what has already been recorded so they can play along. All of this can become quite complex in that there can be numerous audio signals running throughout the console in multiple directions as well as between the console and all the various external equipment such as effects devices, recorders, monitor speakers, and headphone systems (see Figure 1.5). Keeping track of this complex signal flow requires a clear mental picture of what is going on and again supports the need for an accurate system model.
The critical difference between analog and digital systems is how the core signal flow operations and options are represented to the engineer. Digital hardware may feature a few familiar physical controls, such as microphone level pots and volume faders, but most functions are microprocessor based (see Figure 1.6). This provides the power of digital processing and many options for controlling audio signals at a much less expensive cost than is possible with traditional analog equipment. However, figuring out all those functions and finding them amongst multiple screens of menu-driven commands is a daunting task for even experienced professionals, who at least have a frame of reference from which to draw upon.
Software-based recording systems such as Pro Tools have essentially replaced stacks of analog equipment, providing all their functionality (and more) within the software environment. This has been made possible in recent years through the advance of digital signal processing capability, improved programming algorithms, and very powerful computer systems only dreamed of a decade ago. This has revolutionized the recording industry, but has dramatically changed the interface between the recording engineer and the sounds they are capturing. Whereas with analog consoles the engineer need only reach for a dedicated control for a specific function, in software the user must figure out where most of those functions are located and how they are utilized. Often, certain standard controls are represented on the screen graphically similar to the “real thing”, such as level faders and buttons (see Figure 1.7). This follows the analog
paradigm and provides for positive transfer of previous analog knowledge to software systems.

Figure 1.7. Pro Tools software-based mixer showing “real” controls.

Most of the other options, however, are accessed via one of the many menu commands and pop-up controls. Even some of the terminology has changed, making direct comparisons with analog consoles difficult. For example, on an analog console each channel has an assignment matrix, which is simply a column of numbered buttons that correspond to numbered tracks on the multitrack recorder. When routing an incoming microphone signal, the engineer will select one particular track number (output bus) for that musical part, thereby sending it to the recorder (see Figure 1.8).
Figure 1.8. Assignment matrix on an analog recording console.

In Pro Tools, there is no assignment matrix, and yet there is a similar control on each channel. Pro Tools allows the engineer to select *track outputs*, which on first glance would seem to be identical in operation to the assignment matrix. However, as this is a virtual recording system there is no such thing as separate multitrack recorder tracks. The recorded track is automatically related to that particular channel. The *track output* selector instead allows signals to be routed to either virtual internal buses or the external hardware interface (connected to the computer), which can then either be heard via loudspeakers, routed to a headphone system, or sent to an external processing device—altogether different functions than what a traditional assignment matrix performs. To make matters worse, this function is not even labeled *output select* or
something similarly descriptive, but rather A 1-2, thereby obscuring its function from a new user (see Figure 1.9).

Figure 1.9. Pro Tools output selector, which routes signals to various processing buses and hardware outputs. It does not replicate the traditional assignment matrix function, and it is difficult to grasp exactly what all these options are at first glance.

Although such a design can be quite confusing for novices, once an engineer begins to understand and become comfortable with the freedom and flexibility this system provides, productivity increases dramatically. The challenge is understanding conceptually what is going on when routing signals to the actual hardware outputs as
opposed to internal, virtual destinations. An experienced engineer readily sees this, but novices cannot. Again, the issue is bridging the gap for students to help them better think like professionals as much as is possible in an instructional environment.

**Problem statement and research questions**

By comparing the mental models of expert recording engineers with college students who have largely completed their program of study in recording technology, any discrepancies in model development can be identified and used to improve instruction in the degree program. In order to determine potentially useful recommendations, the following questions will be explored:

1. What mental models do professional, expert recording engineers possess about recording systems?
2. How do these engineers apply this structural knowledge specifically to Pro Tools recording systems?
3. What mental models do graduating seniors possess about recording systems?
4. How do these novices apply this structural knowledge to Pro Tools recording systems?

**Delimitations of the study**

Although the primary objective is to compare experts’ and novices’ mental models for recording systems, it is probable that no absolutely clear-cut conclusions may result. Though there is a long established literature of methodology supporting the elicitation of mental models and expert/novice comparisons, extracting mental models is a somewhat fuzzy exercise. Time limitations, as well as the fact that there is but a single
researcher for this study, dictate a finite number of professionals and students that can be investigated. Additionally, the field of audio production is fairly unique in that individuals drawn to the profession are inherently artistic, creative, and perhaps a tad non-conformist, meaning they do not always adhere to a traditionally imposed, carefully structured method as is typical for most “normal” occupations. Whereas accountants and doctors are expected to go by the book, audio engineers have no such long-standing protocols or expectations. Each engineer learns perhaps a little differently and approaches the science and art of making recordings in their own way. Ask an engineer their preferences for processing musical parts during tracking and you are certain to get as many different answers. Certainly there are core operational patterns that recording equipment has standardized upon, and this is the central premise behind this study. However, it will be interesting to see the results of the professionals in particular, and it is noted that if there were opportunity to work with ten or twenty more engineers it would merely add a richness and variety to the outcomes. The expectation is that in spite of this there should emerge certain core patterns that will be useful for instructional purposes and is therefore worth the energy and time spent.

Usefulness of the results centers around the fact that the investigation involves students who have attended the same college degree program. There are numerous advantages to this approach: while the hope is that the results (and methodology) can potentially be useful to all audio educators, the primary concern is to investigate the situation at the researcher’s institution specifically so as to make a direct impact on program design. Additionally it makes more sense to compare students who have
undergone the same educational experience in order to eliminate a host of confounds related to varying instructional approaches, lab equipment, curricular designs, and so on.

   Therefore it is presumed that specific outcomes from this study may not be generalizable to other institutions, but that many of the issues brought forth can indeed inform audio educators and administrators as they consider their own program design and equipment selection.
Definition of terms

Recording system terms

- **Recording console**
  - The set of controls (and the enclosure containing them) by which the recording engineer selects various input signals (mic, line, or tape playback), adjusts their relative volume, tone, etc., and routes them, either to multitrack tape, mixdown, control room monitors, studio headphones, or other destination. Also called a board or desk (Wadhams, 1988).

- **Multitrack recorder**
  - A tape recorder, console, or other device capable of handling more than two tracks of information separately (Wadhams, 1988).

- **Signal processing device**
  - Any audio system used to alter the characteristics of a signal passing through it. Examples: equalizer, compressor, noise gate, etc (Wadhams, 1988).

- **Recording system**
  - Audio system for recording sound ([www.wordreference.com](http://www.wordreference.com)).
  - Recording systems are comprised of a console, signal processors, and recorders. These may be separate hardware devices, or the functions may be included within a single unit or software application.

- **Audio signal**
  - Audio information (music, sound) transmitted through a recording system via electric signal or digital data.
• Signal flow
  o …all possible signal paths in a recording console or other audio system
    (Wadhams, 1988).

• Analog recording
  o Any method of recording in which the recorded waveform is a continuous
    representation of the original signal.

• Digital recording
  o Any recording process in which the voltage of the original analog signal is
    repeatedly sampled and stored as numerical data during the recording, then
    reconverted back to an analog waveform during playback (Wadhams, 1988).

• Software-based recording system
  o A recording system that provides the functionality of all components (console,
    recorder, processing) entirely within the software application.

• Pro Tools
  o A specific software-based recording system developed by Digidesign, a
    subsidiary of Avid, Inc. Pro Tools has become the most widely established
    system in the recording industry and consists of the software application,
    which includes all console, recording, and processing functionality, as well as
    a hardware device that interfaces between analog signals and the host
    computer.
Mental model terms

• Mental models
  o Psychological representations that “describe a cognitive mechanism for representing and making inferences about a system or problem which the user builds as he or she interacts with and learns about the system” (Borgman, 1986, p. 48).
  o For this study, mental models represent structural and procedural knowledge that engineers develop about recording systems through experience. This allows them to understand and predict how such systems operate and therefore how to predict a new system they may encounter.

• Declarative knowledge
  o Cognizance or awareness of some object, event, or idea (Jonassen et al., 1993).
  o Refers to content knowledge, such as facts, but not necessarily the relationships among this information.

• Structural knowledge (knowledge structure, cognitive structure)
  o An intermediate category of knowledge, structural knowledge “mediates the translation of declarative into procedural knowledge and facilitates the application of procedural knowledge” (Jonassen et al., 1993).
  o “…a hypothetical construct referring to the organization of the relationships of concepts in long-term memory” (Shavelson, 1972, pp. 226-227).
  o Structural knowledge represents the advanced connections gradually developed by individuals as they organize and make sense of declarative
knowledge within a particular content domain. Experts possess a highly structured, advanced hierarchy of content knowledge whereas novices are only beginning to “see” connections among all the various terms they encounter.

- **Procedural knowledge**
  - Applying declarative and structural knowledge. “Accessing and interrelating relevant schemata and extracting the relevant attributes to apply to the situation” (Jonassen et al., 1993).
  - For this study, the interest in procedural knowledge is how experts and novices apply their structural knowledge of recording systems to a specific software-based system (Pro Tools).

- **Expert**
  - “A person with a high degree of skill in or knowledge of a certain subject” ([www.dictionary.com](http://www.dictionary.com)).
  - For this study, expert participants are professional recording engineers with several years experience with recording systems.

- **Novice**
  - “A person new to a field or activity; a beginner” ([www.dictionary.com](http://www.dictionary.com)).
  - For this study, novice participants are seniors majoring in recording technology in a college degree program.

- **Pairwise ratings (similarity ratings)**
  - “…rate the similarity between concepts in an individual’s cognitive structure” (Jonassen et al., 1993).
This type of data reveals how direct or indirect an individual associates various terms within a content domain. When processed into Pathfinder nets, the resulting diagram indicates how closely or distantly related concepts are by distributing them spatially.

- Card sort
  - “...a technique that seeks to elicit individual conceptual frameworks by giving a subject a collection of cards each pre-printed, for example, with a word or phrase and asking the subject to partition (sort) the cards into subsets based on the subject’s own criteria (http://www.cogsci.northwestern.edu/cogsci2004/papers/paper411.pdf).
  - “The card sort technique is an advanced level sorting task that can be used to identify how concepts in a content area are organized in a learner’s knowledge structure” (Jonassen et al., 1993).
  - For this study, a list of significant concepts of recording systems is presented to each participant who then sorts them into piles of their own choosing. This indicates how they see relationships among the various terms, thereby representing their structural knowledge.

- Cognitive map
  - Spatial representations of cognitive structure, derived by taking pairwise relatedness data and using the semantic distance between the concepts to indicate geometric distances between the objects. (adapted from Jonassen et al., 1993).
Cognitive maps are dimensional representations generated through statistical processing techniques such as multidimensional scaling and cluster analysis.

- Cognitive analysis
  - The process of extracting an individual’s structural knowledge of a particular content domain and generating a cognitive map.

- Pathfinder nets (PFNets)
  - A visual diagram in which “concepts (objects, events, actions, or entities) are represented as nodes and relationships (among the concepts) as links connecting the nodes” (Jonassen et al., 1993).

- Multidimensional scaling
  - A statistical technique for producing cognitive maps generated from similarity data such as card sort scores.

- Think aloud protocol
  - A procedural task data collection method where “participants are explicitly instructed to focus on the task while thinking aloud and merely to verbalize their thoughts rather than describe or explain them” (Ericsson, 1998).
  - For this study, the technique used for extracting procedural knowledge.
CHAPTER 2
REVIEW OF THE LITERATURE

Mental model overview

The concept behind mental models was first formed by Kenneth Craik, a Scottish psychologist, who in 1943 wrote “the mind constructs ‘small-scale models’ of reality that it uses to anticipate events, to reason, and to underlie explanation” (Craik, 1943, cited in Johnson-Laird, Girotto, & Legrenzi, 1998, Introduction, para. 1). Johnson-Laird, one of the foremost authorities of early mental model theory, defines mental models as “psychological representations of real, hypothetical, or imaginary situations” (Johnson-Laird et al., 1998, Introduction, para. 1). His seminal text *Mental Models* (1983) has been the theoretical base cited throughout the literature. Though definitions and ideas about mental models vary widely, the general concept is that mental models “describe a cognitive mechanism for representing and making inferences about a system or problem which the user builds as he or she interacts with and learns about the system.” (Borgman, 1986, p. 48).

Mental models are not mental pictures or physical models of a system (Johnson-Laird et al., 1998), but rather the underlying knowledge structure that allows an individual to construct their perception of a system or content domain. Holland, Holyoak, Nisbett, and Thagard (1986) describe models as “assemblages of synchronic and diachronic rules organized into default hierarchies and clustered into categories” (cited in Kearsley, n.d., para. 3). These categories are comprised of three types of knowledge: declarative, structural, and procedural. *Declarative knowledge* is the “knowing what”. Individuals can know about something, but not necessarily what to do with it nor why.
Structural knowledge represents the connections, or networks, between declarative knowledge. This is what allows humans to build schemata and mental models for any particular subject. Lastly, procedural knowledge is “knowing how to do” something, utilizing the connections made of knowledge generated through experience (Jonassen, Beissner, & Yacci, 1993). Thus humans can use their knowledge base and perform meaningful actions. Structural knowledge is the key to mental models and how they assist individuals in how they perceive a system or content domain, providing the underlying rules and connections.

Operationalization of this knowledge structure is what provides individuals with the ability to understand a system through possession of a causal model and a runnable model. Knowledge of the system’s components and rules/operations allows an awareness of how the system works (causal), and an individual can mentally operate the system to predict actions and results (runnable) (Jonassen et al., 1993; Norman, 2002; de Kleer & Brown, 1988). When this internal understanding matches that of the actual system design, correct operation can occur (Norman, 1983; Norman, 2002).

A good conceptual model allows us to predict the effects of our actions. Without a good model we operate by rote, blindly; we do operations as we were told to do them; we can’t fully appreciate why, what effects to expect, or what to do if things go wrong. As long as things work properly, we can manage. When things go wrong, however, or when we come upon a novel situation, then we need a deeper understanding, a good model. (Norman, 2002, p. 13)
Mental models are necessary to deal with problems and novel situations (Jonassen et al., 1993; Norman, 2002). They facilitate correct operation or functioning within a specific content domain, but more importantly they provide the capability for predicting what might happen based on certain actions. To merely learn a procedural task or memorize a list of information requires no more than rote rehearsal. To go beyond this and successfully apply or use knowledge in a different way necessitates understanding the fundamental principles and relationships among relevant knowledge so as to formalize potential actions and forecast the results. What happens when understanding is incorrect, as it often is to some extent? “If you are actually doing the task and there is a problem, they (models) let you figure out what is happening. If the model is wrong, you will be wrong too” (Norman, 2002, p. 71). Borgman (1986) agrees that appropriate models are “helpful and perhaps necessary” when they are correct, but that performance will suffer when the model is inadequate. Thus for individuals to solve problems and learn to operate complex systems, they must possess accurate structural knowledge of that system or content domain. “Domain-specific problem solving relies on adequate structural knowledge of the ideas in the domain being explored” (Jonassen et al., 1993, p. 10).

Mental models are messy, ill-defined, inaccurate, and incomplete. They are constantly evolving as individuals encounter new experiences, compare them to what they have previously stored in their models, and then alter their conceptual image accordingly. Johnson-Laird states “cognitive scientists have argued that the mind constructs mental models as a result of perception, imagination and knowledge, and the comprehension of discourse” (Johnson-Laird et al., 1998, Introduction, para. 1).
Similarly, Donald Norman explains “in interacting with the environment, with others, and with the artifacts of technology, people form internal, mental models of themselves and of the things with which they are interacting. These models provide predictive and explanatory power for understanding the interaction” (Norman, 1983, p. 7). Norman provides a few generalizations about his observations from studying mental models (Norman, 1983):

- Mental models are incomplete.
- People’s abilities to “run” their models are severely limited.
- Mental models are unstable: people forget the details of the system they are using, especially when those details (or the whole system) have not been used for some period.
- Mental models do not have firm boundaries: similar devices and operations get confused with one another.
- Mental models are “unscientific”: people maintain “superstitious” behavior patterns even when they know they are unneeded because they cost little in physical effort and save mental effort.
- Mental models are parsimonious: often people do extra physical operations rather than the mental planning that would allow them to avoid those actions; they were willing to trade-off extra physical action for reduced mental complexity. (p. 8)

In other words, people do not carefully organize and file in their minds a complete blueprint for any one system or content domain. Rather they accumulate and assimilate an assortment of concepts, rules, and relationships as they perceive them to make sense at
the time. These can and do change over time, but often original perceptions and beliefs persevere even in the face of contradictory evidence. People are inherently rational beings, but not completely logical as we might assume. Mental models “are often constructed…with a kind of naïve psychology that postulates causes, mechanisms, and relationships even where there are none” (Norman, 2002, p. 38). This has crucial implications for many fields of concern including education, professional training, user-interface design, system design, etc.

**Mental model development**

Advancement of mental model theory has been concurrent with modern approaches in the fields of cognitive psychology and computational research, specifically artificial intelligence and human-computer interaction (Gentner & Stevens, 1983; Jonassen, 1995). The quest to understand what happens in the mind, how individuals process, store and recall information, and how people think about things is at the root of mental model theory development and study.

There is no such thing as a physical “mental model”. The concept is a theoretical construct, so there are several notions about how they are formed and operate. The prevalent opinion about the basis for mental models is that they consist of organized knowledge structures—the concepts, rules, and connections that were discussed earlier. This network of knowledge and its complex relationships is at the heart of the assumptions made by Carley & Palmquist (1992):

- Mental models are internal representations.
- Mental models can be represented as networks of concepts.
The meaning of a concept for an individual is embedded in its relations to other concepts in the individual’s mental model. (p. 2)

Knowledge structure, or structural knowledge, is also referred to as cognitive structure. The general concept is based on the storing of particular chunks of information in a way that is associative and particular to the individual (Jonassen et al., 1993). Individuals inherently try to make sense of the environment around them (Bruner, 1966) and therefore develop their own personal “account” of what it all means. These representations, or mental models, can and do vary considerably amongst individuals.

Information processing theory and schema theory support the notion of a mental structure that people build over time. This structure of specific content domains is what allows people to recall enough concepts, rules, and relationships to process a current situation. Due to limited information processing capability, mental models do not store everything, but rather contain (hopefully) enough information necessary to “run the model” and in effect see what might happen and determine what actions are required. “When individuals construct mental models…they make explicit as little as possible, and they focus on that information which is explicit in their models. Concomitantly, they fail to consider possibilities that lie outside their models.” (Johnson-Baird et al., 1998, Focusing in decision making, para. 2). This underscores the importance of fostering well-developed models that encompass a wide array of experiences and relevant connections in a content domain.

Another way of looking at mental model development is that individuals tend to map new experiences or knowledge onto existing structural relationships (Jonassen, 1995; Gentner & Stevens, 1983). When individuals are confronted with a new
phenomenon, they first attempt to connect it with a prior experience or schema that is perceived to be similar in some way. For example, when teaching students about electricity flow it is most common to explain this concept in terms of flowing water, pressure being analogous to voltage and flow equating to current. Mapping similar characteristics is effective for knowledge retention and enhancement of existing mental models to understand the new concept (Gentner & Stevens, 1983; Jonassen & Henning, 1999; Staggers & Norcio, 1993). Parush (2004) quotes Donald Norman as agreeing that a “metaphor is a stepping stone to a mental model” (para. 19). From the perspective of teaching people new technologies, Brandt (1999) believes that “analogy in particular has been shown to be an effective tool for teaching a conceptual understanding of technology” (p. 42). However, to underscore the point that mental model research is not an exact science, Borgman (1986) investigated user’s ability to understand and operate an online information retrieval system, a new technology at the time, and whether participants would be able to express their view of the system in any sort of analogy such as a card catalog (which had been the analogical model used during the brief training session). They were not able to do so, and several reasons were speculated including insufficient time to develop a model, methodology issues in attempting to extract participants’ conceptual models, and the fact that the relatively simple tasks perhaps did not need a mental model. Thus, in spite of the fact that most researchers agree about the effectiveness of analogy and metaphor, mental model extraction still depends on the situation and method(s) of measurement.

Mental model theory also provides an alternative explanation for human reasoning and inference. As opposed to a step-by-step, train-of-thought process, evidence
shows that humans are simply not that logical. Decision-making is based on scattered, incomplete information in a way that is often incorrectly focused. When confronted by a situation requiring action or other response, humans fall upon their particular schema and mental model that seems to apply. Sometimes their own model is insufficient or inaccurate, leading to erroneous decisions as is evidenced through such situations as Three Mile Island and other human-centered disasters. (Seel, 1999; Gentner & Stevens, 1983; Johnson-Laird et al., 1998; Norman, 2002). Williams Holland Stevens, in Gentner and Stevens’ Mental Models (1983), determined that mental models were a critical component for human reasoning. Forbus (Gentner & Stevens, 1983) studied how people qualitatively reason about space and motion physics. He found that “the models we use…seem to be simpler than formal mechanics and appear to be based on our experience in the physical world” (p. 53). Instead of applying algebraic theorems, people tend to form a visualization of the phenomenon which offers relationships among the objects involved as well as the ability to “interpret these relationships.” Thus an individual develops a structural network of objects, concepts, and relationships that seems to describe and predict a physical phenomenon.

**Eliciting and representing mental models**

The task of extracting mental models is challenging due to several factors. Models are dynamic structures, changing constantly through experiences. Individuals themselves often have difficulty articulating exactly what is “in their head”. As is true of all cognitive psychology, issues of the mind are difficult at best to understand, but over the years a number of approaches have emerged as acceptable methods for eliciting and representing mental models. These generally fall into one of three categories: content analysis
(declarative knowledge), procedural analysis (procedural knowledge), and cognitive mapping (structural knowledge).

Content analysis examines written text to determine the presence and frequency of certain terms, though it typically does not provide any context of meaning or relationships between recurring concepts. Therefore it can reveal what an individual knows about a subject, but not meaningful connections within this knowledge. Content analysis is suitable for determining declarative knowledge. Social science research has generally utilized content analysis, which is generalizable and easy to administer (Carley & Palmquist, 1992).

Procedural analysis observes how an individual performs a given task, focusing on both implicit and explicit factors as to not only how, but also why certain actions are performed. This method provides good insight on the structure of what the individual is thinking throughout the process. However, this is limited to the task itself and not particularly on the general knowledge the individual may possess. It is suitable for eliciting procedural knowledge within an individual’s mental model. Procedural analysis, task analysis, and similar techniques are often employed as a complement to cognitive assessment methods (Carley & Palmquist, 1992; Jonassen, 1995).

Cognitive analysis is ideal for describing knowledge structures (structural knowledge), which includes content as well as relationships between concepts, and comparing them among a group of people. This makes this method ideal for expert/novice comparisons and classroom/training analysis (Carley & Palmquist, 1992).

Researchers have long theorized about how humans mediate their internal knowledge structures and how to elicit that information. Carley and Palmquist (1992)
believe that language is the key to understanding and mediating mental models. They believe that:

- Both the cognitive structure and the text can be modeled using symbols, i.e. concepts.
- The text is a sample of what is known by the individual and hence of the contents of the individual’s cognitive structure.
- The symbolic or verbal structure extracted from the text is a sample of the full symbolic representation of the individual’s cognitive structure. In other words, mental models can be represented as networks. (p. 3)

Others agree that symbol manipulation plays a key role in mental model theory, described by the philosopher Wartofsky as “cognitive constructions in which individuals organize symbols of their experience or of their thought in such a way that they effect a systematic representation of this experience or thought, as a means of subjective understanding” (Seel, 1999, Overview, para. 1). The belief that individuals interact with and represent the world through symbols has received considerable attention over the years from various perspectives (Carley & Palmquist, 1992). The notion that language is central to model formation is the basis for text-based analysis of domain knowledge, where verbal or written text descriptions from subjects is analyzed for frequency of occurrence of key terms and relations to other concepts, resulting in a network diagram showing these relationships (Carley & Palmquist, 1992). This is the basis for eliciting declarative and cognitive (structural) knowledge, discussed next. Procedural knowledge methods will be covered later in this section.
Cognitive analysis elicitation methods

Representing a mental model in terms of the individual’s structural knowledge requires two steps: elicitation of their knowledge structure and representation of this data in some way that reflects that structure. Some of the more common methods for eliciting structural knowledge include pair-wise ratings, card sorts, verbal protocol, and concept maps (concept mapping combines elicitation and representation in one process) (Jonassen et al., 1993; www.tpl.ucf.edu; Jonassen, 1995; Scielzo, Fiore, Cuevas, & Salas, 2002; Evans, Harper, & Jentsch, 2004; Subramani, Nerur, & Mahapatra, 2002). Pair-wise, card sorts, and concept maps are generally scored through quantitative methods, though there are occasional exceptions for concept mapping. Verbal protocols may also be treated similarly to quantitative written text analysis, but is often examined through qualitative procedures.

Pairwise ratings, or similarity ratings, involve having subjects rate the similarity of two paired concepts on a scale of around 7 or 9 degrees, much like a Likert-type scale. A list of significant concepts is compiled, then all possible pairings of these terms are presented to the participant. For each pair, the individual scores either a lower or higher degree of similarity between the two. This method thus determines relationships between concepts and their relative strengths or weaknesses. Pairwise data can then be fed into a statistical package for producing cognitive maps or Pathfinder nets (discussed later) that provide a visual diagram of concepts and relatedness. This method is based on the assumption that cognitive structure can be viewed in a spatial dimension, that “geometric distances between concepts…reflect the psychological proximity of the concepts in the individual’s cognitive structure” (Jonassen et al., 1993). Research has shown similarity
ratings to have a high degree of reliability. Care must be taken, however, in selection of concepts presented in the study to avoid “vague definitions of relatedness” and an overly broad range of concepts that might present different levels of relatedness, confusing the subject. Consequently, selection of concepts should be restricted to a fairly narrow content area (Jonassen et al., 1993).

The card sort exercise presents the individual with a stack of cards which have concepts written on them. They are to sort these into piles that the individual believes to have some meaningful relationship or similarity. Miller (1969) notes that card sort is good “for identifying distances between concepts that are organized in a hierarchical structure” (cited in Jonassen et al., 1993, p. 50). Jonassen also notes that they are also good for identifying “organization of knowledge in a content area and to identify areas of knowledge deficiency” (p. 51). There are variations in how this activity is performed; sometimes the piles are pre-determined, most often the user can create piles as they wish. The researcher may also have participants label the piles they create. With computer software, this activity is easily administered and scored, even allowing for multimedia files such as video, audio, and graphics. Card sort similarity data can be used to generate Pathfinder nets and cognitive maps.

Concept mapping is a modeling procedure that simultaneously elicits and represents structural knowledge. The subject writes concepts and draws labeled links between them to indicate relationship structures. Another variation is when the researcher provides the concepts and asks the participants to generate links. As opposed to pairwise ratings and card sorts, which probe internal knowledge structures that the individual may not even be aware of, concept maps are user generated and therefore can only reveal what
the participant actually recalls from memory. However, one benefit of this is that it reduces researcher intrusiveness, allowing the subject to “explicitly state the relationships they see” (Williams, 1995, p. 3). Concept maps are very popular for educational applications, where teachers have students draw these diagrams to help in assessment of learning, informing further instructional needs, and providing feedback for the student (Freeman & Urbaczewski, 2002; Enger, 1998; Williams, 1995; Kinchin & Hay, 2000). One drawback to concept mapping is the learning curve involved. The “activity of concept mapping also requires instruction and practice to become “fluent” in the act of setting concepts out on paper or a computer platform” (Enger, 1998, p. 2). “Novak (1990) also noted that skill in concept mapping took at least a year to develop” (cited in Enger, 1998, p. 5). Other factors include the lack of consistency in scoring techniques and other issues (Ruiz-Primo & Shavelson, 1996; Jonassen et al., 1997).

**Representation methods**

Representing the data from these elicitation procedures is usually performed through quantitative analysis that produces network diagrams. A variety of procedures have been shown effective, the most popular being multidimensional scaling (cognitive mapping) and Pathfinder nets (Jonassen et al., 1993). Resulting diagrams visually indicate presence of concepts and relatedness among these (see Figures 2.1, 2.2).

Multidimensional scaling, a technique for cognitive mapping, produces a geometric spatial representation of similarity ratings or card sort data. Physical proximity on the graph equates to how the individual connects those concepts in their mental model, so items that are perceived as similar or related in some way are more closely grouped together on the graph. Primarily used in psychology, MDS is designed to help the
researcher discover psychological perceptions underneath the data. As explained by Borg and Groenen (1997), “…when used in an exploratory manner, MDS thus typically carried with it, as an implicit purpose, the search for “underlying dimensions” that would explain observed similarities or dissimilarities…in the kind of exploratory MDS that is typical for psychologists the researcher is interested in discovering psychological dimensions that would meaningfully explain the data” (p. 9).

MDS is only one statistical method for analyzing and producing cognitive maps, but it is the most common. The assumptions behind cognitive mapping is found from Fenker (1975) (cited in Jonassen et al., 1993):

- Information about a topic area is organized and interpreted on the basis of a set of dimensions which represent organizational features of the topic area.
- These dimensions can be represented in $n$-dimension geometric space.
- There are many relationships that can exist among concepts. (p. 62)

Cognitive mapping is effective for comparing knowledge structures of experts and novices, getting a “picture” of how students perceive content domain material, and for other similar applications that can inform instructors and students of class progress. They have even shown to be practical alternatives for classroom assessment instead of basic recall and essay exams (Jonassen et al., 1993). Whereas essay exams are subject to grader bias and subjective interpretation, cognitive maps “provide a relatively objective measure of an individual’s cognitive structure in a given content area” (Jonassen et al., 1993, p. 70).
Some additional advantages for employing cognitive mapping include (Jonassen et al., 1993):

- Effective for assessment of students’ understanding of higher-level knowledge.
- Can measure knowledge structures for a group of people, such as students in a class for instructional assessment and feedback.
- Similarity rating exams are easier to create than multiple choice exams. (p. 70)

![Cognitive map of physical therapy concepts generated by multidimensional scaling](image)

**Figure 2.1.** Cognitive map of physical therapy concepts generated by multidimensional scaling (from Jonassen et al., 1993, p. 64).

Pathfinder nets represent structural knowledge by connecting concepts (nodes) with links. The stronger the relationship between two concepts, as obtained from similarity ratings data, the shorter the link on the resulting graph. The algorithm defines a minimum level of relatedness among concepts, so not all relationships will be graphed; this is different from the dimensional graphing of cognitive maps that provides a more overall picture of connections. Pathfinder nets focus on lower-level comparisons between specific concepts, providing links that are more meaningful. It should be noted that the
diagram does not indicate any hierarchy of concepts, which might be first assumed by looking at a Pathfinder net result.

Pathfinder nets (PFNets) are based on a specific software algorithm found in a program called KNOT (Knowledge Network and Orientation Tool). They are ideal for comparison of structural knowledge and have been used extensively for expert/novice comparisons. They have also been used to compare how people from different backgrounds or opinions view a certain subject (Jonassen et al., 1993).

A few advantages for employing PFNet analysis include:

• Identifying meaningful links between concepts.

• Comparing experts and novices.

• Combining knowledge structures from multiple individuals, such as obtaining an overall mental model from a group of experts.

• Can be easily produced through fully-automated software.
Cognitive analysis via qualitative techniques

Although the methodologies described above are employed extensively in mental model studies, many researchers believe that quantitative, text-based analysis is not adequate and that more qualitative methods are needed to provide a richer, more accurate picture of what is happening in the individual’s mind (Gray, 1990; Campbell, Mack, & Roemer, 1989; Ehrlich, 1990; Galliers & Land, 1987). Some studies have attempted the formulation of qualitative scoring methods for concept mapping. Enger (1998) approached student concept maps to determine such issues as alterations of terminology usage over a period of time, changes in organization of the maps, new knowledge added to the maps, etc. Although such an approach is time intensive and subjective, changes in
knowledge representations between pre- and post-instruction were identified. These results successfully revealed inadequacies in student conceptions of subject matter that could then be used for improving instruction. Kinchin and Hay (2000) also discovered a qualitative assessment technique for student concept maps for the purposes of informing instruction. As opposed to typical quantitative approaches that focus on counting valid and invalid links in a map, they revealed a pattern of how students organized their maps based on levels of structural knowledge. The lowest level of competence is indicated by a *chain* design where concepts are simply linked in series with little or no relationships evident. Somewhat increased understanding of the subject is shown through a *spoke* arrangement, where concepts are linked somehow to the main topic, but little is indicated in the way of inter-related links among the concepts. This feature would instead be representative of a *net* map design. The higher-level structural pattern found in a net-oriented map reflects “deep understanding of the topic” (p. 47) and is the goal for instructional design. Williams (1995) applied both quantitative and qualitative techniques for comparing different instructional approaches in mathematics education. Twenty-eight college calculus students drew concept maps that were then compared with eight experts. The student maps did not match well with the expert maps, showing differences in core concepts and an overall less-structured understanding.

Qualitative methodology is more often applied to verbal protocol data, usually obtained through interviews with subjects and many times in conjunction with other methods of eliciting knowledge structures. Borgman (1986) conducted interviews of subjects who were trained to use an online information retrieval catalog, the objective being to compare conceptual training (providing a mental model during instruction) with
more traditional procedural training methods. After attempting several procedural tasks on the system, the participants were asked to describe their conceptual models of the system; these data were then coded along four dimensions: “completeness of the model, accuracy of the model, level of abstraction, and use of a model in approaching the tasks” (p. 56). It was found that providing the conceptual model during instruction facilitated completion of more complex tasks, but did not provide significant differences for simple procedures. This is consistent with mental model research literature (Parush, 2004; Halasz & Moran, 1983). Coll and Tregust (2003) conducted interactive interviews with high school, undergraduate, and graduate students to extract their models of ionic bonding in various substances. These were compared against a set of established criteria that represented expert opinions of appropriate bonding models. They found that aside from predictable higher-level structural awareness with the older students, all subjects tend to “use simple mental models” (p. 464). Their recommendations stated that instructors need to “provide stronger links between the detailed nature of a model and its intended purpose” (p. 464). To compare mental models of computer networks between pre- and in-service teachers, Levin, Stuve, and Jacobson (1996) employed qualitative methods using data from surveys, interviews, and observations of task performance. The study revealed that subjects applied their individual mental model of computer networks to actual performance of various tasks. The model type and structure, however, varied widely among the subjects, with a more structured model and approach exhibited from the experts (in-service teachers). From a methodological perspective, it should be noted that although they use the term “in-depth interviews”, these were actually in context of a task analysis procedure and think-aloud protocol, discussed later.
Procedural analysis techniques

Jonassen (1995) agrees with Carley and Palmquist that mental models are mediated through language and symbols and that they can be represented as “networks of concepts.” His 1993 text Structural Knowledge is based on this assumption and describes a multitude of methods for eliciting knowledge structures through text-based analysis. He also states, however, that this alone is insufficient for completely representing an individual’s mental model. “Mental models are more than structural maps of components. They are dynamic constructions…Mental models, like all knowledge, must be inferred from performance of some sort” (Jonassen, 1995, sect. 3.1). Fiore, Cuevas, and Oser (2003) also believe that a variety of techniques are needed since any single method is insufficient for an accurate assessment of declarative, procedural, and structural knowledge. Alternative methods that seek to extract procedural knowledge include behavioral task analysis, cognitive task analysis (think-aloud protocols), user drawings, and interviews (Gray, 1990; Johnson, 1988; Jonassen, 1995).

Behavioral task analysis is useful for observing how an individual completes a task, but is not suited to discover the mental thinking behind the action (Johnson, 1988). This method is often used for instructional design applications whereby skills relevant to the instructional objectives are broken down into discrete components in order to better present the procedures to learners (Jonassen, Tessmer, & Hannum, 1999). A variation of behavioral task analysis is to ask experts how they complete specific tasks. However, Norman (1983) notes that observing and describing behavioral activities is more effective at eliciting mental model performance than attempting to provoke explanations simply because as expertise develops the individual is not consciously aware of everything that
is involved in completing a task. Numerous studies of computer usage have employed task analysis such as Levin et al. (1996) who studied teachers’ perceptions of the Internet (discussed earlier), Gray (1990) who examined mental model construction during hypertext navigation, and Johnson (1988) who observed subjects’ operation and troubleshooting of complex equipment.

Johnson actually employs a variation of task analysis called *cognitive task analysis*. Cognitive task analysis typically involves a think-aloud protocol where the participant verbally states how they complete a task or solve a problem while they perform that task (Johnson, 1988; Ericsson & Simon, 1993). The objective is to have the subject simply say aloud what they are thinking as they work and to not ask them to attempt explaining their actions. Digging deeper into *why* changes the cognitive process in an undesirable way for mental model extraction; the researcher wants only to see into the mind state as they perform, not as they analyze their performance. Both describing and explaining alters their thinking, which then affects the performance flow data.

Obtaining an individual’s mental conception or image of how a system works—their “runnable model”—is related to both their structural and procedural knowledge of the system. What they know about the system and how it works is represented in their internal mental image. Capturing this image can simply include having subjects draw a diagram or graphic that attempts to show what they “see” in their minds (Gray, 1990; Glynn, 1997; Coll & Treagust, 2003; Jonassen, 1995; Gott et al., 1986). However, Jonassen (1995) cautions that the act of drawing itself requires specific skills that may impact the accuracy of drawings that subjects attempt to make. Another option is to conduct an interview and have the subject verbally explain their image of the
system. This verbal protocol can be analyzed and coded for comparison with the “real” system model or with that of experts (Taylor & Tversky, 1992).

Most researchers use a combination of some of these methods, including quantitative and qualitative approaches. Gray (1990) employed exclusively qualitative measures, selecting three techniques (think-aloud, behavioral, user drawings) to study how participants used and perceived a hypertext environment. Jonassen and Henning (1999) describe an approach to comparing expert and novice problem solving using several methods: pair-wise data for Pathfinder net diagrams (structural knowledge), think-aloud protocol for problem solving (procedural knowledge), interviews for obtaining participants’ mental images of the system, and requesting participants to generate a metaphor or analogy that they relate to how the system works. Coll and Teagust (2003) studied student perceptions of ionic bonding through interviews and user drawings to determine how they “see” bonding structures. Johnson (1988) used behavioral and cognitive analysis methods to study technical troubleshooting techniques. Behavioral analysis identified how the technical equipment operated, while cognitive analysis provided insight into how individuals sought solutions to faulty equipment. Enger (1998) provides an example of using one data set (concept maps) for both quantitative and qualitative analysis. The key is to select methods appropriate for each situation that, when combined, provide as clear a picture as possible to meet the researcher’s objectives.

**General applications of mental model research**

Mental models are perhaps most often thought to be relevant to the study of computer systems and design/use of technical devices. De Kleer and Brown (in Gentner
& Stevens, 1983) are often cited regarding their examinations into how people build an understanding of mechanistic devices. Borgman (1986) investigated how users conceived the virtual structure of an electronic catalog system. Gray (1990) similarly examined mental model construction while users navigated through a hypertext information system. Norman (1983, 2002) has provided a useful terminology for comparing the models between the designer of a device, the user of a device, and the device itself. A designer has a model in mind when developing a device, though sometimes the implementation of the device itself may end up presenting a different conceptualization (system image) to the user. The user then develops their own conceptual model of how it works. The objective for designers (hardware and software) is to map the original model as closely as possible to the actual device and ensure that this model is conducive to fostering an accurate user model. “The mental model of a device is formed largely by interpreting its perceived actions and its visible structure” (Norman, 2002, p.17).

Though technology-related disciplines represent a significant share of mental model research, mental models are generally agreed to be applicable to most any content domain (Jonassen, 1995). Coll and Treagust (2002) compared high school, undergraduate, and graduate students’ mental models of ionic bonding. Borges and Gilbert (1999) examined mental models of electricity, one of many studies on people’s conceptions of physics concepts. Buzydlowski (2002) applied mental model elicitation techniques to determine how experts in the field of humanities literature compared and related well-known authors. Evans et al. (2004) compared models of team members to determine how to improve interaction and foster like-minded decision-making processes for military purposes. All of these studies revealed differences in structural knowledge
where more advanced students or skilled performers possessed a more highly organized, interrelated awareness of concepts in the content domain.

**Mental model research in education and training**

Mental model techniques have been applied to various aspects of education and training. Measuring outcomes of instruction, either for comparison with experts or for official assessment, has been employed extensively. One of the most popular techniques for assessing students’ conceptions of content matter is through concept mapping, a method for representing knowledge structures. Williams (1995) examined varying instructional approaches to the teaching of calculus by comparing concept maps of instructors and students, finding that instructor and student maps varied considerably in core concept organization and awareness. Freeman and Urbaczewski (2002) used concept mapping as an enhancement to traditional assessment by examining the mental models of undergraduate students in a telecommunications course. By having undergraduate students draw concept maps at three points during the semester, a positive progression of development was easily seen in student mental models of the subject content, especially when compared with a referent expert model. Using different techniques, Scielzo et al. (2002) also dealt with assessing learners’ developing mental models in a training environment to determine how accurately they represented the knowledge structures being acquired. Quantitative methodology concluded that accuracy of one’s mental model directly correlates to performance ability, knowledge acquisition, and instructional efficiency.

Tools used for representing mental models can also be effective throughout the instructional process by helping learners gradually build appropriate knowledge.
structures of the subject matter. Richard Mayer has advocated that instruction can be
designed specifically to help learners construct meaning and therefore develop mental
models of the subject, even without “hands-on” opportunities (1989, 1999). His SOI
model of learning (selecting, organizing, and integrating) specifically directs learners’
attention to the significant concepts and shows how they relate, the goal being to help
them form a mental picture of this organization that is encoded into long-term memory.
This is provided through specific diagrams and other materials carefully designed to
make such connections (such as cause and effect) explicit. Enger (1998) qualitatively
examined concept maps drawn by 7th graders before and after an instructional unit and
documented increased conceptual understanding after instruction. Similarly, Kinchin and
Hay (2000) describe a qualitative approach to evaluating student concept maps, having
students begin diagramming maps early in instruction. These are used as an ongoing tool
to help them self-monitor and develop their knowledge base as well as provide the
instructor with feedback on how students perceive content. Clement and Steinberg (2002)
detail a learning aloud case study featuring a student who was presented with an electric
circuit problem to solve. Interactive discussion between the instructor and student helped
the individual’s mental model evolve into a more accurate and appropriate knowledge
structure in the content domain. Macklin (2003), in a discussion paper directed toward
librarians, discusses how to apply constructivist learning concepts when developing
instruction for academic library usage. His ideas are based on David Jonassen’s
constructivist learning environments (CLEs) where various experiences and problem
solving processes work together to build each learner’s particular mental model. For
complex problems, the issue is helping students become aware of their own thinking
(metacognition) to determine how accurate their model is for solving problems (finding information in the library, for example), thus enabling them to successfully confront future questions as they arise.

Another approach, though not well-grounded in the literature, is to have students simply draw their mental models as they attempt to diagram what they see in their mind about the subject (Glynn, 1997). These were then used more as a diagnostic tool rather than official assessment, thereby providing feedback for instructional decisions and strategies. Part of a more formal study conducted by Gray (1990) involved having participants draw diagrams showing their personal conceptions of how a hypertext information system was structured. This was an attempt to allow participants to articulate what was in their heads, which can be difficult to accomplish verbally (Borgman, 1986). In this example, most early drawings resembled linear document (book) models, but as users spent more time navigating through the electronic information their models gradually morphed, though in various directions. The virtual nature of a hypertext information system makes it difficult for new users to comprehend how it is structured, therefore the drawings were quite revealing in how different individuals perceived how such a system operated. Methodologically, the diagrams were a complementary approach to a procedural think-aloud protocol method, which enabled a clearer overall picture.

Fiore, Cuevas, and Oser (2003) manipulated the inclusion of diagrams during training to determine resulting effects on task performance and mental model development. The question of whether performance improvement would result by providing a design model of a system during instruction has also been studied, for example by Kieras and Bovair (1984) who compared providing system diagrams against
rote procedural training during instruction to see if they helped users learn how to operate a simple control panel. They found that the model group “learned procedures faster, retained them more accurately, executed them faster (and) simplified inefficient procedures…” (p. 255). Studies from Cuevas, Fiore, & Oser (2002), Gyselinck and Tardieu (1999), and Mayer (1989) support this notion.

Though much research has shown that providing learners with appropriate system or conceptual models during instruction can be beneficial, the presence of a mental model may not be helpful for simple tasks and may possibly even impede performance (Borgman, 1986; Halasz & Moran, 1983). The power of a mental model is to predict the performance or outcome of a system or action. This requires “running the model” using data imbedded within the individual’s knowledge structure. The model then extrapolates this information to form conclusions. For rote procedural actions that are straightforward, this process can get in the way (Parush, 2004).

Kieras and Bovair (1984) discovered that though providing system models to learners can be helpful, there are a few issues to keep in mind:

- The model must support inferences about how the system works in detail. Thus general metaphors or analogies do not support the user to be able to infer performance or operation of the system.
- Any relevant information on how the system works (as opposed to how to work it) does not need to be very detailed or in-depth.
- If operation of a system does not require any inferring from basic information, then a model is not necessary. Using an ordinary telephone is an example where
people do not need to really understand the system model in order to operate the device.

So, providing a system or conceptual model during instruction has been shown to improve performance, but there are issues to keep in mind. Mental models are useful in that they allow an individual to infer (predict) causes and results of a system. Therefore, instruction must provide enough information to allow inference of the system operation, but not necessarily all the detail (and yet not lacking in enough information to foster the ability to “run” their model).

**Expert-novice mental model comparisons**

Expertise is developed over time through extensive experiences with novel situations; this accumulation and assimilation of new knowledge and connections in turn facilitates a highly advanced network of structural knowledge that novices simply do not possess. Johnson (1988) cites several studies that indicate that it is this knowledge structure, or organization, that determines expert versus novice performance (Egan & Schwartz, 1979; Chase & Simon, 1973; Anderson, Spiro & Anderson, 1978) and that mental models are fundamental to expert development (Kuipers & Kassirer, 1984; Kieras & Bovair, 1984; Bouwman, 1983; Lajoie, 1986; Logan & Eastman, 1986; White & Frederiksen, 1987). “Expert knowledge consists of increased connectedness among critical concepts…” (Fiore et al., 2003, p. 193). It is important to note that the emphasis is on increased connections, not just new knowledge. The network that experts develop, implicitly as well as explicitly, is the key to their ability in making decisions and solving problems efficiently. Experts approach a novel task or problem by identifying relevant
components and developing solutions while bypassing irrelevant information. Novices, while they may “know” some amount of information (declarative knowledge) about the subject, and may have even learned some procedural tasks, do not yet possess meaningful connections that facilitate such a process; therefore they must consider and process one bit of data at a time regardless of its actual importance, relevance, or sequence (Johnson, 1988; Landa, 1999). “Beginner’s knowledge is spotty, consisting of isolated definitions and superficial understandings of central terms and concepts. With experience, these items of information become structured, are integrated with past organizations of knowledge” (Glaser, 1989, p. 272). Villachica et al. (2001) cites other studies that have identified differences in knowledge structure sophistication between experts and novices. Some of these involve the game of chess, (De Groot, 1978; Chase & Simon, 1973; Reingold, Charness, Pomplun, & Stampe, 2001) others in the field of physics (Chi, Feltovich, & Glaser, 1981). In one of their own studies, Villachica et al. (2001) sought to elicit expert knowledge structures in the field of human performance technology. The objective was to see “how HPT experts organize their knowledge of the discipline” as well as to compare how HPT knowledge was organized differently between experts and novices (p. 437).

The most obvious application for expert-novice research is for informing instructional practice. Since the intent of most educational programs is to help learners begin to think and operate in a more mature way, i.e. like professionals in a particular field, it is first necessary to understand what experts know and are able to do. This proves more complex than might be first imagined. Typically when designing instruction educators rely on experts’ own ideas, whether from an author’s textbook, journals in the
discipline, or the knowledge of the instructor himself, to determine the content and structure of a particular subject (Villachica et al., 2001; Fiore et al., 2003). However, understanding exactly how experts conceive of a particular subject and apply their expertise in solving problems and completing tasks is not easily nor completely accurately accomplished by relying on expert self-description. Often the individual does not consciously realize why or how they understand or do certain things. Some knowledge is implicitly learned, making it impossible for an individual to articulate. Other knowledge has become automated to the extent they no longer consider it during performance. Mental model elicitation and representation have shown to be effective at getting the “meat” of the mental knowledge structure an expert possesses (Villachica et al., 2001).

Therefore, it is logical to then take this and do two things: 1) compare to the mental models of novices, perhaps students or beginners in the field, to determine gaps and inconsistencies, and 2) to develop instructional materials that attempt to transfer expert knowledge structures to learners. Expert-novice mental model comparisons have been performed extensively in a variety of educational and training situations.

Faculty at the University of Illinois Urbana-Champaign needed to determine how in-service and student teachers conceptualized computer networks as part of an effort for improving technology literacy (Levin, Stuve, & Jacobson, 1996). They specifically wanted to know how teachers used computer networks in their everyday activities and what their mental models were for networks and related tasks. As described earlier, subjects actually did apply their own particular models of networks to task performance; these models were more structured for the experts than for novices. The outcomes were
used to examine predominant models that might emerge which could then be incorporated into the educational program. Williams (1995) compared concept maps of instructors and students in two aspects: instructor vs student comparison as well as two different methods for calculus instruction. Results indicated that students’ mental models of the material were much less structured and sophisticated and lacked core concepts that instructors’ models showed.

Troubleshooting, operating complex systems, and solving novel problems require individuals to possess rich, accurate mental models of the task and content domain (Jonassen & Henning, 1999; Gott, Bennett, & Gillet, 1986). Jonassen and Henning (1999) believe that mental models cannot be solely represented by content analysis of structural knowledge, as many elicitation techniques are based, but also by considering actual performance as an individual applies their knowledge in a meaningful way. They therefore have investigated mental models of experts and novices through a series of techniques targeted at content (presence of concepts and relationships) as well as procedure (how content is applied in situations). Although not unique in combining content and procedural methodology for analysis, this study in particular contrasts with many mental model investigations where typically researchers have utilized one or two methods for extracting participants’ models. Jonassen and Henning employed three techniques aimed at triangulating a rich model outcome. The results provided significant comparisons that showed that the participants who performed the slowest on the procedural tasks also generated Pathfinder nets featuring “fewer links, fewer levels…(that were) less integrated than that of the fastest performer” (p. 39). Similarly the slower performers could not provide well-structured descriptions of how the system
worked (system images). This provides a much more complete picture that reinforces the results from each of the individual exercises.

Scott Johnson at the University of Illinois Urbana-Champaign has also investigated expert-novice troubleshooting performance (1988). He compared cognitive and behavioral (performance) knowledge of experienced and novice service technicians in the diagnosis of faulty complex equipment. The result, typical of expert-novice comparisons, is that the novices did not possess the rich structural knowledge that facilitates efficient and accurate analysis of a problem situation.

Fiore et al. (2003) studied the effects of taking expert models, developing corresponding diagrams, and incorporating them into instruction to “encourage the acquisition of knowledge structures more similar to an expert model” (Fiore et al., 2003, p. 188). They found that this was effective in terms of helping participants to “accurately draw connections”, and that the model diagrams “facilitated performance on measures of integrative knowledge” (p. 185). However, results did not seem to indicate performance improvements of declarative knowledge.

Another example of using experts to derive structural knowledge models that can be used to inform instructional design is from Diekhoff and Wigginton (1982). Working with college faculty in psychology, history and systems, and statistics, they employed multidimensional scaling to produce cognitive maps that were then used in class instruction. Test scores for students in these classes were superior to those in classes using traditional instruction.

The issue of team members and their ability to anticipate each other’s actions during a task was the focus of a potential expert/novice study designed to enhance
military training efforts (Evans et al., 2004). Their hypothesis was that team members who possess similar mental models should perform at greater levels as a unit. This article merely describes how such a study would be conducted and does not include actual data and findings.

**Implications for the current study**

The literature supports a study design comparing mental models of experts and novices for the purpose of evaluating and improving instructional design. Although a variety of techniques have proven effective, it is clear that an overall picture of an individual’s mental model requires a combination of approaches that seek to elicit declarative, structural, and procedural knowledge of a particular content domain. Therefore, this particular study investigates the content that professional recording engineers and students possess about recording systems, how they relate and organize various terms and concepts significant to the subject, and how they then apply that knowledge to actual tasks common for recording procedures. As previous studies in the literature have discovered, results should reveal that professional engineers have developed over time a more structured, hierarchical “picture” of recording system concepts and procedures. While seniors in the college degree program have learned a great deal, presumably their results will be less structured and even inaccurate in many instances. This is normal for novices, though the objective is to determine how much of this can be corrected, or at least improved, through revised instructional approaches. The benefit of this study is the potential for revealing such inadequacies and issues so that corrective measures can be pursued.
CHAPTER 3

METHODOLOGY

Participants

There are two participant categories for this study: expert and novice. Potential subjects that qualified as experts were professional recording engineers who work full-time in the music industry. Criteria for selection required that they have several years of experience in studio recording on some variety of recording system types. Mastery of Pro Tools was not considered essential for all experts, as the objective was to determine how an experienced engineer approaches task performance on such a system, not how quickly they can complete the task. Ideally, a mixture of expertise between the experts was considered desirable so as to provide a diverse comparison of mental model approaches. Five local engineers were identified as potential candidates in order to determine a final selection of three experts. These individuals received no financial compensation, but as expected they were all quite interested in participating for the sake of helping inform the educational program at the college.

The three participating experts possessed many years experience in the recording industry, though in differing aspects of engineering. Two of them had predominantly analog backgrounds and experience and were also quite familiar with computer-based systems. One of these, however, had never worked with the Pro Tools system before and therefore made an excellent candidate for observing how he might apply his particular mental model to figuring out the task procedures in an unfamiliar environment. The third engineer had worked almost exclusively in a Pro Tools software-based environment since graduation from college; her educational program provided both analog and digital
experience, so there was some limited background similar to the other professionals. Her work experience, though, had been very procedural, even rote-oriented, with little variation and almost no exposure to other recording systems other than Pro Tools.

One expert taught for the first time a senior-level capstone course at the study site during fall semester 2004. All potential students for the study were in this class where they worked on various recording techniques and issues. They used primarily analog recording systems with limited work in Pro Tools. While the brief Pro Tools experience might have served as a refresher from an earlier class, it was insufficient in terms of providing in-depth instruction and practice on this system.

Novice participants were selected from students enrolled full-time in a small, private four-year liberal arts college in south-central Pennsylvania. All of the students were seniors majoring in a recording engineering degree program, administered in the music department. Student selection involved identifying individuals who had generally performed well academically throughout the program while also representing a range of competency in their understanding and performance with recording systems. Poor academic performers were not considered as their lack of progress in the program might be attributable to any of a variety of reasons other than instructional and would therefore potentially skew the results. The objective was to identify “typical” students who seem to have performed well overall, but yet provide a range of capability and personal interests. Ten students were identified and approached as potential participants for the study. The purpose of the study and a general description of the procedure were explained; this was conducted on an individual basis in order to better personalize the importance to each student and was deemed more effective than having a general group presentation. Student
participation was entirely voluntary; they were not being recruited from a particular class context, so no grade points could be used. They were paid a small stipend for their time, but many of them were interested in the process and willing to help the investigator. A subject pool of seven students, the target number for the study, agreed to participate. The approximate age range of the students was 21-22 and provided a mix of five males and two females, fairly accurately reflecting the gender balance in the degree program. All participants completed consent forms as required by the Office of Regulatory Compliance at Pennsylvania State University as well as the Institutional Review Board at the study site (see Appendix A).

These students had completed sophomore, junior and senior-level courses which utilized a combination of Pro Tools (and other software applications) and analog recording equipment for class and individual projects. Students have significant hands-on experience in this degree program, so by the time this group participated in the study they were at a functional level of competence, probably more so in the analog studios than on Pro Tools. Thus while they were still considered novices at this stage, they had had the opportunity to develop experiential knowledge that helped develop their mental models over the course of three years.

**Context of the study**

The researcher is a full-time faculty member and director of the degree program in which the student participants were enrolled. In addition to knowing the students throughout their four years at the college, the researcher is the instructor for two sequential, required courses in the degree program, MRT 277 Recording Engineering I and MRT 278 Recording Engineering II. These courses are taken at the sophomore level
and are the first recording courses experienced by majors in the program. Primary objectives for these two classes focus on providing a foundational knowledge base that facilitates their more advanced experiences throughout the rest of the program. More specifically, the instructional goals are to help students generate appropriate declarative and structural knowledge of the field and of recording systems as well as develop an appropriate generic system model of how all recording systems work. The idea is for them to take these core operational characteristics, common to all systems, and be able to “run the model” as they attempt to learn a new and different type of recording system. During the first recording class they learn to record on a large-format analog recording console, the Sony MXP-3036. The console is the central controller, so to speak, of a recording system and is where the engineer routes signals, processes them in some way, and mixes them together during music production. It is a fairly complex piece of equipment, and it takes quite some time for novices to get a sense of what is happening and how the operational patterns work. The challenge is to help them not merely learn rote procedures that happen to work in a particular situation, but rather the underlying operating principles, i.e. the generic system model.

In the second class they then take this foundational knowledge and learn a similar, but more complex recording console, the Trident Vector (see Figure 3.1). This console is larger than the Sony, has more options and controls, and is also more flexible in its operation which makes it more confusing to novices (see Figure 3.2). A good mental model of how both consoles work greatly benefits students and is one of the primary issues for the class.
Figure 3.1. The Trident Vector 432, a large format analog recording console.
Figure 3.2. The left diagram is the equalizer (tone control) section from the Sony console, the right from the Trident. Note how more complex the Trident controls are, including more knobs and switches. This is representative of how the entire console compares to the Sony.

Making the jump from the Sony to the Trident is significant, but the real test is applying their knowledge of recording systems to such devices as digital consoles and software-based recording systems like Pro Tools. These systems, though based on a traditional analog paradigm represented in the Sony and the Trident, are by their nature virtual and menu-driven. For the professional engineer this flexibility, along with the multitude of options they feature, provides powerful production capabilities, but for novices it is simply overwhelming. The virtual interface provides little visual cue as to layout, functionality, and signal flow that is much easier to see on a physical, analog console.

This step in their educational experience occurs during their third major recording class. The expectation is that students should be able to apply their knowledge of recording operations to the software. However, anecdotal evidence from the faculty member who teaches this course appeared to indicate students were not transferring appropriate system models, apparently learning just enough procedural instructions to accomplish tasks as needed. Essentially they often seemed to be throwing out the analog experience with the mindset that software-based recording systems are an entirely different realm. Determining whether this was the case and what might be really happening therefore became the impetus for the purpose and design of this study.
Research design

The review of the literature of mental model studies clearly suggests the need for a combination of methods for the purpose of eliciting and comparing mental models. While structural knowledge is often extracted and represented through quantitative data collection and analysis, eliciting complementary procedural knowledge, along with the cognitive thought processes involved, requires actual performance by participants (Jonassen, Beissner, & Yacci, 1993; Jonassen, 1995). This type of procedure is best served through think-aloud protocol and rich description of the activity (Ericsson & Simon, 1993). Additionally, a qualitative examination of the structural knowledge data provides a more in-depth, potentially more meaningful insight into what the data might represent. Thus a mixed-method research design was proposed, combining quantitative and qualitative methodology. Though this approach is a relatively recent trend in research design, it has been accepted as necessary and desirable for the purpose of triangulating results from different data and procedures (Greene, Caracelli, & Graham, 1989). Bransford, Brown, & Cocking (1999) discuss the development of an interdisciplinary approach to studying the mind and learning and notes that “the introduction of rigorous qualitative research methodologies have provided perspectives on learning that complement and enrich the experimental research traditions” (chap. 1, Development of the science of learning).

This current study can be characterized as an exploratory, descriptive investigation that sought to reveal existing knowledge structures of experts and novices in order to inform and improve instructional practice.
Summary of procedures

A flowchart outline of the study procedures is included in Figure 3.3. Detailed explanation is provided throughout Chapter 3 as it pertains to each of the two study exercises.
Figure 3.3. Flowchart of research procedures.
Data collection and analysis methods

Structural knowledge elicitation

Among the several methods that have been used for eliciting structural knowledge, the one that seemed best suited for this study is the card sort. Participants are presented a list of concepts that are to be sorted into different piles according to their own perception of similarity or other association. This represents their structural organization of the concepts deemed important and relevant to the content area being researched. Cognitive maps derived from card sort data are effective for comparisons between groups, such as between the students and the professional engineers. This is one of the most common applications for cognitive mapping where assessing student knowledge of a particular content area is the objective. Their maps are then compared to an expert map to determine inaccuracies and gaps in their knowledge structure, thereby informing instruction (Jonassen et al., 1993). This is directly relevant to the goals of this study.

Card sorts allow examination into the meanings of words as perceived by the individual (Miller, 1969). The resulting stacks, or categories, indicate the relationships the individual sees among the concepts, and these can then be compared with other individuals or groups. Harper, Jentsch, Berry, Lau, Bowers, and Salas (2003) cite two studies (Fiore, Cuevas, & Oser, 2003; Fiore, Cuevas, Scielzo, & Salas, 2002) where the data consistently shows that “similarity to an expert model is positively related to performance on measures of knowledge acquisition” (p. 579). Therefore the card sort procedure has been utilized extensively for expert/novice comparisons and is good for identifying “organization of knowledge in a content area and to identify areas of
knowledge deficiency” (Jonassen et al., 1993, p. 51). Thus while it provides a snapshot of an individual’s cognitive structure, it is also suitable for discovering gaps and misperceptions in students’ understanding of subject area content. Chi, Hutchinson, and Robin (1989) conducted a card sort as part of a multi-method study and discovered that the sort task clearly revealed a higher-level hierarchy as opposed to merely indicating basic similarity among concepts.

Research on the card sort exercise indicates that it is a stable, reliable method. Studies by Evans, Hitt, and Jentsch (2001) and Tessmer, Perrin, and Bennett (1998) all concluded that repeated card sorts by individuals produce reliable, consistent outcomes. Validity concerns have been raised by some researchers; however, many believe that it is still an effective tool for assessing how an individual views conceptual relations (Harper et al., 2003; Fiore, Cuevas, & Oser, 2003; Jonassen et al., 1993), especially if implemented as one component of a multi-method approach. Cheatham and Lane (2002) found that card sorting predicted user performance better than other methods. A comparison study by Fiore, Fowlkes, Martin-Milham, and Oser (2000) showed card sort and similarity ratings to be equally effective. Recommendations regarding the number of subjects desired for satisfactory results generally range between five and thirty (Card sorting, n.d.; Nielsen, 2004; Maurer & Warfel, 2004). The primary issue is balancing accurate, usable data with the effort required to administer and process large numbers of participants. Nielsen describes how a pool of fifteen participants provides an adequate correlation when compared with a much larger group, though general results can be obtained with as few as five subjects.
There are variations on how the card sort is administered, usually consisting of whether to control how the subject sorts the cards and whether they can add their own concepts and/or piles. These options are situation specific and do not impact the integrity of the exercise (Team Performance Laboratory, Card Sort, 2002). The most efficient and reliable method for administering a card sort task is through computer software. This allows automation and accuracy of the exercise procedure as well as simplified data scoring and analysis. The Team Performance Laboratory at the University of Central Florida has developed such an application called TPL-KATS (2002). This program supposedly runs on most any computer platform that contains a virtual Java runtime environment, though neither of the two Apple Macintosh versions worked at the time this study was being developed. The investigator sets up the concepts and categories (stacks), including options that control how the user (subject) interacts with the file. Then the program is set to experiment mode for subjects to complete the sort exercise. If desired, individuals can be required to log-in so the program can track their activity and data, then afterwards the investigator can run various scoring functions and output data for direct analysis as well as for statistical processing into cognitive maps or Pathfinder nets. Though many methods for scoring card sort data have been implemented, the most common is a simple “hit or miss” comparison, where all concepts in the exercise are paired to see how many pairs were in the same pile. This is then examined to determine characteristics of the sort, such as whether each individual sorted concepts based on content structural features or hierarchical relationships.

Lastly, card sorts are ideal for eliciting structural knowledge because they require little pre-requisite skills for subjects to complete. In comparison to concept mapping and
drawing, card sorts are easy to perform and provide effective representations of cognitive structure (Jonassen et al., 1993; Enger, 1998; Novak, 1990).

*Card sort exercise setup*

Setting up the card sort exercise for this study consisted of several steps including 1) obtaining and learning the TPL-KATS software to administer the exercise, 2) developing an appropriate list of concepts to be included in the sort, and 3) determining how best to score and process the results for comparison.

The TPL-KATS software application is available for free download from the University of Central Florida’s Team Performance Laboratory website (www.tpl.ucf.edu). It is a very simple program to set up and operate both for the researcher and the participants. The list of concepts that the investigator creates appears along the upper-left side of the screen (see Figure 3.4). Pre-determined piles (categories), if set by the researcher, are located in a list along the bottom-left side of the screen. The participant simply drags each pile name to the main workspace in the middle of the screen, then drags any of the concepts to one of these piles. Concepts can be moved between piles as the subject works through the exercise. The piles can also be determined by the participants if so desired. Instructions can be given either by the investigator or within the software through a tutorial and an online help file, though a brief demonstration was sufficient in this situation. Intervention by the investigator was minimal, reducing external factors that might have affected the outcome.
After several trial configurations, it was deemed best for this study to provide a limited set of concepts, where the participant could not add new ones to the list. This was to reduce additional variables between participants that would make it more difficult to make comparisons. Subjects had to create and label their own categories (piles). Otherwise there were no restrictions—participants were asked to organize all 20 concepts into piles in any way that made sense to them. The objective was to put concepts that seemed most similar into the same pile, and they could create any number of piles they wished. This approach, known as a free sort, seemed best to facilitate each individual’s particular knowledge organization without constraining and influencing the results. The list of 20 recording engineering concepts was determined through a series of resources. First, the investigator’s own personal knowledge of the subject matter was sufficient to develop a fairly comprehensive list of terms. Next, audio recording texts, the Pro Tools operating manual, and 3rd party books on using Pro Tools were examined for
additional/alternate terminology that reflected the content and relevant procedural tasks involved. Finally, consultation with other audio educators and professionals provided different perspectives and content validity that informed the investigator’s judgment. Of the twenty concepts, seven were graphics taken either from Pro Tools screenshot excerpts or photos of analog hardware, each focused on a particular function or control.

The card sort software was installed on an IBM Thinkpad laptop running Windows 2000 Professional. Participants had a choice of the computer’s tracking device or an external mouse or trackball. No other equipment was required for this procedure.

Instructions to each participant involved explaining what a card sort is, what they were to do with the cards, and demonstrating how to operate the software in order to organize and name the cards. It was emphasized that this was not an exam of any kind and that there were no correct answers to strive for. They were encouraged to take as much time as they wished; each participant took about 20-30 minutes to complete both sort tasks.

Each participant was asked to complete two sorts using the same list of cards. They could decide whether to change anything for the second sort, but it was made clear that it was completely up to them. Sometimes during a sort one finds different, equally viable ways to organize the terms. Providing a second opportunity allows the participant to pursue a different mode of thinking if they wish. Once each sort was completed the file was saved and a screenshot of the sort work area was exported using the KATS software “Export as JPEG” command. These were printed during analysis for visual comparison as described later.
Analyzing card sort data

“Analyzing card sort data is part science, part magic. Analysis can be done in two ways: by looking for broad patterns in the data or by using cluster analysis software” (Maurer & Warfel, 2004, Analyzing the results). With this in mind, it is desirable to analyze the card sort results by reviewing all the resulting piles and looking for apparent patterns in how participants organized the terms; a statistical analysis is also helpful for providing a quantitative perspective. For this study, each participant created their own pile categories, so exact one-to-one comparisons were not possible. However, general organizational structures would most likely be reflected among the resulting sort files, and it was expected that similar categories would be apparent, though perhaps more so with the experts’ sorts.

Although the investigator’s original intent for statistically analyzing the sort data was to compare multidimensional scaling and Pathfinder network procedures, it was discovered that the current version of PC KNOT software, used for producing PFNets, is an old software application that ran only under DOS-based systems. Discussions with the developer confirmed this, so that option was discarded. Therefore all statistical processing of the sort data was performed with multidimensional scaling procedures using the SPSS software application.

Scoring the card sorts was accomplished by selecting the “Score” function within the software for each sort. This generated three data files that were used for analysis. One of these is a list of all piles for that particular sort along with the pile names given by the participant. The second is a list of all twenty cards along with the names of the piles where each was placed. The third file is a correlation matrix that lists each possible
pairing of all cards included in the exercise and which pairs were included in the same pile. A “1” next to a particular pair of cards indicates they were placed in the same pile whereas a “0” signifies they were not.

Of the three output data files from the card sort software, the one used for analysis was the correlation matrix. This binary data for each case was entered manually into a separate Microsoft Excel spreadsheet, creating a 20x20 lower triangular data matrix (see Appendix B for an example). At this stage the binary configuration was such that a “1” in any cell indicated that those two cards were in the same pile while a “0” indicated they were not. For processing with SPSS, however, it was found that these needed to be inverted so that a “0” indicated a positive pairing. This is known as a dissimilarity matrix. Therefore, a simple formula was employed in Excel so that when the data was entered it was automatically converted (=IF(B2=1,0,1)). These resulting matrices were then imported into SPSS and saved as SPSS data format files.

SPSS is a sophisticated statistics/data measurement software package. The current version (13.0 for Windows) provided two different algorithms for multidimensional scaling (MDS) analysis: ALSCAL and PROXSCAL. Though the latter is the most recently developed approach to producing MDS maps, it was not available on the version licensed at the investigator’s institution. ALSCAL was therefore used for all MDS analysis; ALSCAL has been used extensively for many years and for most applications it works well. Though a detailed mathematical explanation of the MDS process is not appropriate here, a brief overview of the primary issues is presented to instruct the reader in the process of employing this method.
Multidimensional scaling is a mathematical means employed “to detect meaningful underlying dimensions that allow the researcher to explain observed similarities or dissimilarities (distances) between the investigated objects” (Statsoft, Multidimensional scaling, General purpose, n.d.). “Multidimensional scaling (MDS) encompasses a collection of methods which allow to gain insight [sic] in the underlying structure of relations between entities by providing a geometrical representation of these relations” (van Deun & Delbeke, 2000, Introduction). In other words, the procedure generates a spatial map with data points located relative to how they were perceived to be similar or dissimilar; similar items will be grouped together while dissimilar items are spread farther apart (Kruskal & Wish, 1978; Borg & Groenen, 1997). Thus this method is ideal for analyzing proximity measures of structural knowledge patterns, providing results that can be depicted in a visual representation showing relative strengths of connections among concepts in a specified content domain. Of the various methods employed for analyzing card sort data, MDS is one of the most commonly selected methods in the literature (Jonassen et al., 1993).
There are two primary components to examine when interpreting MDS maps: clusters and dimensions. When analyzing card sort data, cards (concepts) that were deemed more similar by a majority of the participants are shown clustered together on the plot map, while cards that were never placed in the same pile are located far away from each other on the map. Individual cards that were placed in different piles by the various cases will be “pulled” somewhere between those relevant clusters.

Dimensions tend to show a higher-level categorical division from the data. Cards/clusters along one dimension tend to show reasons why certain cards/piles are considered similar. For example, consider a sort task that included different types of mammals and reptiles. Subjects may divide the mammals into various piles (two-legged, four-legged, etc) and reptiles into different piles; the MDS map might show reptiles leaning toward one dimension (X-axis) while mammals are mostly located toward the Y-axis. This can be a useful clue in interpreting a hierarchical categorization of the results.
It is important to note that the actual location among specific dimensions on the map is meaningless. The resulting plot can be rotated among the dimensions and quadrants and not change the interpretation in any way. It is the relative position among the plotted cards/clusters that represents the underlying data patterns.

This representation attempts to reproduce the data patterns as faithfully as possible. However, a perfect match is impossible. Stress and squared correlation values are used to indicate the distortion of the resulting map, otherwise known as “goodness of fit”. Higher stress values indicate higher distortion and less accurate representation. Some stress will always be present; the issue is how much is considered unacceptable. Values less than 0.1 are quite ideal; values over 0.15 are considered by some to be not usable as the map output is too distorted to show accurate information (Multidimensional scaling, Stress, n.d.). Conversely, higher squared correlation results are better (as close to 1.0 as possible). Compounding this issue is the large number of variables that affect stress such as data error, level of similarity/dissimilarity in the data, number of dimensions plotted, number of data points, etc.

**Iteration history for the 2 dimensional solution (in squared distances)**

Young's S-stress formula 1 is used.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stress</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>.12616</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>.11287</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>.10412</td>
<td>.00875</td>
</tr>
<tr>
<td>3</td>
<td>.09926</td>
<td>.00486</td>
</tr>
<tr>
<td>4</td>
<td>.09632</td>
<td>.00294</td>
</tr>
<tr>
<td>5</td>
<td>.09429</td>
<td>.00202</td>
</tr>
<tr>
<td>6</td>
<td>.09276</td>
<td>.00154</td>
</tr>
<tr>
<td>7</td>
<td>.09152</td>
<td>.00124</td>
</tr>
<tr>
<td>8</td>
<td>.09048</td>
<td>.00104</td>
</tr>
<tr>
<td>9</td>
<td>.08957</td>
<td>.00091</td>
</tr>
</tbody>
</table>

Iterations stopped because
S-stress improvement is less than .001000

Stress and squared correlation (RSQ) in distances

RSQ values are the proportion of variance of the scaled data (disparities) in the partition (row, matrix, or entire data) which is accounted for by their corresponding distances. Stress values are Kruskal's stress formula 1.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Stress</th>
<th>RSQ</th>
<th>Matrix</th>
<th>Stress</th>
<th>RSQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.073</td>
<td>.974</td>
<td>2</td>
<td>.118</td>
<td>.938</td>
</tr>
<tr>
<td>3</td>
<td>.168</td>
<td>.903</td>
<td>4</td>
<td>.117</td>
<td>.933</td>
</tr>
<tr>
<td>5</td>
<td>.091</td>
<td>.959</td>
<td>6</td>
<td>.117</td>
<td>.942</td>
</tr>
<tr>
<td>7</td>
<td>.104</td>
<td>.949</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Averaged (rms) over matrices
Stress = .11591      RSQ = .94252

Figure 3.6. MDS stress and squared correlation output.

There are two primary methods within SPSS to improve stress values. The ALSCAL algorithm is designed to repeat the calculation process (iterations) until stress improvement is less than 0.001. Occasionally it is necessary to increase the number of dimensions in the output, which will lower final stress values. Two dimensions are the most typical and adequate for most situations. Three dimensions are sometimes used, but it is more difficult to read (consider trying to see distances in a 3-dimensional cube on a flat sheet of paper). Four or more dimensions are practically impossible to interpret and are not generally useful.

Stress is somewhat dependent on the number of cases plotted (subjects in a card sort). Larger numbers of cases will typically result in a wider variation of data responses, therefore resulting in a less tidy map plot. This also depends a great deal on the type of data used for the sort task. Cards that represent a very concrete subject content, meaning
a group of concepts that can only be sorted in a limited number of “correct” categories, should result in a more consistent map plot (but only if subjects “score” well). Consistent plots with highly clustered cards will have lower stress values. For example, if graduate music students were asked to sort composers into stylistic periods, the map should be fairly tightly clustered as most students should be able to put Bach into a Baroque pile and Mozart into a Classical period pile. Conversely, if 20 terms are presented for sorting that are based on personal preference and are less concrete, the data would probably show a wide variation in response categories. However, though in this situation clearly defined clusters may not be present, data points that lean along one axis may represent a general dimensional category underlying the data.

MDS also produces a weighted subject measurement that can indicate individuals within the plot that appeared to be different for whatever reason (hurrying through the research task, etc). If a particular case lies fairly distant from the others, it might indicate the need to eliminate it from the measurements.
In spite of these numerous technical issues with MDS output and interpretation, Borg and Groenen (1997) summarize by stating “All of these criteria are mechanical ones. They are not sufficient for evaluating Stress, nor are they always important. Kruskal (1964a) writes: ‘A second criterion lies in the interpretability of the coordinates. If the m-dimensional solution provides a satisfying interpretation…it may be well to use [it].’ (p. 45). Thus, while the researcher should keep an eye on stress results and overall goodness of fit, this should not override simply looking at the maps to determine if it makes sense based on the input data and subject matter.
Interpreting MDS maps

Is the MDS map valid?

When interpreting an MDS map one should look for the following to ensure the map is a valid representation of the data:

- Low stress values (closer to 0.0 is best)
- High squared correlation (RSQ) values (close to 1.0 is best)
- Are there clusters on the map? Do they make sense? Clusters indicate categories representing similarity among certain cards. Lack of clusters, or clusters that inaccurately represent the data, may indicate an incorrect MDS data input and/or solution.
- Are there data points that lie along one particular axis or dimension of the map? These can indicate underlying principles for how the data was sorted by participants.

Are all subjects (cases) valid?

- Look for “goodness of fit” among the individual subject (case) data.
- Derived and flattened subject weights show which individuals may lie outside the norm.
- Higher stress values for those individuals also indicate poor fit.

Reading the MDS map

- Clearly-defined sort piles with no overlap among subjects will be clustered in distant areas of the map.
- Cards which may end up in different piles will be “pulled” somewhere in between. Sometimes this means they actually fit in both piles, while other
times it may mean the participants’ mental models are either inadequate, inaccurate, etc. This can also be caused by an individual who does not focus adequately on the sort exercise.

Figure 3.8. MDS map showing outlier cards that were placed in different stacks, shown being “pulled” between both clusters. Note that the shaded clusters were manually added and labeled during analysis and not generated by the MDS process.

This demonstrates the need for background information on the subjects whenever possible in the attempt to explain anomalies. For example, in this study the experts all had different backgrounds and job experiences. This proved valuable in explaining variations in responses on the sort exercise. It was also valuable to visually examine the individual sort results provided by the JPEG exports from the sort software. This becomes overly cumbersome in a study involving a large number of individuals, but for
this study it was both simple and very instructive to examine the ten sorts from the experts and students.

**MDS procedures for this study**

The MDS calculation method employed for this study was ALSCAL. For processing a summed matrix of individual cases (such as for creating an all-student map), an option was selected that invokes the INDSCAL algorithm to provide a weighted, individual subject comparison. Using the individual dissimilarity matrices saved as SPSS files, a new file was created by opening the first of the individual cases. Then each of the other case files (either expert or student) were added using the *Data>*Merge Files>*Add Cases* command. SPSS sums these while vertically stacking them in the data view window. To run the MDS procedure, *Analyze>*Scale>*Multidimensional Scaling (ALSCAL)* was selected from the SPSS menu. All variables were selected in the left window and transferred to the “Variables” window. Specific configuration selections were as follows:

- **Distances**: *Data are distances / Square symmetric*
- **Model**:
  - Level of measurement: *Ordinal / Untie tied observations*
  - Conditionality: *Matrix*
  - Dimensions: 2 x 2
  - Scaling Model: *Individual differences Euclidean distance*
- **Options**:
  - Display: *Group plots, Individual subject plots, Model and options summary*
Default criteria values were used

Output information includes the following:

- Alscal procedure options (summary of the configuration options selected)
  - Stress / squared correlation information
  - Map coordinates for each card
  - Subject weight values
  - “Derived stimulus configuration” (main MDS map)
  - Derived and flattened subject weights plots
  - Scatterplots of linear fit (which helps show outliers as well)

Selected examples of these are included in the data analysis discussion in Chapter 4. Separate plots were run for experts and students in various groupings (all students, select students, etc). For processing a single case, i.e. a single expert, only one data file was opened in SPSS with no merging of multiple cases. The only difference in processing was that the “scaling model” was left as default and not set to “individual differences Euclidean distance”.

Procedural knowledge elicitation

Procedure design issues

Observing how students actually apply their knowledge to performing real-world tasks can show inaccuracies, gaps, and misperceptions of their structural knowledge that can affect how well they complete the tasks. Comparing this to the professional engineers provides insight as to where they may be going wrong and/or what they are missing. This triangulation allows for a more complete picture of the situation under investigation,
providing in-depth, complementary data that should help answer the research questions and inform future instructional practice. Therefore, participants completed a series of operational tasks using the Pro Tools system, talking aloud as they worked through each exercise. This think-aloud procedure fostered operationalization of structural knowledge as it relates to actual tasks. Following think-aloud protocol guidelines, the investigator did not ask individuals to explain or otherwise think about why they do certain things, but merely instructed them to say aloud what they were thinking as they considered possible options and actions. This provides the most direct view into their thought processes during “normal” activity, whereas asking for explanations alters their level of processing and information retrieval and interferes with the mental processing of the task at hand (Ericsson & Simon, 1993).

Appropriate exercises were developed that match specific tasks in professional recording studio operations. Each one was required to meet the following criteria: 1) involve some type of signal routing within the system; 2) represent an actual, real-world task that an engineer might perform during a recording session; and 3) represent tasks that would be performed on both analog and digital recording systems. Additionally, a range of difficulty was included, the first simple exercise being followed by others requiring varying degrees of complexity (but not in order of difficulty). The ones selected for this study include:

Practice. Apply compression to the snare. Do not worry with actual compression settings; just make sure it is working.

1. Apply a stereo compressor to the entire mix. Again, do not worry with settings; just make sure it works.
2. Apply one common reverberation effect to the following tracks: Kick, snare, overhead left, overhead right. Do not apply individually on each channel.

3. Set up a single cue mix for all tracks and route to an output bus connected to the studio’s headphone system. Provide a master cue mix level control.

4. Apply a single stereo compressor to tracks 1-3.

5. Apply a single stereo compressor to tracks 1-3, but leave the original, unaffected tracks blended at the mix bus so we hear them as well.

Since audio signal flow, which is the connection and routing of signals in a recording system, is the principle foundation for all recording engineering, these procedures focused on virtual signal routing within Pro Tools. If an engineer understands fundamentals of audio signal flow they should be able to apply that to the software environment; even if they do not know specifically how to complete certain functions within the software program, they should at least be able to articulate what they are trying to do. This is enough to help indicate how their mental model might be structured.

Each of these tasks except the practice exercise required the user to create additional tracks in order to facilitate grouping and processing of existing tracks as well as assigning various routing/processing options to certain tracks. For example, to apply a stereo compressor to the entire mix, the user must create a master fader track which sums all other musical parts together in a stereo blend. The compressor processor is then inserted into this new master track to affect the entire mix. This is identical to what
happens on an analog recording console, but is much more difficult to “see” and comprehend in a software-based system.

As another example of signal flow tasks in Pro Tools, adding compressors and reverberation is usually accomplished by assigning plug-ins to audio tracks, which are extra software programs that operate from within Pro Tools to process audio signals. Doing this on an individual track is straightforward; performing this operation on a group of tracks requires a higher-level conceptual understanding of signal routing. While this routing is completely virtual in Pro Tools (no hardware or cables required—see Figures 3.9 and 3.10 for a comparison), it still generally follows the traditional, analog concept of how to “connect” these different types of devices. While the professional engineers were expected to demonstrate this as they perform the tasks, it was thought that students might possess a less clear, even arbitrary method for accomplishing the procedures. As it was very likely they had accumulated a rote-based method for completing certain specific operations, the think-aloud might have possibly revealed little in the way of appropriate conceptual patterns that could be applied to solving these tasks.

![Figure 3.9. Analog patchbay routing signals between console and external processor. By connecting cables between both devices, engineers can easily see where the signal goes.](image-url)
Figure 3.10. The same signal routing in Pro Tools has no cable connections, requiring the user to figure out how to complete the virtual routing. It is actually more difficult than it appears in this example.

Think-aloud procedure implementation

A simple Pro Tools recording system was adequate for conducting this study, consisting of the Pro Tools software (version 6.7), a simple hardware interface required to operate the software (Digidesign’s M-Box), and a host computer (Apple Macintosh 1Ghz PowerBook running OS 10.3.7). No other control surface hardware was utilized.
Control surfaces are devices that feature actual knobs, switches, and faders that facilitate hands-on manipulation of software operations, thereby reducing the need to use the computer mouse for navigating through menus and such. For the purpose of this study such devices would merely complicate the process as users must know how to operate that particular device in order to accomplish tasks within the Pro Tools software.

One master Pro Tools template file was developed featuring several existing tracks of musical parts. This template was opened for each individual task; once completed, each was saved as a separate Pro Tools file for later screenshot capture and analysis.

At the start of each session, the researcher provided an overview of what the procedure involved, specifying that the objective was not to see how quickly the tasks are accomplished, but rather to simply observe how they go about approaching each task component. The concept of how to “think aloud” was explained, and the participant completed a practice exercise in Pro Tools in order to get accustomed to the decidedly unnatural experience of thinking aloud. A sample QuickTime movie file was originally developed to demonstrate the procedure, but it quickly became evident it was generally unnecessary, time-consuming to watch, and not as effective as having the participant actually try the procedure.

Interaction between researcher and participant was minimal, consisting primarily of an occasional reminder to “keep talking” as needed. For each of the five tasks, the instructions specified what was to be performed, just as if a chief engineer had given orders to their assistant for setting up the system before a recording session. No other task-related instructions or hints were provided other than general information regarding
the exercise overall. Each task was presented on a separate sheet of paper with the following instructions at the top of each: “Complete this task using the Pro Tools file currently open. Remember to keep talking aloud and describing what you’re doing and thinking. Take your time.” See Appendix C for the exercise instruction document (reduced to a single page).

During each task setup, a computer program running simultaneously with Pro Tools recorded all screen activity along with the verbalizations captured by the computer’s internal microphone. Snapz Pro is a simple, yet powerful capture application that provides complete control over static screenshots and audio/video recording of computer activity. These files were saved as individual QuickTime movies using a 10 frame-per-second video setting (to reduce file size) and an audio sample rate of 22kHz. This software-based method streamlines transcription by eliminating the need for transfer or digitization of video footage and also provides the capability for capturing screenshots or video clips directly from the movie file. One of the negative aspects of this approach is that physical activity (non-verbal indicators) is not captured. This requires the investigator to take notes throughout each exercise, which was done for this study.

To aid the investigator in ensuring that each exercise was administered completely and correctly (saving each exact file, not saving to the template file, using correct filenames for each movie, etc.), a checklist document was developed so that each specific action could be followed and checked when completed. (See Appendix D.)

Data from the think-aloud procedure include:

- Pro Tools files for each task, for each participant. These were used for capturing screenshots of the final setup configuration.
QuickTime movies for each task. These include all screen activity (mouse movements, menu commands, dialog boxes, etc.) as well as the audio verbalizations of the participants.

- Investigator’s notes from each session.

Analyzing think-aloud data

Transcription of the QuickTime movies consisted of capturing all verbalizations with the added benefit of viewing subjects’ actual software procedures in the attempt to confirm intentions along the way. This proved invaluable in evaluating the direction of a subject’s thought process along with confirming which specific action or function they were referring to in the software while they worked. Additionally, the investigator’s notes taken during the protocol procedures informed this data to provide a more complete picture of what each subject was thinking and doing.

After movie transcription and screenshot printouts were compiled, analysis of the think-aloud exercises followed Miles and Huberman’s (1994) three-stage process: data reduction, data display, and conclusion drawing and verification. Repeated reading of the transcriptions gradually revealed patterns and themes that were used for creating a coding system. To help focus the data mining process a theoretical framework of key issues was established based on the original research questions for this study. With this in mind, an appropriate coding scheme was gradually developed through an iterative process, made possible by constant reflection upon the research questions and overall intent for the study.

Throughout the process of coding, lists and descriptions of emerging, relevant concepts were sorted and experimented with to produce a series of data displays that
present significant concepts in a format easy to compare. Initial data displays were
descriptive in nature, following the format of an event listing as described by Miles and
Huberman (1994). Each of the fifty task protocols, after coding was complete, was
sequenced to indicate the flow of events. These were not broken into discrete steps in the
task procedures, however, as that was not the intent of the study. Rather, the flow of each
individual’s thinking process along with their activity in the software as they worked
through each exercise is presented in linear fashion, allowing a powerful comparison to
be made of various issues and thought processes involved between subjects.

The event lists provide a data reduction result that then allows for further
examination and processing for comparisons in a more explanatory manner, attempting to
make sense of why significant patterns and themes emerged. This resulted in a variety of
data displays that examine results from various angles, making comparisons within the
groups expert and novice as well as between them. This triangulation procedure was
intended to directly address the research questions for this study, the objective being to
identify possible deficiencies or differing patterns that may be hindering novices in their
understanding of recording systems.

**Final analysis**

Once both the card sort and think-aloud procedural activities were analyzed,
comparisons were made in order to provide a more complete mental model analysis than
would be possible (or accurate) with only one activity. This was accomplished primarily
by comparing the structural organization of content knowledge with how each individual
approached the procedural tasks. Presumably the more formally and accurately structured
an individual’s mental model is, the more their procedural operations should exhibit a
corresponding structured approach. Even if a subject is unfamiliar with this particular software system, procedural mistakes made should at least follow a structured concept of what should be happening. Individuals who do not seem to have a grasp of an appropriate organization of the domain content, when confronted by an unfamiliar task to be completed, should demonstrate procedural and conceptual mistakes for which their underdeveloped model cannot resolve. The primary purpose of this study was to expose whether the graduating students have sufficient mental models of recording systems that can give them a likely chance of success in transferring their knowledge to new situations, the benchmark being the models demonstrated by the experts.
CHAPTER 4
RESULTS AND DISCUSSION

Card sort

Several MDS plots were run from the expert and student matrices in order to determine how best to facilitate the expert/novice comparison objectives for this study. Once the investigator had developed a solid grasp of how MDS was plotting various data structures (individual, small group, large group, various option settings, etc), specific data sets were selected for use in the analysis. All participants had completed two sorts using the same list of cards; these were compared to discern any significant changes that might alter the individual’s model approach. While there were a few minor differences for various individuals, there was little perceived as having significance for the purposes of analysis; exceptions are discussed throughout the chapter. Therefore nearly all data processing and analysis were performed with the first sort data. To aid the reader in understanding the card names and how they fit types of categories, a brief list defining each is provided in Appendix E.

Experts

The first sort results for each of the three experts were input into SPSS as individual dissimilarity matrices and a group INDSCAL MDS plot was run. Results were as follows:
Table 4.1. Expert sort results.

<table>
<thead>
<tr>
<th></th>
<th>Group</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final S-stress</td>
<td>0.05272</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(16 iterations)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Averaged S-stress</td>
<td>0.072120</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RSQ</td>
<td>0.97825</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Individual S-stress</td>
<td></td>
<td>0.057</td>
<td>0.083</td>
<td>0.071</td>
</tr>
<tr>
<td>Individual RSQ</td>
<td></td>
<td>0.992</td>
<td>0.966</td>
<td>0.977</td>
</tr>
</tbody>
</table>

First, the S-stress and RSQ values for this 2-dimensional map are satisfactory and indicate the map is probably a valid representation of the data. Second, there are loosely organized clusters of cards, though this may only be evident for someone who understands the content area. There are three primary clusters identified from this map plot.
• Cluster 1: Monitoring
• Cluster 2: Insert points for signal processing
• Cluster 3: Routing / busing

Additionally, there is a less-defined 4th cluster that changes slightly with each individual’s card pile configurations. This group primarily includes channel path/track related functions. All clusters are somewhat loose due to different sort criteria by the three experts. However, there are several cards “hanging” by themselves for no apparent reason. Most of these were the Pro Tools cards, which seemed to indicate some sort of pattern that did not fit expected operational or conceptual paradigms.

Since the map output from SPSS was deemed adequate to represent the source data, the issue was to identify potential factors that were fragmenting the card clusters. Determining which individuals were causing skewing in the plot was accomplished first by reviewing the individual S-stress and RSQ values. By looking at table 4.1, it can be seen that Expert 1 has a stress value significantly different from the other two experts. This clue can be followed up by examining each individual’s original sort workspace, i.e. the JPEG screenshots from the card sort exercises. These show the actual piles of cards and the pile names assigned by the participant, making it easy to compare.

The most different of the experts was Expert 1, who followed a completely different paradigm. Instead of sorting based on a hierarchical operational or conceptual framework, he simply divided all computer-based routing and all console-based routing into two piles, with a third comprising hardware outboard gear (compressor, effects, PB insert). This was surprising, but it is a viable method to sort the concepts—participants
were asked to sort them in any way that made sense, and a computer vs hardware division is logical. Post-study questioning of this individual confirmed the investigator’s assumption about his rationale behind this organization—as a mastering engineer he determines whether to use software or hardware tools for any given project. For the purpose of developing an operational/conceptual mental model for instructional purposes, however, this result is not useful.

As mentioned, the second sort performed by most individuals did not reveal any significant differences. For Expert 1, the second sort represents a completely different approach that is worth examining. There are still three piles, but there is no computer vs hardware division. One pile represents the highest level operational concept offered by the twenty cards—the two primary signal paths in a recording system (*channel path*, *monitor path*). Another pile includes all cards related to “mixes”, meaning all functions utilized for setting up monitor mixes, cue mixes, and signal grouping during recording and mixing. The third pile includes all functions and equipment related to signal processing with hardware outboard gear (including virtual processing in Pro Tools which functions like outboard processors). One small accuracy issue arises by placing *pre-fader* in the “outboard” pile; this card typically belongs with “mixes” (although there are exceptions). Otherwise the sort is accurate. This second sort is viable in its logic, and provides a much better source for instructional considerations than Expert 1’s first sort. Its utility as a foundational system model is limited, however, due to a lack of depth of hierarchical divisions. From an educator standpoint, it may not be helpful in helping novices “see” the fundamental system model and its intricate signal routing. However, it provides another perspective that is mostly accurate and potentially helpful for students
as they progress through their instruction. Once they have some background and experience with such systems, such a model may indeed clarify operational categorization of functions they would then be somewhat familiar with.

It is Expert 1’s first sort information that explains a majority of the outlying cards in the expert MDS plot. Since these are the Pro Tools (computer-based) cards, these were pulled away from the common clusters identified by the other two experts. In other words, Expert 1’s sort approach was from such a completely different perspective it renders the map barely useful in attempting to identify a reference expert structural map to be used in instructional design. Even using Expert 1’s second sort to run another expert plot did not provide a coherent output for the purposes of this study, primarily due to a lack of hierarchical layering and refinement in the piles. Therefore a second MDS plot was run using only Expert 2 and Expert 3 data matrices. Examination of their card sort piles shows much in common in terms of pile configurations; though not without contradictions and variations, these two are far more similar. Results of this combined MDS plot are as follows:

<table>
<thead>
<tr>
<th></th>
<th>Group</th>
<th>E2</th>
<th>E3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final S-stress</td>
<td>0.03902</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(9 iterations)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Averaged S-stress</td>
<td>0.06269</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RSQ</td>
<td>0.98184</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Individual S-stress</td>
<td>0.071</td>
<td>0.054</td>
<td></td>
</tr>
<tr>
<td>Individual RSQ</td>
<td>0.977</td>
<td>0.987</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2. Expert 2 and 3 sort results.
S-stress and RSQ values are entirely satisfactory for this plot. The map clearly shows tightly clustered cards with fewer outliers compared with the first expert map. By discarding Expert 1 from the plot, Experts 2 and 3 provide a more application-oriented display. The previously separated Pro Tools cards are now mostly clustered with their operational categories. This map produces five identifiable clusters:

- Cluster 1: Main channel routing/busing
- Cluster 2: Auxiliary send/return routing
- Cluster 3: Monitoring
- Cluster 4: Inserts for signal processing
- Cluster 5: Channel properties/routing
Most of the clusters are well-organized and consistent between the two experts. One particular card was pulled in between two clusters for appropriate reasons. *Cue mix* was placed in both cluster 2 (aux send/return routing) and cluster 3 (monitoring); this is operationally correct as aux sends are used to generate the cue mix, while the purpose of having a cue mix is so the musicians can monitor what they are playing in the studio.

Another card, *Assignmatrix*, was pulled in between two clusters that are very similar, yet were defined somewhat differently between the two experts. However, there are a few anomalies that need to be addressed in determining the viability of this map as a reference model. A few examples are explained here.

Cluster 1 primarily includes cards *HW assign, Bus,* and *Subgroup.* These were all put into the same piles by both experts, but Expert 2 also included these with cluster 2 (aux send/return routing). This pulls the entire cluster 1 closer toward cluster 2 on the map, indicating a link between these two clusters. In fact, combining these indicates a lack of hierarchical understanding on the various sends and routing functions in recording systems. There are indeed distinctions among these, and piling them together loses this depth of operationalization.

For cluster 2 (aux send/return routing), Expert 2 did not include *pre-fader* and *post-fader* cards, although she included the Pro Tools equivalent screen captures. Perhaps this is explained by the fact that the type of work this individual performs daily probably rarely requires the use of a pre-fader function, and post-fader is set by default in the software without the user knowing or thinking about it. Apparently not being aware of what else to do with these, she placed them in a generic and fairly meaningless “channel properties” pile.
Expert 2 placed *PB insert* with cluster 2 (aux send/return routing). This is most interesting in that these particular connection jacks on a console patchbay are not used in relation to aux sends and returns. One assumption is that she saw the photo of patchbay jacks and mentally associated that with connecting external effects processors. Again, this does not reflect a refined, categorical understanding of patchbays, routing, and processors and may be a result of a predominantly Pro Tools-based work experience.

Both experts misinterpreted *PT aux return*, thinking the graphic merely represented a channel strip from Pro Tools. It is actually a specific track in Pro Tools created for adding reverb effects to the mix, so it operates as an aux return.

Seeing several issues with Expert 2’s sort approach questions the utility of this particular map plot as an expert reference. Although much closer than the first map which included all three experts, the differences in background and experience between Experts 2 and 3 make it difficult to produce a clear-cut model (not an unexpected outcome). Expert 2 uses Pro Tools software-based systems almost exclusively and rarely spends much time with traditional analog hardware systems and methods. Her only major experience with analog systems was in college many years before. Upon graduation she was immediately employed in a Pro Tools-based post-production company. This type of work is somewhat different than traditional studio recording and thus the mental model was expected to be different in various ways. In this case not only is the model different, but awareness of all the concepts included in the exercise is incomplete. Additionally, Expert 2 spent relatively short time on the sort exercises. In contrast, Expert 3 labored over his decisions, attempting to get everything exactly “right” in his mind. For these reasons it was decided to eliminate Expert 2 from the MDS map and examine Expert 3’s
sort data and plot output individually. This demonstrates the simple point that real-world data is messy and ridden with multiple factors that affect the outcome. However, even though Experts 1 and 2 were eliminated from the reference model, it is important to examine each of their particular approaches and use that information to help guide curricular design. The results of an individual ALSCAL plot for Expert 3 follows:

<table>
<thead>
<tr>
<th></th>
<th>E3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final S-stress</td>
<td>0.00000</td>
</tr>
<tr>
<td>(1 iteration)</td>
<td></td>
</tr>
<tr>
<td>Averaged S-stress</td>
<td>0.00000</td>
</tr>
<tr>
<td>RSQ</td>
<td>1.00000</td>
</tr>
</tbody>
</table>

Table 4.3. Expert 3 results.

It should be noted that perfect stress and squared correlation results are present only because there are no different data matrices for the software to compare and reconcile.
This map features perfectly clustered categories (again because there was only one data source for processing). What is shown in this plot are simply the five piles from this individual’s sort. The location of each cluster on the map is determined by the number of cards placed in each pile; there were seven cards placed in cluster 1 whereas all other piles had from two to four cards each. There is nothing significant about this for the purposes of this study as there was no pre-determined “balance” of how the cards would or should be organized. Clusters for this map are as follows:

- Cluster 1: Aux send/return routing
- Cluster 2: Inserts for signal processing
- Cluster 3: Track output routing/functions (Channel and monitor path)
- Cluster 4: Track output routing/functions (Channel path)
- Cluster 5: Monitoring
As noted earlier, Expert 3 demonstrated a hierarchical categorization of the cards based on their operational functions. Rather than clustering all cards related to “channel routing”, this individual sees the difference between different signal paths on the console (both analog and software) and further distinguishes the operations of the short and large faders. Briefly, there are typically two primary signal paths in use during a recording session: channel (incoming microphone signals to be recorded) and monitor (signals from the multitrack recorder that are already recorded or are being recorded that are used to hear everything). Often the large faders on a console are used to control the monitor path signals and the short faders control incoming channel path signals (though this can be reversed as desired). One approach to sorting the cards is to group channel-path related functions together separate from monitor-path related functions. In this case, Expert 3 separates functions in terms of which fader would control them, a very operational approach. Thus, some routing-related cards are grouped under a “large fader functions and routing” pile while others are placed in a “grouping and routing functions (short and large fader)” pile. A third routing pile is termed “short fader functions and routing”, i.e. the aux send and return pile. This is a decidedly analog perspective that has been adapted to the Pro Tools environment as needed. An example of this is the labeling of the aux send pile. On analog consoles rotary potentiometers (knobs) are used for controlling aux send levels. In Pro Tools, however, short faders are used for aux sends, thus these related cards were grouped in the pile and given the “short fader” umbrella. Another possible example of adapting analog thinking to Pro Tools systems is the inclusion of effects in the insert pile. While this is accurate for analog systems, it is much more predominant with Pro Tools to think of adding effects through the channel inserts. In analog recording the
inserts are most often used for compressors and other similar tools rather than for effects, which are more often accessed via aux sends and returns.

There are a few curious decisions in this sort. Since large faders are typically used for monitoring (both in analog and Pro Tools systems), it is surprising Expert 3 did not group monitor path and monitor mix with the large fader pile. His placement of channel path in the large fader pile is also interesting as channel path typically coincides operationally with busing and routing signals, which would be the “grouping and routing functions (short and large fader)” pile. To be sure, there are a variety of interchangeable cards and piles with this particular sort exercise, so it is difficult to create a perfect sort from certain approaches unless one could duplicate cards.

The issue at hand at this point was whether the higher-level awareness and conceptual thinking behind Expert 3’s sort choices was viable for instructional purposes. The selection of an appropriate model for students to assimilate and develop is a critical one, especially in a complex content area such as this. The entire point of this study was to determine some of the issues in how student engineers are trained with the objective being to facilitate transfer of their still-developing model to various situations they may find themselves in after graduation. Expert 3 is a professional engineer and not an educator. His sort choices make logical sense with no contradictions from an operational point of view. He is able to easily transition between analog and Pro Tools systems due to his extensive background and experience with both. Would this model be appropriate for educational purposes? One serious issue is the fact that not all recording systems feature large and small faders; thus training students to think along those terms may not be helpful from a global transfer perspective. Obviously Expert 3 does not think exclusively
in terms of short and large faders; he understands conceptually and operationally what they represent and are used for, and thus can easily adapt to different situations, systems, etc. In other words, he possesses a multidimensional, relational model that is flexible and accurate no matter what he is working with. This, however, is likely too complex for novices to comprehend. Though Expert 3’s model has been refined over extended periods of time, it is based nearly exclusively on a wide range of experiences. Since students do not share this range of experience and level of expertise, it is crucial to identify the significant, essential aspects of an expert’s model that are accessible for the novice and that can be refined over time.

In order to determine a better instructional model, a breakdown of the primary signal flow and operational functions needs to be defined. Operationally, the primary categories of signal routing in a recording system are as follows:

- Channel path: incoming microphone signals being recorded to multitrack
  - Typically includes insert points for adding channel-specific processing (compressors, etc)

- Monitor path: after being routed to tracks, all signals return through the monitor path for a variety of functions:
  - Monitor mix in the control room (listening to everything)
  - Cue mix for the musicians in the studio (using aux sends)
  - Aux sends and returns for adding effects

To complicate matters, this changes somewhat depending on 1) whether the session is a multitrack recording or mixdown session and 2) the specifics of the recording
system. In Pro Tools, the concept of channel/monitor path is practically non-existent. There is no dual-path configuration, merely individual tracks. For mixdown sessions all insert points and aux sends are controlled from the same signal paths (returns from the multitrack recorder). Therefore there are two main concepts of signal routing that can be distinguished for instructional purposes:

- Channel vs monitor path signal flow
- Channel-specific vs multi-channel routing/processing

When recording, student engineers must mentally separate and track incoming microphone signals and multitrack returns that are running through different locations in the console/recording system. All monitoring is performed with the signals returning from multitrack, but any processing and routing to get signals to the multitrack have to be performed in the channel path. To further confuse the novice, incoming signals on a particular channel are often routed to a differently numbered track for recording, thus there are controls on two different channel strips on the console that affect that same part. Channel-specific processing involves affecting the signal quality on one particular signal/track whereas multi-channel processing is used to affect a group of signals. These are accomplished via individual channel insert points, bus insert points, aux sends, etc. Routing and processing options are rather complex and add to the difficulty of helping novices develop accurate mental models.

Considering these issues, it seemed that although Expert 3’s model was accurate, logical, and insightful, it might not represent the best primary source for an instructional model. Now thinking that his model could serve as a secondary reference that would be extremely useful for informing curricular and instructional design, the investigator
decided to complete a card sort himself and run an MDS plot. Since the investigator is the instructor for these classes and is also an experienced recording engineer, it was considered useful to compare an engineer/educator’s model with the experts’ perspectives. The advantage for the investigator comes from many years of teaching recording engineering, which provided much insight into how students assimilate new information and work through the complexities of recording systems. Two sorts were completed following slightly different criteria (explained below). The sort data dissimilarity matrices were processed with the ALSCAL algorithm resulting in the following plots:

![Investigator MDS Map #1](image)

Figure 4.4. Investigator sort #1 MDS map.
Again, the stress and squared correlation values will always be perfect with only one case so they are not included here. The only reason to run an MDS plot is to generate a visual map that provides a means for direct comparison with the student-generated map. A point should be made regarding the location of clusters on these maps. Map #1 (Figure 4.4) shows a cluster on the extreme left with all others in the upper and lower right quadrants. With a single case MDS map there is no other classification of data points other than which cards were in which pile. The software cannot make inferences based on the concepts on those cards. Therefore no significance can be found in the spacing of these clusters other than simple mathematics—the left cluster contains the most cards by a considerable margin (6 cards). All other clusters have three cards or less and so were grouped in closer proximity to each other by the ALSCAL algorithm.
The two sorts were approached from a slightly different perspective. Sort #1 is designed to represent an operational conception of the cards arranged hierarchically. As can be seen from the actual sort workspace (see Figure 4.6) even the piles are arranged in levels and groups representing system signal routing fundamentals. For example, the two
primary signal paths as described earlier are channel and monitor path. In this sort, operations associated with either of these are placed below these headers. So, a typical setup for a cue mix for the studio musicians would be facilitated with monitor path controls, whereas compressing a signal during recording would normally occur in the channel path (mic signal being routed to the multitrack recorder). As an instructional model this is much clearer and more accurate than a single-level sort. It should be noted that the participants were not asked to arrange their piles physically to represent any particular order or association, but merely to place similar items in piles. Standard procedures for performing card sorts from the literature do not indicate employing this physical arrangement of the piles, and scoring methods are unable to reflect this. Therefore, in order to more closely resemble the sort criteria followed by the participants, the second sort does not reflect anything in the physical arrangement of the piles, but a hierarchical perspective of operational concepts is still present—note the first pile that includes both primary signal paths in a recording system (see Figure 4.7).
Cards related to specific functions are grouped together, such as associating pre-fader, PT pre-fade send, and cue mix under the functional heading of setting up a cue mix, while all other aux-related cards are associated with routing signals for multichannel effects. Of course, there is overlap and several possibilities for swapping cards in certain piles; monitor mix does not really have an appropriate pile to fit into with this sort approach, yet in most participant sorts this card was associated (correctly) with monitor path. However, both of the investigator sorts represent probably the most accurate, meaningful, and accessible organization of the terms included in this exercise from a mixture of both operational and pedagogical perspectives. Students require a relatively simple description (model) that locates all the various controls and functions in specific categories that make sense to them at the time. The model must be foundational in nature, facilitating future refinement and development as each individual progresses. Considering
all of these issues, the decision was made to employ either or both of the investigator models as the primary reference source both for analyzing student data from this study as well as for future instructional design.

This does not render the original expert sorts unusable. Even both of Expert 1’s sorts (the first individual eliminated in determining an ultimate reference model) indicate something about how a professional from a certain background sees this subject matter, and thus demonstrates the powerful method of structural knowledge elicitation that card sorts provide. Merely observing these individuals performing tasks on various systems would probably not reveal some of the thinking apparent in the various sort results, and it is highly informative when designing curriculum and specific instructional situations for recording engineering.

Students

Seven college seniors each completed two sets of card sorts. As previously mentioned, the second sorts did not differ substantially enough to detect different model patterns, so all analysis focused on the first sort from each participant. This data was input into SPSS as individual dissimilarity matrices and a group INDSCAL MDS plot was run. Results are as follows:
As expected, the MDS map is much less organized; individual cards are spread throughout the map with most located in the right two quadrants. However, although clusters are more difficult to identify, they are indeed present and generally follow the experts’ categories.

- Cluster 1: Inserts for signal processing
• Cluster 2: Channel path routing
• Cluster 3: Monitoring
• Cluster 4: Aux send/return routing

Having put labels on these clusters it is important to note that the data is not nearly so clean and defined for two reasons. First, due to the distortions present in any MDS output, precise locations of data points on the map are not very meaningful. Second, even considering this issue, the MDS map proves to be an excellent means to reveal and represent the still-developing model of students’ structural knowledge. While many related cards are located generally within a cluster area, several of them are pulled in various directions based on the fact that individual students treated them each somewhat differently. Some of these are conceptually and operationally correct while others make little sense at all. For example, cue mix is located partway between the aux send cluster and the monitoring-related cards (monitor mix, monitor path). This makes operational sense and is an accurate placement. However, PT post-fade send is pulled away from the aux send pile toward the inserts cluster. This does not make sense and is primarily a result of inaccurate individual sorts.

S-stress and RSQ values indicate two students (S1 and S3) who seem somewhat more different than the others. The weighted subject plot also indicates this, but upon examination of the original sort data it seems that there are actually multiple “outliers” who differ for various reasons. Student 1 created three piles that were insufficient to indicate a hierarchical organization of the concepts. He blurs several operational concepts in his piles, resulting in an unclear model. This reflects the observed lack of focus
brought to the sort task (typical for this individual). Students 3, 6, and 7 suffer the most from a lower level understanding of the concepts presented in the exercise. Their sorts reflect a vague, unclear, and sometimes inaccurate picture of how these terms relate. This is a direct reflection of these three individuals as neither spent a great deal of time engineering in the studios outside of class projects.

In contrast, Students 2, 4, and 5 seemed to have the best grasp of the structural model of how these concepts relate. Student 2, though he sorted from a conceptual and not an operational perspective, presented a very thoughtful organization that was hierarchical and logical. As an example of applying a conceptual vs operational approach, Student 2 separated all aux send cards from a pile labeled “effects”, even though aux sends and returns are the means for adding effects to a recording. He did discern the difference between aux-related signal flow and other sub-mix functions, which indicates a fairly refined awareness of their operational uses. Student 5 presented the most analog-based sort of the entire group. For example, “inserts” are grouped according to analog console operations rather than how they are implemented in Pro Tools. He also thinks in terms of analog signal paths (channel and monitor). This was primarily a conceptual sort; for instance aux sends and returns were divided into two different piles although they are directly related operationally. Even so, Student 5’s fairly accurate overall perspective is partially a result of his extensive experience in the studios along with his internship at a major recording studio. Student 4 had probably the most operational sort of all students. Piles were developed and even labeled from an application-oriented approach: “tracking/channel path routing”, “aux/internal bus routing and application”, and “insert routing and application”. He was one of only two students to combine cue mix and aux
send-related cards in an application-oriented pile; most others mentally separated aux-related concepts from monitor functions (which includes cue mixes). He also presented a highly analog approach although he had spent considerable time using Pro Tools to produce his own albums; many cards “interchangeable” between analog and Pro Tools perspectives ended up in an analog organization. *PT insert* is included with *PB insert* and *compressor*, not *effects* as would be the case from a typical Pro Tools user. The explanation for this may be that this individual was a highly capable academic student, so the class-related, analog-oriented training had obviously “stuck” better than with many students.

Considering the fact that Students 1, 3, 6, and 7 were so different, a separate plot was run with the three “strongest” students, Students 2, 4, and 5. Their sorts seemed to match most closely with the experts. A map was generated from the Student 1, 3, 6, and 7 group as well, but as predicted the results were scattered, ill-defined, and unusable for analysis purposes. Each of those individual’s sort JPEGs were examined separately and notes made concerning issues that arose. These were primarily inaccuracies and vague understandings of conceptual and operational relationships, again most likely due to a combination of these students’ limited experience and incomplete understanding from classes. The results for the Student 2, 4, and 5 group are discussed next.
<table>
<thead>
<tr>
<th></th>
<th>Group</th>
<th>S2</th>
<th>S4</th>
<th>S5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final S-stress (10 iterations)</td>
<td>0.05276</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Averaged S-stress</td>
<td>0.08355</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RSQ</td>
<td>0.96902</td>
<td>0.086</td>
<td>0.090</td>
<td>0.073</td>
</tr>
<tr>
<td>Individual RSQ</td>
<td>0.967</td>
<td>0.962</td>
<td>0.977</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.5. Students 2, 4, and 5 sort results.

Figure 4.9. Students 2, 4, and 5 MDS map.

S-stress and RSQ values are satisfactory for this plot. The map is much more tightly clustered in meaningful categories. Four primary clusters can be identified:

- Cluster 1: Inserts for signal processing
• Cluster 2: Aux sends and returns
• Cluster 3: Signal routing
• Cluster 4: Monitoring

Although these clusters do not indicate a very hierarchical sorting as the expert reference sorts provided, at least conceptually it seems to show that these students seem to understand the majority of how these concepts relate. Cluster 1 is the most clearly distinctive category as all students placed these cards together. Interestingly, only one of these three individuals placed effects into this category. Keeping effects separate from the “inserts” cluster reflects more of an analog perspective—in Pro Tools inserts are used extensively for effects, whereas on analog equipment effects are added predominantly via aux sends and returns, not inserts. Again, their instructional background was heavily grounded in analog signal flow and much of their hands-on experience had been in analog studios (though not exclusively).

Signal routing cards were sorted in different ways which is why they are slightly scattered on the map. In spite of this inconsistency, there is nothing to indicate cause for concern over whether or not these students understand the fundamentals. For instance, Student 5 put bus under “tracking/channel path routing” and subgroup under “main mix routing”, two cards that were often placed in the same pile by other students. This is hierarchical and makes operational sense in that a subgroup is often used to combine multiple tracks during mixdown. On the other hand, a bus output is used to create the subgroup, so these are closely related concepts; either sort approach is satisfactory. Much
of the differences are minor and do not significantly impact the viability of their models at this stage of their development.

The sole straggler, *effects*, was explained above as being placed in two piles that operationally and conceptually make sense. Monitoring-related cards (cluster 4) were placed in various piles depending on the specific organization/definition of piles, so there is no definitive monitor cluster. They are close enough on the map, however, to indicate that most individuals considered them related in some way.

Overall this map corresponds well with the expert maps. Even though each student had somewhat different thoughts behind their sorting and pile-labeling decisions, they mostly saw the relationships between the concepts. Their sorts were not as hierarchical nor as operational as the expert results, but clearly more so than the other four students.

**Think-Aloud Protocol**

Since analysis of protocol data can take the researcher down multiple avenues of investigation, it was important to reaffirm the desired results from this study. This in turn drove development of appropriate code categories and types of patterns that were derived from the data. Based on the research questions, this study was designed not to prove any particular theory, but rather to determine how well students in a particular instructional situation had developed in terms of their mental model for recording systems. The procedural think-aloud protocol sought to reveal how well they could apply their individual models to specific situations, some familiar, others not so familiar. Fundamental patterns of how recording systems work had been emphasized throughout their instruction in the degree program. Were these evident during the procedural tasks?
How did they compare with the experts? Which aspects indicate weakness in the results? Were their models insufficient for the tasks, or were there other issues that emerged?

Major themes for coding and analysis included quality of the model evidenced, mental processing required to solve the task within the specific software program, whether procedures followed in the software were correct or, if incorrect, whether it was a model or software issue, and results for each task, again determining whether an incorrect result was due to a faulty model or understanding the software. If several students exhibited incorrect procedures, insufficient models, and incorrect results, then the instructional design of the degree program should be examined. If most students completed the tasks correctly with coherent models, then presumably the instructional program is relatively sound. Even if some students produced incorrect procedures and results, if these seem to be based on fairly solid models but are caused by difficulties interpreting the software interface, then the underlying instructional program would be deemed adequate. In this case attention should be focused on helping improve transfer of their sufficient but still developing mental models to more complex situations.

As described in Chapter 3, transcriptions of the Pro Tools task procedures were analyzed and coded for the purpose of producing data displays that first described and then compared activities between subjects. Review of the card sort exercises also informed the protocol analysis, providing a triangulation of data that helped confirm or dispel patterns that seemed to be emerging. Examples of the transcriptions, a complete code list, and select event lists are provided in Appendices F, G, and H. Analysis will be described here in three primary sections: the expert group, students, and comparisons between various experts and students.
The experts

Figures 4.10 and 4.11 include the coded event lists for two of the experts. Expert 3 demonstrated the most fully developed and accurate mental model through both the procedural protocol and the card sort. The event list shows not only a straightforward sequence for solving each task (CP), but is also embedded with lots of structured thinking about the problem (SMC), multiple ways to approach the task (PSC), and rich comparisons with his long years of experience (RE). All tasks are completed successfully (CR) and with confidence (Con). He is not having to figure out how to solve each task, but is interested in actively making connections as he proceeds, and the overall impression given is of an individual who is in analysis/synthesis mode, purposefully, even painstakingly evaluating and connecting information from both past and present. This seems to promote a deeper, more deliberate, and highly structured mental model that is applicable to a wide variety of novel situations. Even though he was quite familiar with the Pro Tools software, it seems evident that he could quite easily figure out any other new system. One example demonstrating this point is the fact that this expert was also teaching a senior-level course at the college. He was completely unfamiliar with the complex recording console used for the class, yet was easily able to find his way around on his own.
A brief explanation of the event list can aid the reader in understanding these data displays. The transcriptions were not broken down into discrete steps that would be defined by actions in the software, but rather as processing events, or episodes, that reflect what the subject is undergoing at a given moment. So, for example, in Expert 3’s task 1 event list there are two CP (*correct procedure*) events, separated by an SMC (*structured model concept*). As defined in the code list (Appendix G), CP represents a correct procedure performed in the software. However, this may represent a series of actions that are performed without interruption in thought by the participant. Thus the two CP codes included in this example represent several steps performed in order to complete the task requirements. The objective was not to merely list software-related activities, but to document what the subject was concentrating on as they performed each procedure. So the SMC code in the middle represents the subject describing in his head a structured (accurate) model concept that is guiding his procedures in the software. This typically meant subjects were recalling/affirming their models as they worked through
the procedures. Two or more codes assigned simultaneously represent an episode that is a mixture of these events and not easily segmented. For example, the subject would be stating software procedures as they also spell out their model concept, ensuring that their procedures follow what they believe to be correct. See Appendix G for the complete code list with definitions.

Expert 2 produced a very different event list. As Figure 4.11 illustrates, mental processing was less than that of Expert 3, resulting in fewer discrete events to be coded. Verbalized model concepts (SMC) are rare as this subject merely performed the necessary procedures (CP) accurately (CR), with confidence (Con), and without interruption; the notable exception was task 3 which represented a situation she does not perform in her work, requiring her to slow down a bit to think it through. Overall her tasks were largely completed in rote fashion, simply recalling her years of experience performing identical or similar tasks in the Pro Tools system. She was thoroughly familiar with the software and only had to think occasionally about tasks that she did not perform regularly in her job. Minimal effort was expended to complete each task with little or no commentary or reflection. Automation and speed are the keys to her work, and that is what she has developed over the years. It would make an interesting comparison to have this expert perform the same tasks with a different recording system in order to test her model more aggressively. If the system was sufficiently different, then it is possible the fundamentals of the generic model might not be developed or rich enough to sustain her as is the case with Expert 3. This conclusion is supported by Expert 2’s card sort, which was largely correct, but contained a few anomalies and inaccuracies and which was biased more toward a Pro Tools interpretation of the generic model concepts.
Expert 1 had extensive experience with both analog and computer-based systems, but had never seen Pro Tools before. This was an excellent opportunity to see how an experienced engineer approached a completely novel situation. Figure 4.12 is a process flowchart for the first task derived from the coded event list. This task requires considerable time and effort for him, involving lots of analysis of the software, incorrect attempts, wrong turns, and finally an incorrect result. Note how many events result from him trying to figure out the rather arcane software interface (dog-eared boxes). However, throughout the process Expert 1 demonstrates a solid grasp of the model concepts involved in the procedures, referring back to these time and again as he tries several avenues. Each time he hits a roadblock, he regroups conceptually before trying a different procedural route. Though he finally gives up on task 1, it is clear that his model is highly structured and adequate—it is simply a case of learning specifics of this software program and how they relate to the generic recording system model. In the case of Pro Tools, some functions are cryptic enough so as to require even an experienced engineer to consult the operating manual as there is no way to tell otherwise.
Figure 4.12. Expert 1 processing flowchart for task 1.
As he gradually figures out the software interface, making reasoned assumptions about specific Pro Tools functions (based on his solid operating model), the required processing and procedures decline significantly, even though the tasks required increasingly more complex thinking about the signal routing. He is also successful in attaining correct results on the last 3 of the 5 tasks—a noteworthy accomplishment. Though it took laborious effort, his model was sufficient to sustain him as he plugged away at the problem, the goal clear in his mind.

Figure 4.13 shows Expert 1’s overall pattern of model quality, processing involved in figuring out Pro Tools, procedure attempts, and results. The chart reveals no partial (PMC) or inadequate model concepts (IMC)—his mental model that guides his work is highly structured (SMC). There is considerable activity analyzing Pro Tools functions (A Soft) and figuring how to complete procedures within the software (A Proc). Overall, Expert 1 completed more procedures correctly than not (CP), and several of the incorrect procedures (IP) were software issues and not model issues (IP-CC). As he searches through the software interface he looks in likely places (targeted search—TS), resorting to random searching (RS) only occasionally when he has no idea how Pro Tools handles a certain function. Most telling for Expert 1 is that he got 3 out of 5 tasks correct (CR) in spite of issues with the software (IR-CC).
Figure 4.13. Expert 1 results. Bar heights represent the number of occurrences present for each type of coded event.
Summary of results for research questions 1 and 2

What mental models do professional, expert recording engineers possess about recording systems?

Although each of the three experts demonstrated different model aspects, there were common elements revealed from the study exercises. Both the card sorts and procedural tasks indicate relatively well-structured, accurate mental concepts of how recording systems operate, interrelate, etc. Experts 1 and 3 exhibited this the most, while Expert 2 showed a somewhat less well-defined structure in her card sort model. Her rote performance in Pro Tools, matched with the limited variety of her daily work experience, seems to match her card sort model in terms of limited awareness, less depth and breadth, and lacking enough diversity that would enable transfer in a truly unique situation. By contrast, the other two experts possess and often referred to their extensive experience. This seems to indicate a necessity for a wide variety of experiences in order to develop a rich model capable of handling novel problems. The two most proficient experts also had extensive background with analog systems, which are much more transparent for understanding operations and signal routing. This seems to be a major factor in how these individuals approach and attempt to understand software-based recording systems. This begs the question: is it possible for a new engineer to learn exclusively on software-based systems and develop adequate models that can be applied to other systems? This study cannot fully address this important issue which has major implications for audio education. It is the belief of the investigator after years of teaching that initial, ample analog exposure and experience provides a highly structured, grounded awareness that can then be best applied to virtual software-based, digital systems. The results from the
three experts seem to support that idea, but a different study design is required to completely test it.

Reviewing the results from Expert 3’s card sort reveals a higher-level, hierarchical conceptual awareness that allows him to transfer between analog and virtual systems. As described in that section, his approach and terminology make sense operationally, but can be quite confusing for a novice. This is an example when long experience not only develops a more accurate model, but one that possesses hierarchical, hyper-structurally organized concepts.

Another factor that seems to contribute to model development is the individual him/herself. Expert 3 is obviously a deep thinker, analyzing everything he does in order to make new connections. Contrast this with an individual who simply learns how to perform the work without a great deal of thought; their model would most likely be much less structured, less rich, and even insufficient for transfer purposes. Expert 2 falls into this category to some extent, although this is quite relative. Expert 2 is a highly intelligent, capable engineer. Compared with the other two experts in this study, however, one can clearly see differences in their background, approach, and revealed mental models.

*How do these engineers apply this structural knowledge specifically to Pro Tools recording systems?*

Expert 3, though experienced and confident using the Pro Tools system, does not merely complete the tasks quickly by rote. He takes his time and constantly thinks about what is happening, considering alternative approaches and making connections to analog
systems and real recording sessions he has been involved in. He is always interested in learning. During the tasks he would occasionally stop and figure out a new method or shortcut for performing an operation in the software, thus adding to his knowledgebase. It is clear he has a solid mental model of how recording systems operate and interconnect, and he applies this consciously to his performance in the software.

Expert 2 applies her structural knowledge almost unconsciously. Her knowledge and capability in Pro Tools is so well automated she does not have to process much at all. Interestingly, Expert 2 has never taken time to learn many of the available keyboard shortcuts, which allow the user to work faster in the software. She has learned what is necessary for getting the job done. This approach does not lend itself to seeking new, alternative methods or approaches, thereby limiting the potential for further model development.

Figure 4.14 helps illustrate the difference between Experts 2 and 3 in how often they brought in outside experiences (RE) and considered current task comparisons (TC) within the study exercises.
Figure 4.14. Experts 3 and 2 integration of experience and task comparison.

Expert 1’s performance is the most revealing for how an experienced professional can apply a well-developed mental model to novel situations. He knew nothing about this particular software, but his solid, grounded model was sufficient to carry him through tedious, multiple trial-and-error attempts. His familiarity with other software systems also helped target his searching through the interface as he looked for similar representations of functions. He was very methodical and confident in his overall assumptions about what needed to occur, doubting not his knowledge or judgment about the task concept but rather how to interpret specific software functionality.
The students

The seven student subjects exhibited a wide range of performance characteristics and task completion success rates. The initial intent of the study was to determine the general accuracy and adequacy of student models as a product of their instructional program at the college. Overall, the procedural task results indicate a cautiously positive outcome. A majority of the students had fairly well-developed models that, although sprinkled with inaccuracies and vagueness from time to time, generally contain the basics of generic recording systems. There were no students whose model was completely inadequate to handle all the tasks presented in the study exercises, but a few students exhibited uncertainty in the procedural tasks that are supported by their weak performance on the card sort. There are multiple likely causes for this that will be discussed throughout this chapter as well as in Chapter 5. (See Figure 4.15.)

**Student model quality**

Figure 4.15. Quality of student models evidenced from protocol data.
Aside from the few instances where the conceptual model was either inaccurate or misapplied, the main issue that arose repeatedly was the software itself. There were many incorrect results based on correct concepts (IR-CC), meaning that students seemed to have the right model concept to solve the task, but either misunderstood the Pro Tools interface or could not figure it out. (See Figure 4.16.)

![Students: Correct vs Incorrect Results](image)

Figure 4.16. Student results: Correct, incorrect, & incorrect but with correct model concepts.

For example, consider Student 5. Figure 4.17 shows a comparison between his demonstrated model quality and task completion results. All verbalized model concepts were structured and accurate (SMC), and there were no partial (PMC) or incomplete model concepts (IMC). He successfully completed only one task (CR), but the remaining four he missed were due to issues understanding the software rather than not knowing how to solve the problem (IR-CC).
Another way to examine Student 5’s situation is to compare his results (CR, IR-CC) with whether his attempted procedures were correct or not. Figure 4.18 indicates that all of his procedures were either correct (CP) or, if incorrect, due to a misunderstanding of the software interface (IP-CC). His model is sound for a novice; he had extensive experience not only in the college’s recording studios but also through a very successful internship at a major recording studio in New York City. He knows recording systems quite well and was very confident about this performance in the exercise, but simply misinterpreted some of the functions in Pro Tools.
Two students performed very well on the tasks, but perhaps for somewhat different reasons. Both of these individuals had more experience with Pro Tools than the other students, providing them with more familiarity in understanding the software. They had performed some of the tasks before, but not all of them, so the initial assumption was that both their generic mental model of recording systems as well as the extension of their model specifically related to Pro Tools enabled them to figure out the novel problems. The challenge during analysis is that the more familiar a subject is with the software the
less overt processing is required to perform the task, thereby making it more difficult to extract the underlying model. Referring back to these students’ card sorts, it is further seen that Student 1’s approach to the sort was very non-operational, which does not match his proficient protocol performance. In fact, Student 1’s sort was one of the cases eliminated from the “stronger student” MDS map. Student 4, on the other hand, completed a card sort that more closely resembled his excellent performance in the software, indicating that his underlying model seems fairly solid for a novice at this stage.

Figure 4.19 shows Student 1’s procedural model quality (from the protocol data, not card sort) compared to his task results. Only on one task did this student seem to exhibit a partially understood model concept, resulting in incorrect procedures that were based on generally correct concepts. Otherwise his performance is flawless through the other tasks. Does this indicate a solid model? Student 1 completed the tasks so quickly and efficiently there was relatively little conceptual thinking in his verbalizations to confirm either way. He seemed to understand what he was doing conceptually, but considering the low-level organization of his card sort exercise it seems possible he has become comfortable with various procedures and is perhaps not so aware of the underlying model. The investigator’s knowledge of this student indicates that this is a likely possibility as he was not the type to spend time thinking through and analyzing issues. His almost flighty attention to the card sort and think aloud protocol tasks reflects this as well. This merely reinforces what has been stated throughout the literature—mental models are messy, complex, and difficult to clearly understand.
Figure 4.19. Student 1 model quality compared to results.

Not all students performed as well; there were three students whose only experience with Pro Tools had been a year earlier in class. Remembering how to accomplish certain tasks in any software program requires regular use of the program, something these students had not done. Therefore, the goal during analysis was to distinguish between issues with their mental models as compared to not understanding the software. Reviewing their card sort results indicated large gaps and errors in their structural knowledge. The MDS map produced from these three students’ sorts was quite scattered, with limited coherent organization. To their credit, all three of them have at
least some understanding of the conceptual issues, but their models are vague and inexperienced enough to undermine their ability to figure out all of the tasks.

Student 3 was a very bright individual who had relatively less experience in the college’s recording studios. His interest and proficiency was primarily in computers (but not Pro Tools), so it was informative to include him in the study to see how well he managed the Pro Tools system. As indicated in Figure 4.20, his results are scattered throughout the coding spectrum. His applied model is not completely solid, but portions are intact much more often than not (SMC, PMC, IMC), he spends a great deal of time searching and processing the software to figure it out (TS, RS, A Proc, & A Soft), and he has more incorrect procedures than correct (CP, IP, IP-CC). His results, however, are encouraging in that in spite of his lack of experience and structured model, enough is there to allow him to either get most of the tasks correct or incorrect because of software issues (CR, IR, IR-CC).
One last student example serves to show what happens when a model is largely insufficient to the task. Student 7 had the general idea of what was supposed to happen in most cases, but her model was inaccurate and vague enough to prevent successful completion of all the tasks. Combined with her limited experience with Pro Tools, this resulted in lots of software processing, model processing, incorrect procedures and incorrect results (see Figure 4.21).
A brief look at this subject’s event list is also quite revealing. Exactly opposite that of Expert 1, this student knew enough to easily complete the first simple task. However, as the problems became more complex her model became increasingly insufficient to support adequate performance, resulting in a great amount of struggling, uncertainty, and incorrect results (Figure 4.22).
This individual was an excellent student overall, but was more interested in songwriting and performing than engineering, so her extra time was not spent in the studio practicing recording techniques. Thus any knowledge gained in earlier classes was not reinforced sufficiently to sustain and develop the model. Her card sort results bear this out as well, so again triangulation between both research techniques proved useful in deriving an accurate portrayal of her mental model.
Summary of results for research questions 3 and 4

What mental models do graduating seniors possess about recording systems?

In this particular study of students who had largely completed the instructional program at the college, they seemed to possess at least a basic understanding of system models, though the level of accuracy and model perspective varied. There were a few strong performers whose card sorts and Pro Tools procedural protocols revealed fairly accurate, structured models of recording systems. Their MDS maps and protocol performance was very similar to that of the experts. These individuals shared a few common characteristics: they are bright and easily understood the technology studied throughout their coursework, they spent considerable time in the recording studios in addition to that required for class projects, they had used Pro Tools more than what was required in class, and they shared a deep personal interest in recording technology and procedures.

Many student card sorts revealed more of a conceptual organization rather than an operational approach, and hardly any demonstrated a hierarchical perspective. Some subjects did not group functional items together, such as aux send > post-fader > effects > aux return. Cue mix should include the pre-fader function and possibly monitor path. Instead they see aux sends and returns, with all related controls such as pre- and post-fader, as one category and effects in another. One of the best protocol performers (Student 1) organized his card sort by grouping anything related to aux send and aux return in one pile, a decidedly non-operational approach. A few times subjects grouped functions based on their physical proximity on the console/system. This rarely has any direct relationship to functionality in system design. These approaches do not reflect what
was presented during instruction, and there could be any number of reasons as to why this occurs. Why would the strong protocol performer complete such a non-operational card sort? His protocol data suggests the presence of a fairly accurate model. Is it that students tend to segregate the “theory” studied in class from the “real” hands-on application? Could he be familiar enough with procedures so as to “talk the talk”, but yet not quite understand conceptually how the model is structured? In any case, the solution is probably to reinforce functional patterns based on their conceptual relationships, presented visually and supported with lots of guided hands-on experience to make the connection to meaningful applications.

Even novices with strong models suffer from simple lack of experience. They do not realize issues as quickly as an experienced professional would. Student 4, one of the strongest performers, is an excellent example of this as he correctly completed the cue system task (task 3), but then tried a more complex setup following a method that could not work—it violated the model of how aux sends and cue systems work in a recording system. If he had spent a little more time thinking about it he most likely would have seen the problem with the concept, but did not at the time.

Strong students are similar to experts in that if they already have the conceptual plan from their mental models, they do not have to develop one for each situation. They simply “run the model” and derive a solution. Weaker performers must sift through their model to see what might work, then figure out the specific procedures in the system.

Novices are also prone to doubt their models as they face unexpected and unfamiliar situations. Though it took a long time to attempt figuring out the software,
Expert 1 never doubted himself, but instead applied his processing efforts to the software interface.

*How do these novices apply this structural knowledge specifically to Pro Tools recording systems?*

The strong performers not only knew basically how to use Pro Tools based on experience with the software, their analog model was sufficient to facilitate transfer of various operations to the Pro Tools system with only a few misperceptions about how the software operated. This is evidenced in the tasks that were unfamiliar to them—either they had not performed the task in Pro Tools or had never performed it at all. Their knowledge of the program was sufficient (mostly) to figure how to apply the conceptual model they believed was correct for the task.

Even a majority of the individuals who could not complete a task in Pro Tools at least had some correct concept of what needed to be done. In other words, their model was present—it was transfer in this particular situation that was ineffective. This is due primarily to the cryptic interface design that unnecessarily complicates understanding of Pro Tools. Partly, though, this could be attributed to the relatively large leap between a very discrete, analog model to that of a virtual system. Pro Tools not only provides the same signal routing capabilities as analog systems, its virtual nature allows for a flexibility not possible in the analog realm and is therefore overwhelming to someone who cannot conceptualize it in their mind. If an individual does not have a solid grasp of signal flow concepts, then it is far more difficult to determine solutions to novel situations in the software.
Along the same lines, it was evidenced a few times where the interface had obliterated the analog model, leaving the student to pursue incorrect concepts and procedures. Student 3 is a prime example of this, where at one point his analog model should have kicked in for a procedure they had performed many times in the studios, but the terminology of the software seemingly skewed it to the point where he simply became confused, unable to apply his now-distorted model.

Overall, vague and incomplete mental models lead to a less structured approach to problem solving. Student 6 is an example of this where she simply wanders around trying to figure out everything—the big picture, the procedures, and the software. The analogy is that of a medical intern who must review their complete notes in the attempt to formulate a diagnosis, whereas the attending physician simply “jumps to” the correct relevant chunk of information to pronounce an answer. Long experience helps build and organize an effective mental model. While novices do not have extensive experience to draw upon, the objective in instruction is to provide as much experience in a variety of situations as is possible in the attempt to mimic the status of an expert.

*Experts and students compared*

To further illustrate some of the points made in this section, a few cross-case comparisons are presented. Figure 4.23 displays model qualities for all subjects, revealing the presence of partial and inadequate model concepts among some of the students. Note that the specific height of any subject’s bar as compared to other subjects is insignificant, only the relative balance for each subject is used for analysis. Thus Expert 1’s structured model concepts (SMC) are not any better than Expert 3; he merely referenced model
concepts much more often as he worked through the software. In the case of Student 7, this individual exhibited several cases each of structured, partial, and inadequate model concepts as she worked through the tasks, representing a more incomplete, inaccurate, and vague model as compared to the experts.

Figure 4.23. Expert & student model qualities compared.

Expert 1 was the experienced engineer who had never seen the Pro Tools system; his event list reveals a long, laborious attempt to figure out the software in order to solve the first few tasks. However, it is useful to compare this expert with some of the students
who also labored through the exercises, albeit with less structured mental models. Following are a few data displays that visually depict this expert/novice difference.

First, comparing the event lists between Expert 1 and Student 7 shows a clear inverse pattern: Expert 1 requires considerable time to figure out the software for a very simple task in Pro Tools, but then quickly adjusts and is able to complete the more complex tasks more efficiently. Student 7 is easily able to complete the first task applying her memory from class, but as the tasks become more involved she gets bogged down, her model not quite sufficient to solve the problems.
Figure 4.24. Expert 1 event list curve showing decreasing difficulty.
Figure 4.25. Student 7 event list curve showing increasing difficulty.

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP / SMC</td>
<td>IMC</td>
<td>SMC</td>
<td>TC</td>
<td>SMC / A Proc</td>
</tr>
<tr>
<td>Comp / CR</td>
<td>ES</td>
<td>A Proc</td>
<td>PMC / A Proc</td>
<td>IMC</td>
</tr>
<tr>
<td>IP</td>
<td>CP</td>
<td>IP</td>
<td>A Proc / RS</td>
<td>A Soft</td>
</tr>
<tr>
<td>A Soft / IMC</td>
<td>A Soft</td>
<td>UE</td>
<td>SMC / A Proc</td>
<td>A Soft</td>
</tr>
<tr>
<td>IR</td>
<td>A Proc</td>
<td>A Soft / PSC</td>
<td>A Soft</td>
<td>IP</td>
</tr>
<tr>
<td>A Soft</td>
<td>PMC / A Proc</td>
<td>IP</td>
<td>A Proc</td>
<td>A Soft</td>
</tr>
<tr>
<td>UC</td>
<td>IP / CC</td>
<td>A Soft</td>
<td>SMC / RS / A Proc</td>
<td>IP</td>
</tr>
<tr>
<td>A Soft</td>
<td>CP</td>
<td>SMC</td>
<td>IP / CC</td>
<td>A Soft</td>
</tr>
<tr>
<td>IR-CC</td>
<td>PMC</td>
<td>SMC / A Proc</td>
<td>IP / CC</td>
<td>A Soft</td>
</tr>
<tr>
<td></td>
<td>IR</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.26 allows a comparison of various aspects of the protocol performance between Expert 1 and Student 7. First, Expert 1 evidenced no partial or inadequate model concepts as compared to Student 7’s vague understanding (SMC, PMC, IMC). Expert 1 spent considerable time processing both the procedures and interface in the software (A Proc, A Soft) whereas Student 7 did not have the ability to continue the effort—her model was not sufficient to provide multiple approaches to solve the problem within the software. Also note the inverse relationship between targeted search and random search (TS, RS). Expert 1 was able to apply his model and familiarity with general software systems to look in likely places for certain functions. Student 7 had little of this, relying on more random searches of the program to find something that might look familiar. Finally, Student 7 had one correct result (CR), but mostly incorrect results. Some of these were due to software issues (IR-CC), but just as often due to inability to comprehend the appropriate conceptual approach to solve the problem (IR).
To provide a general summary of how both experts and students performed on the tasks overall, the final results and model qualities can be displayed. Figure 4.27 shows that a majority of the subjects were able to apply either structured or at least partially structured models to the tasks. Figure 4.28 indicates that the final results did not correspond as directly to the model qualities—instead of mostly correct results as would
be expected from mostly strong mental models, many subjects had issues understanding the software itself. Implications from this will be discussed further in Chapter 5.

**Expert & Students Combined: Model Quality**

Figure 4.27. Combined expert and student model qualities.

**Experts & Students Combined: Correct vs Incorrect Results**

Figure 4.28. Combined expert and student results.
CHAPTER 5
GENERAL DISCUSSION

Factors involved in mental model development

This study set out to examine four main questions in order to determine whether students at this institution were developing appropriate mental models of recording systems. The literature and the outcomes from this study make it clear there are numerous factors that influence the development of an individual’s mental model of a particular subject domain.

Importance of experience for model development

The experts in this study benefited from years of experience in a variety of situations working with different types of systems. The better performing novices also were distinguished by having spent more time in the recording studios than the others. Not only does this promote automation of common procedures, but each novel situation forces consulting and expansion of the model as new information, processing, and results are assimilated. The level of accuracy of the model as it develops comes through 1) grounded awareness of the fundamentals that underlie more complex situations (Norman, 2002; Jonassen et al., 1993; Parush, 2004; Bransford et al., 1999), and 2) a large variety of similar, but not identical, experiences for reinforcement and novelty (Salomon & Perkins, 1989; Johnson-Baird et al., 1998; Bransford et al., 1999).

Highly structured mental models are fundamental for expert development (Anderson, Spiro, & Anderson, 1978; Kuipers & Kassirer, 1984; Kieras & Bovair, 1984). This structure comes from accumulation and assimilation of new knowledge and, more
importantly, meaningful connections. Experience is the key, and the result is that experts can apply relevant information to a new problem with a higher level perspective that comes from their extensive experience in a variety of situations. Novices have no such streamlined, targeted processing abilities and must sift through a great deal of information that may or may not be applicable or even present in their knowledgebase (Johnson, 1988; Jonassen & Henning, 1999; Gentner & Stevens, 1983).

**Accurate models depend on appropriate system models during instruction**

Though some of the students possessed fairly structured models of recording systems, in general there were numerous uncertainties and inaccuracies in spite of the instructional emphasis on fundamental principles. The challenge for teaching novices is to help them develop accurate, though relatively simple, models that are fundamentally sound (Mayer, 1989; Coll & Treagust, 2003; Kieras & Bovair, 1984; Fiore, Cuevas, & Oser, 2003). The key is to help them learn correct fundamentals in order to reduce misunderstanding while providing extended experiences that force them to see when their inadequate models will not work (Clement & Steinberg, 2002, Norman, 2002). These correct fundamentals come from presentation and reinforcement of appropriate system models during instruction. This increases the likelihood that students will base continued experiences and instruction on a grounded foundation. If left on their own, individual perceptions will dominate model development, almost ensuring inadequate and inaccurate system models that will hamper performance (Norman, 2002).

Instructional models must be simple enough for the novice to understand, yet accurate and faithful to the larger system or content area so that learners can fill in the
details as their structural knowledge develops. Solid foundations can better facilitate
transfer to more complex and novel problems, but the experiences must be authentic and
extendable to real applications. Many instructional design theories support this concept:
microworlds, elaboration theory, constructivist learning environments, and goal-based
scenarios to name a few (Reiber, 1992; Reigeluth, 1999; Jonassen, 1999; Schank, 1999).
Mental model research supports the notion that individuals’ models “seem to be simpler
than formal mechanics” (Gentner & Stevens, 1983, p. 53). In other words, complex
system images are not likely to coincide with the models people tend to develop, thereby
creating the potential for inaccurate understanding and use of the system (Norman, 2002).
However, it should be cautioned that over-simplifying the instructional model has the
potential to cause confusion when applied to the more complex, complete system (Seel,
1999), again reinforcing the issue of not sacrificing accuracy and applicability in the
interest of accessibility for novices.

Accessibility of system models is often enhanced through use of analogies
(Clement & Steinberg, 2002; Brandt, 1999). By presenting a new concept as similar in
some way to something already understood, learners can more easily identify with and
begin to understand the new material (Bransford, 1999; Perkins & Unger, 1994). A
common example is that of using water pressure and flow to explain how electrical
voltage and current works (Gentner & Gentner, 1983).

Significant concepts of the system or subject need to be clearly presented.
Instructional materials and methods should make meaningful connections of the system
explicit to the learner so as to help encode these into their models (Mayer, 1989).
Mental models are unstable

Models require use. Briefly learning a concept and/or procedure and then not practicing it for some time will cause erosion of whatever connections might have existed. The students who performed most poorly on the card sort and software protocol had not spent much time in the studios, nor had they used Pro Tools for about a year. They were then forced to recall information from long-term memory that had not been utilized. These chunks of information are fragmented, often inaccurate, and largely devoid of meaningful relationships (Norman, 1983).

Accurate mental models require active thought and attention

Deep thinking seems to encourage development of more highly detailed and structured models. Active processing has been shown to make a significant difference in learning environments—passive reception of inert knowledge fails to “stick” and enable transfer to other situations (Salomon & Perkins, 1989; Bransford et al., 1999; Perkins, 1992). The students who during their coursework had demonstrated the ability and interest in thinking through various concepts about recording seemed to perform better overall. Those who did not seem to “dig in” as much tended to show more inaccuracies and less structure in their models. Expert 3 is a deep thinker who labored over his tasks, considering different options and recalling prior experiences. Expert 1 is very similar in this respect. Student 5 took seriously the challenge of learning all he could about the recording process throughout his degree program, and his deep awareness of the field is evident both through his card sort and the procedural protocol.
Personal interest fosters active mental model development

Active thought processing most often occurs when the subject area is personally relevant and meaningful to an individual (Perkins, 1992; Bransford et al., 1999; Schank, 1999, 2000). Personal interest in the process is a critical factor—students who want to learn recording will apply their attention to the tasks. Some students are not as intrinsically motivated, resulting in higher instances of incomplete and incorrect understanding of material from class. The comparison between Experts 3 and 2 also supports this; while Expert 2 is very capable at what she does with Pro Tools at work, she is not as driven to learn more, to apply her knowledge in different situations. Expert 3 is very interested in the process, carefully and deliberately looking for connections, new information, etc. As David Perkins asserts in *Smart Schools* (1992), “people learn much of what they have a reasonable opportunity and motivation to learn.” (p. 45). Thus personal interest and hands-on application are key for students who want to develop adequate mental models of recording systems and processes. The students who performed better than others demonstrated a stronger personal interest for recording systems; not only did they apply themselves in classes, but they also spent additional time in the studios and, in some cases, on their own Pro Tools systems they had purchased.

Thus there are numerous factors that contribute to mental model development. Each individual possesses a unique combination of these issues, resulting in a wide variation in how novices and even experts perceive any content area. This causes tremendous implications for educators as they attempt to develop adequate instructional environments. Figure 5.1 summarizes the points discussed here.
Issues with the Pro Tools system

Mental model research originated primarily from the two disciplines of cognitive psychology and human-computer interaction. A great deal is known about how systems should be designed in order to facilitate user operation, but that does not ensure those lessons are always implemented. For one, it is quite difficult to design a complex system such as software that provides the power and flexibility expected of the user, yet presents affordances to the user that clearly indicate how it operates (Norman, 2002; de Kleer &
Brown, 1983; Kieras & Bovair, 1984). The system image presented to the user must match the designer’s image of how it really works or problems will occur (Norman, 1983, 2002). Similarly, a system that violates traditional operational paradigms of similar systems will prevent transfer of previous knowledge—in other words, it violates the user’s mental model of how such a system is expected to work (Norman, 2002). de Kleer & Brown (1983) concluded “there is much evidence that suggests that the major mode humans use to “understand” a complex device involves recognizing structural patterns in the device whose functionality is already understood” (p. 189).

If an engineer understands fundamentals of audio signal flow they should be able to apply that to a virtual digital or software environment. This depends entirely, however, on the design of the virtual system and how faithfully it follows traditional analog system models of operation (or even simple logic). During the design of this study, one of the primary expectations was that the Pro Tools software was a cause for concern. The interface design is a curious blend of obvious, easy to understand functionality and cryptic, meaningless controls. Once an individual learns the software this is not a problem, but there are numerous issues with the design that prevent even an experienced engineer, such as in the case of Expert 1, from confidently figuring out how it works. In the case of novices it is nearly hopeless as their model is often not strong enough to make the leap. To make matters worse, many educational institutions are installing Pro Tools systems as their primary (or only) lab system for students to learn with. Comparing the overall model qualities with actual results clearly reveals the problem: there were a disproportionate number of incorrect (IR) and incorrect with correct concept (IR-CC) results to the overall high percentage of strong models (SMC) (see Figure 5.2). Many
subjects did not complete the task correctly only because they misinterpreted the software either because of functionality vagueness or terminology issues. One of the strongest students knew exactly what he was doing except for the non-obvious difference in how Pro Tools presents signal routing options for aux sends and audio track outputs—they look identical in the software, requiring the user to simply know which to use. Other students misinterpreted the arcane difference between an “audio track” and an “aux track” in Pro Tools. Pro Tools uses faders for aux sends whereas analog consoles employ rotary pots for aux sends—faders are for primary audio channel level control. One student summed up his frustration with the wry statement “something simply labeled aux send would be handy.”
Much of this can be overcome by continued study of the operating manual and trial-and-error practice. Retaining this information, however, requires constant use. Norman (2002) maintains that a balance between knowledge in the head and knowledge in the world must be struck, meaning the system should provide much of the information required for operation, thus freeing the user to concentrate on performance. The user should not be expected to memorize meanings of controls and obscure functions as much as Pro Tools requires. If a recording system does not present available options clearly
enough then engineers will either perform procedures incorrectly or assume they are not able to accomplish certain tasks. Their flexibility to solve novel situations is accordingly diminished.

**Implications for instructional design**

Several conclusions can be offered as suggestions for improving instructional design of audio recording educational programs. Though the design and scope of the current study is limited to that of the investigator’s institution, it is presumed that these ideas are broad enough to be generalized to many audio recording educational programs. Fundamentally, these can be applied to most any instructional situation.

*Design effective system models for instruction*

Appropriate system models must be presented during instruction. These must clearly describe and visualize the operating patterns contained within general concepts. It is not sufficient to simply show students how to accomplish certain procedures on a recording system and assume they see the patterns. It is also quite easy for an experienced instructor to forget that students do not share the same level and amount of experience and knowledge about the field. Therefore concepts that may seem obvious and common are not nearly so for the novice. Expert 3’s card sort revealed a conceptual model that makes sense operationally, but would be very difficult for a novice to grasp. His model was hierarchical and transcended analog and digital system awareness, a status not yet within reach of novices.
Learning tasks must be authentic and meaningful

There has been much research and discussion about making learning meaningful for the learner and authentic to the real world. Decontextualizing instruction does not equip students with cognitive methodologies to handle new situations and solve real problems. Numerous theories and concepts have been developed with this in mind such as constructivism, constructivist learning environments (Jonassen, 1999), goal-based scenarios (Schank, 1999), open learning environments (Hannafin, Land, & Oliver, 1999), cognitive flexibility (Spiro et al., 1992), problem-based learning (Savery & Duffy, 1996), microworlds (Rieber, 1992), anchored instruction (Bransford, 1990) and situated learning (Anderson, Reder, & Simon, 1996).

Johnson (1988) compared troubleshooting performance between experts and novices. One recommendation for instruction was that instead of introducing electronics by studying standard electrical theory equations, students should first begin to understand what a circuit is and does. This qualitative perspective provides an abstraction that then facilitates application of quantitative reasoning, i.e., solving equations.

“Rote learning causes problems” (Norman, 2002, p. 68). Rote learning is decontextualized, so there are no meaningful purposes for the learner. Memorized procedures do not provide information that can help solve problems or figure out different situations.

All of these point to the need for developing learning environments that have meaning, purpose, and are of interest to the learner. In the field of recording this is easily implemented as students can complete recording projects in the lab throughout their coursework. These projects should be grounded in the instructional system models,
reflect practices in the industry, and provide opportunities for students to explore, create, and come away with products they can enjoy and use.

Instruction must provide extensive, guided practice

Students must have opportunity for extensive, guided practice with the systems (Bransford et al., 1999; Salomon & Perkins, 1989). Merely spending lots of time in the lab is insufficient—in order to develop accurate models this experience and exploration must be guided and supervised by a knowledgeable instructor, otherwise misperceptions and bad habits will become embedded into the students’ models. These practices are very difficult to correct, and any misunderstanding of the basics will carry through unless discovered and resolved early (Norman, 2002; Parush, 2004).

Instructional experience must provide variety and novelty

This experience must take place on a variety of system types, preferably both analog and digital. In the investigator’s own institution, students are initially exposed to an analog console that, while relatively complex, is accessible for the novice with proper model presentation and description. They then transfer this knowledge and experience to a more complex analog console before finally working with software-based systems such as Pro Tools. Judging from years of observation, fine-tuning, and the results of this study, this approach appears to be working satisfactorily for most students who are personally interested in developing as recording engineers.

Instructors must be cautious in their choices of projects and assessment design. When a user knows exactly how to perform a task, there is little overt thought processes
involved—in other words, they do not have to run their model to figure it out. One cannot assume in such a case that the learner knows the concepts behind the learned procedures. In order to draw this framework into the open, learners must be taken beyond familiar tasks to force transfer to occur.

Applying mental model analysis in the classroom

The procedures undertaken for this study were successful in triangulating desired outcomes. However, performing both of them is very time consuming; the procedural think-aloud protocol alone is not feasible for most instructors. The card sort, on the other hand, is very simple to set up, easy to administer, and just as easy to review in order to gain insight on how learners are mentally organizing subject matter content. While the multidimensional scaling maps were useful as an analysis tool, merely scanning the sort results is sufficient for getting at least a general idea of how students perceive course content. Instructors would benefit greatly from this approach, even if applied only occasionally. An informal approach to the procedural protocol would also be quite revealing, consisting of nothing more than individually observing a student as they work through different tasks. Thinking patterns, knowledge of the subject, and issues during application can only be truly observed in such a situation. A third approach is described by Glynn (1997) where students draw their mental models of the instructional content simply to get an idea of what they each “see” in their minds; this can be an effective diagnostic tool for determining how instruction should proceed. Gray (1990) similarly had subjects draw their mental images of how an early hypertext retrieval system was organized. Standard written assessments cannot provide the same perspective of what a
student actually believes in their mind—what their mental model looks like. It can be quite revealing for the instructor and is well worth the effort required.

*The use of the Pro Tools recording system for instructional applications*

Finally, it is crucial that educational institutions do not employ Pro Tools for early instruction. There is great pressure and practical reasons to do so—the entire professional industry uses Pro Tools, and so recording programs are expected to train students on what they will be using after graduation. However, this represents a near-sighted perspective. What the recording industry needs long-term are engineers who understand fundamental operating models on which all recording systems are based. These individuals are then able to transfer their knowledge to other systems that they happen to work with. If engineers do not possess adequate mental models of such systems, then they resort to a step-by-step learning process for each different system, learning by rote and less able to apply concepts between systems.

**Implications for future research**

Though this study adequately addressed the investigator’s primary questions in terms of examining a specific instructional situation, there are numerous options for follow-up studies.

To effectively investigate students’ abilities to transfer their instructional system model to other virtual, software-based systems, it would be necessary to perform the think-aloud procedural protocol with a software application they have never seen. Ideally, a comparison would then be made between the card sort and Pro Tools protocols from
this study with their performance on a different system. Another option would be to identify a new group of students and duplicate the current study using a different software system instead of Pro Tools, thus eliminating any rote-learning experience from their coursework.

It would be informative to perform a comparison think-aloud protocol employing both a complex analog and a software-based system. These would have to be systems students are unfamiliar with, and the results should reveal model application issues between analog and virtual recording systems. Such results could inform the recording industry in terms of system design issues and the challenges of educating future engineers considering the rapid trend toward digital, virtual designs.

Along these same lines, mental model development and transfer should be studied with students who have only experienced digital and software-based systems, thus removing the analog paradigm altogether. Does it matter whether novices experience analog thinking and practice, or can they effectively learn a new paradigm? Would this new paradigm provide effective model consistency so that they can transfer between systems that are by their nature much more different than is the case within the analog environment?

As the interface design of Pro Tools is apparently a major issue for performance, a study could target specific problems with this system, perhaps in comparison with another software-based application.

Finally, the current study design could be applied to students at other institutions with similar educational programs. Each institution’s curriculum would be examined and compared along with instructional methods and student experiences in order to determine
possible patterns of model development (or lack thereof). Programmatic differences, however, would present a challenge for producing usable outcomes. This issue can be mitigated somewhat by careful selection of programs that are relatively similar to each other.

**Limitations of the study**

As explained in Chapter 1, the primary limitation of this study is its lack of complete generalizability to other instructional situations. This is due to the fact that each educational program is quite different in design and implementation. Many programs are music-based while others are not. Some are grounded in a liberal-arts environment while others are focused more on professional preparation in the field. Lab experiences provided for students varies greatly. This makes direct comparison very difficult, although the general concepts presented here should be applicable and helpful to other educators.

As there was only one investigator, a limited subject pool was selected. Although careful selection of participants contributed to what is considered a satisfactory outcome, the variety of possibilities is necessarily reduced when the entire relevant population is not addressed.

Investigator bias cannot be completely eliminated from the process. As the investigator was also the instructor for these students in the degree program, there is no way to be completely objective about the participants, the process, and the outcomes. However, in this case there were many positive factors that arose from a combination investigator/instructor situation. In-depth knowledge of the field, the educational program
and methods, and the participants themselves provided in-sight, background, and an expert perspective on the material that proved valuable in the study design, implementation, and analysis. Therefore the potential negative effects of personal bias were deemed minimal for this particular situation.
REFERENCES


Anderson, R. C., Spiro, R. J., & Anderson, M. C. (1978). Schemata as scaffolding for the 
representation of information in connected discourse. *American Educational 


of Science Education, 21*(1), 95-117.

Borgman, C. L. (1986). The user’s mental model of an information retrieval system: An 
experiment on a prototype online catalog. *International Journal of Man-Machine 
Studies, 24*, 47-64.


November/December*, 42-45.

Bransford, J.D. et al. (1990). Anchored instruction: Why we need it and how technology 


Glaser, R. (1989). Expertise in learning: How do we think about instructional processes now that we have discovered knowledge structure?. In Klahr & Kotosfky (Eds.), *Complex information processing: the impact of Herbert A. Simon* (pp. 269-282). Hillsdale, NJ: LEA.


Kinchin, I. M. & Hay, D. B. (2000). How a qualitative approach to concept map analysis can be used to aid learning by illustrating patterns of conceptual development.
Educational Research, 42(1), 43-57.


Retrieved February 12, 2005, from ZDNet White Paper Web site: http://whitepapers.zdnet.co.uk/0,39025945,60095117p-39000539q,00.htm


APPENDIX A. IRB APPROVAL

Informed Consent Form for Social Science Research
The Pennsylvania State University

Title of Project: Eliciting, analyzing, and comparing mental models of complex audio recording systems between professional and novice recording engineers.

Principle Investigator: Barry R. Hill
Associate Professor of Music
111 Blair Music Center
Lebanon Valley College
Annville, PA 17003
Phone: 717-867-6285
Email: hill@lvc.edu

Advisor: Dr. Susan Land
Instructional Systems
315 Keller Building
The Pennsylvania State University
University Park, PA
Phone: 814-865-0473
Email: sland@psu.edu

Purpose of the Study: The purpose of this research is to compare how professional recording engineers and students in an audio recording degree program think about recording systems. By comparing how these two groups approach recording procedures and, specifically, computer-based recording systems, audio classes and degree programs can be designed and/or modified to better teach students to think and work like professionals.

Procedures to be followed: There are two components to this study that you will be involved with. The first is a computer-assisted task called a card sort. You will be presented with a list of audio recording terms, and your task is to organize them in any manner that makes sense to you, essentially creating several piles on the screen. The second exercise involves a series of tasks to be completed within the Pro Tools recording system. You will be asked to set up various signal routing operations and simply state what you are doing and thinking as you work. Your proficiency and speed at completing these tasks are not being measured and have no bearing on the study. The investigator is only looking at how you go about approaching such operations within the software. This exercise will be audio and video recorded for later analysis by the investigator.

Discomforts and Risks: There are no known discomforts or risks associated with this study.
Benefits: By completing the research exercises you should become more aware of your own knowledge about recording systems. The Pro Tools activity may also help focus the way you approach that particular system. By participating, you are also helping make possible the improvement of audio education, an area that has suffered from a lack of targeted research. The results of this study can help audio educators understand better how to help students understand and use recording systems, and even inform the audio community in general in understanding implications for how new recording systems are designed.

Duration/Time: It should take approximately 30 minutes to complete the card sort exercise and about 90 minutes for the Pro Tools exercise. These will be completed in two separate sessions scheduled at your convenience.

Statement of Confidentiality: Only the investigator, dissertation committee, and advisor will know your name; all identities will be coded for anonymity. All data and records for this study will be maintained by the investigator and kept locked in the investigator’s personal home office. Only the principle investigator, dissertation committee, and advisor will have access to this information. In the event of a publication or presentation resulting from the research, no personally identifiable information will be shared. The Office for Research Protections and the Social Science Institutional Review Board may review records related to this project.

Right to Ask Questions: You can ask questions about this research. Contact Barry R. Hill at 867-6285 with questions. If you have questions about your rights as a research participant, contact The Pennsylvania State University’s Office for Research Protections at (814) 865-1775.

Voluntary Participation: Your decision to be in this research is voluntary. You can stop at any time. You do not have to answer any questions you do not want to answer.

Audio and Video Recording: One of the exercises requires the investigator to video record the computer screen as well as audio record your voice. These recordings will be kept locked in the investigator’s home office; only the principle investigator, dissertation committee, and advisor will have access to them. They will be destroyed within five years from the date of your participation both through physical destruction of any recorded media as well as secure erasing of any computer drives (by reformatting the hard drive).

    I agree to these terms for audio and video recording of my participation.

You must be 18 years of age or older to consent to take part in this research study. If you agree to take part in this research study and the information outlined above, please sign your name and indicate the date below.

You will be given a copy of this signed and dated consent form for your records.

Participant Signature  Date

Person Obtaining Consent  Date
## APPENDIX B. CARD SORT SIMILARITY DATA MATRIX

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
<th>C10</th>
<th>C11</th>
<th>C12</th>
<th>C13</th>
<th>C14</th>
<th>C15</th>
<th>C16</th>
<th>C17</th>
<th>C18</th>
<th>C19</th>
<th>C20</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C6</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C7</td>
<td>C8</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C9</td>
<td>C10</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C11</td>
<td>C12</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C13</td>
<td>C14</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C15</td>
<td>C16</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C17</td>
<td>C18</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C19</td>
<td>C20</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX C. THINK-ALOUD EXERCISE INSTRUCTIONS

Practice

Complete this task using the Pro Tools file currently open. Remember to keep talking aloud and describing what you’re doing and thinking. Take your time.

Apply compression to the snare. Do not worry with actual compression setting; just make sure it is working.

1 of 5

Complete this task using the Pro Tools file currently open. Remember to keep talking aloud and describing what you’re doing and thinking. Take your time.

Apply a stereo compressor to the entire mix. Again, do not worry with settings; just make sure it works.

2 of 5

Complete this task using the Pro Tools file currently open. Remember to keep talking aloud and describing what you’re doing and thinking. Take your time.

Apply one common reverberation effect to the following tracks: Kick, snare, overhead left, overhead right. Do not apply individually on each channel.
Complete this task using the Pro Tools file currently open. Remember to keep talking aloud and describing what you’re doing and thinking. Take your time.

Set up a single cue mix for all tracks and route to an output bus connected to the studio’s headphone system. Provide a master cue mix level control.

Apply a single stereo compressor to tracks 1-3.

Apply a single stereo compressor to tracks 1-3, but leave the original, unaffected tracks blended at the mix bus so we hear them as well.
APPENDIX D. INVESTIGATOR THINK-ALOUD CHECKLIST

Participant: __________________________
Date: __________________________________

<table>
<thead>
<tr>
<th>Open ProTools Think-aloud file</th>
<th>Give them the exercise (#1)</th>
<th>When ready, start movie capture</th>
<th>When done, stop movie. Save movie.</th>
<th>Save ProTools file (BH1)</th>
<th>Close Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open ProTools Think-aloud file</td>
<td>Give them the exercise (#2)</td>
<td>When ready, start movie capture</td>
<td>When done, stop movie. Save movie.</td>
<td>Save ProTools file (BH2)</td>
<td>Close Session</td>
</tr>
<tr>
<td>Open ProTools Think-aloud file</td>
<td>Give them the exercise (#3)</td>
<td>When ready, start movie capture</td>
<td>When done, stop movie. Save movie.</td>
<td>Save ProTools file (BH3)</td>
<td>Close Session</td>
</tr>
<tr>
<td>Open ProTools Think-aloud file</td>
<td>Give them the exercise (#4)</td>
<td>When ready, start movie capture</td>
<td>When done, stop movie. Save movie.</td>
<td>Save ProTools file (BH4)</td>
<td>Close Session</td>
</tr>
<tr>
<td>Open ProTools Think-aloud file</td>
<td>Give them the exercise (#5)</td>
<td>When ready, start movie capture</td>
<td>When done, stop movie. Save movie.</td>
<td>Save ProTools file (BH5)</td>
<td>Close Session</td>
</tr>
</tbody>
</table>
## APPENDIX E. CARD SORT CONCEPT NAMES AND BRIEF DEFINITIONS

<table>
<thead>
<tr>
<th>Card code #</th>
<th>Card name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>PT insert</td>
<td>Pro Tools insert point. Used for adding processors such as effects or a compressor to a single track.</td>
</tr>
<tr>
<td>C2</td>
<td>Aux return</td>
<td>Extra channel/track used to bring effects signals (reverb) into the mix.</td>
</tr>
<tr>
<td>C3</td>
<td>Pre-fader</td>
<td>Used in conjunction with an aux send to send a copy of a signal to the cue mix system (headphones in the studio).</td>
</tr>
<tr>
<td>C4</td>
<td>Aux send</td>
<td>Auxiliary output from a channel/track used to send a copy of that signal to an effects device or cue mix.</td>
</tr>
<tr>
<td>C5</td>
<td>PB insert</td>
<td>Patchbay insert point on an analog console. Used to connect a channel-specific processor (compressor) into a channel’s signal.</td>
</tr>
<tr>
<td>C6</td>
<td>PT bus</td>
<td>Pro Tools routing matrix. Routes that particular channel/track signal either to an internal bus (for subgrouping, processing, etc) or to an external hardware output for monitoring.</td>
</tr>
<tr>
<td>C7</td>
<td>Sub-group</td>
<td>Combining two or more tracks or audio signals into a single signal for processing or recording.</td>
</tr>
<tr>
<td>C8</td>
<td>PT pre-fade send</td>
<td>Function in Pro Tools that mimics the Pre-fader function defined above.</td>
</tr>
<tr>
<td>C9</td>
<td>PT aux return</td>
<td>Pro Tools uses “Aux Tracks” for several purposes, one of which is to mimic the analog Aux Return concept used for adding reverb or effects to a group of tracks.</td>
</tr>
<tr>
<td>C10</td>
<td>Channel path</td>
<td>Primary signal path for recording. Opposite concept is Monitor Path used for monitoring recorded signals.</td>
</tr>
<tr>
<td>C11</td>
<td>Cue mix</td>
<td>A submix of parts, grouped via Aux send bus, sent to the studio’s headphone system for musicians.</td>
</tr>
<tr>
<td>C12</td>
<td>Compressor</td>
<td>A signal processing device that controls dynamic levels. Typically accessed via insert points on a single channel, though that channel might have a combination of sub-grouped signals.</td>
</tr>
<tr>
<td>C13</td>
<td>Effects</td>
<td>Adding reverberation, delay, or other special effects to the musical parts, typically via aux send and return.</td>
</tr>
<tr>
<td>C14</td>
<td>Monitor mix</td>
<td>Combination of all parts coming through Monitor Path used for the engineer to hear everything.</td>
</tr>
<tr>
<td>C15</td>
<td>Assignment matrix</td>
<td>Routes a single channel’s signal to a particular output bus, typically a specific track for recording.</td>
</tr>
<tr>
<td>C16</td>
<td>Bus</td>
<td>An output used for routing a channel signal to be combined with other signals, such as for recording or final mixing.</td>
</tr>
<tr>
<td>C17</td>
<td>Post-fader</td>
<td>Function affiliated with an aux send when used for adding effects such as reverb.</td>
</tr>
<tr>
<td>C18</td>
<td>Monitor path</td>
<td>Recorded signals returning through console channels for the purpose of listening. Opposite of Channel path signals which are being recorded.</td>
</tr>
<tr>
<td>C19</td>
<td>PT post-fade send</td>
<td>Function in Pro Tools that mimics the Post-fader function defined above.</td>
</tr>
<tr>
<td>C20</td>
<td>HW assign</td>
<td>An assignment matrix (defined above) located on an analog recording console.</td>
</tr>
</tbody>
</table>
APPENDIX F. SAMPLE TRANSCRIPTION

Expert 1, Task 3. Original transcription.

K, I think what the problem is asking me to do is to, um, not individually insert the same compressor, oh, excuse me the same reverb effect on these four channels, but to um, route it maybe through an aux send and return, so, I would need to first find, not a plugin, but a aux send...(looking through send/insert routing options on audio track)...um...so I’m fishing around for an aux send at this point, and trying to...I’m thinking that...A, interface A-1 would be what I’m looking for, this might or might not be what I’m looking for at this point, um...okay, pre-fader/post-fader, so I’m assuming that’s mute, yes, so...oh okay so this is just a doubling (referring to the pop-up aux send window showing same fader as the aux fader on track). Um...what’s this, Send, okay, so I’m assuming that’s send A, pre or post fader, (prompt to speak louder) uh, sorry, I’ll talk louder, and so, I want to uh, send this information, um, I would probably go post-fader, I’m assuming that black is post fader, blue is pre-fader, pre-fader being in, so uh, I want to send this to...oops, okay...Oh, okay, um, Send A...okay, and then that...I would need to plugin the reverb somehow and return it on a new, uh, channel, cause we’re combing four channels into one reverb, so um, at this, just to get a little ahead of myself a little bit, I would bring in a stereo, not an audio track, but an...I think we would call it an aux input (creating new track), and ..create one stereo, so let’s see if, yes, so this I would tell it that, um, ...say no input..so I want the input to be, oh, let’s see...kay, I think that is what I want (selects A1 like he set on the track send—it’s interface, not bus, but the idea is there)...no way to pan the aux...kay, um, okay, I’m looking now, oh!, okay, so, I’m finding that I could set
this up in the stereo realm, so instead of A1, I want to make it A1-2, and so that’s kick
and snare, and I’m also supposed to put it on overheads, so I go to overheads here, and
put that into there (selecting same A1-2 output for OH tracks), there like that,
and…um…probably end up panning the overheads hard left and hard right, and the kick
and the snare, well, we will try down the middle. Um…alright, and so…at this point uh,
they would, I believe, come into aux one and aux one I would insert a…what’s a multi-
mono plug-in? Multichannel plug-in, I choose reverb, and set that, I’m not going to worry
about the settings cause I can’t play back, let’s choose a hall, that’s fine, and then this
then would need to be routed to the master…output…(looking at output options on aux
track) which I don’t know whether it is or isn’t. I’m still trying to figure out these
interfaces, and…I’m assuming that’s is…that would be the send, that would be the send
(scanning all A1-2s trying to find it), so, …I would need to route this aux one, which uh,
which ends up being the reverb to the output. How I would do that I’m not
sure…um…(scanning output options on aux track). [Actually this could] go on forever,
um…Well, what I could do is bring up the master fader and take a look in there. I would
go to new tracks, bring up a master fader, that would be in stereo, create one of them. Ok,
so there we have our master master, and…I’m assuming that automatically goes out, this
says A1-A2, I’m assuming that’s the output of the master fader, so…I’ll go with A1-A2
here (on aux track), that’s the output of that, and…my guess is that when we would have
playback, we would get some sort of reverb happening, um, I just adjusted my sends so
they would be going to the aux right here, inserted the uh, reverb there, and I’m assuming
that the aux will now go out through the master fader and out to the analog world
somewhere…so done.
K, I think what the problem is asking me to do is to, um, not individually insert the same compressor, oh, excuse me the same reverb effect on these four channels, but to um, route it maybe through an aux send and return,

**STRUCTURED MODEL CONCEPT / ANALYTICAL-PROCEDURAL**

so, I would need to first find, not a plugin, but a aux send…

**CORRECT PROCEDURE**

(looking through send/insert routing options on audio track)…um…so I’m fishing around for an aux send at this point, and trying to…

**TARGETED SEARCH**

I’m thinking that…A, interface A-1 would be what I’m looking for, this might or might not be what I’m looking for at this point,

**ANALYTICAL-SOFTWARE**

um…okay, pre-fader/post-fader, so I’m assuming that’s mute, yes,

**ANALYTICAL-SOFTWARE / STRUCTURED MODEL CONCEPT**

so…oh okay so this is just a doubling (referring to the pop-up aux send window showing same fader as the aux fader on track). Um…what’s this, Send, okay, so I’m assuming that’s send A, pre or post fader, 

**ANALYTICAL-SOFTWARE**

(prompt to speak louder) uh, sorry, I’ll talk louder,

**COMMENT**

and so, I want to uh, send this information, um, I would probably go post-fader,
ANALYTICAL-PROCEDURAL
I’m assuming that black is post fader, blue is pre-fader, pre-fader being in,

ANALYTICAL-SOFTWARE
so uh, I want to send this to..oops, okay…Oh, okay, um, Send A…okay, and then that…

CORRECT PROCEDURE / ANALYTICAL-SOFTWARE
I would need to plugin the reverb somehow and return it on a new, uh, channel, cause
we’re combing four channels into one reverb,

ANALYTICAL-PROCEDURAL / STRUCTURED MODEL CONCEPT
so um, at this, just to get a little ahead of myself a little bit, I would bring in a stereo, not
an audio track, but an…I think we would call it an aux input (*creating new track*), and
..create one stereo,

CORRECT PROCEDURE / ANALYTICAL-SOFTWARE
so let’s see if, yes, so this I would tell it that, um, …say no input..so I want the input to
be, oh, let’s see…kay, I think that is what I want (*selects A1 like he set on the track
send—it’s interface, not bus, but the idea is there*)….no way to pan the aux…kay, um,

ANALYTICAL-SOFTWARE
okay, I’m looking now, oh!, okay,

TARGETED SEARCH / ANALYTICAL-SOFTWARE
so, I’m finding that I could set this up in the stereo realm, so instead of A1, I want to
make it A1-2, and so that’s kick and snare, and I’m also supposed to put it on overheads,
so I go to overheads here, and put that into there (*selecting same A1-2 output for OH
tracks*), there like that, and…

INCORRECT PROCEDURE-CORRECT CONCEPT
um..probably end up panning the overheads hard left and hard right, and the kick and the
snare, well, we will try down the middle. Um…alright, and so…

CORRECT PROCEDURE

at this point uh, they would, I believe, come into aux one

ANALYTICAL-PROCEDURAL / ANALYTICAL-SOFTWARE

and aux one I would insert a…what’s a multi-mono plug-in?

CORRECT PROCEDURE / UNEXPECTED / ANALYTICAL-SOFTWARE

Multichannel plug-in, I choose reverb, and set that, I’m not going to worry about the
settings cause I can’t play back, let’s choose a hall, that’s fine,

CORRECT PROCEDURE

and then this then would need to be routed to the master…output…

STRUCTURED MODEL CONCEPT

(looking at output options on aux track)

TARGETED SEARCH

which I don’t know whether it is or isn’t. I’m still trying to figure out these interfaces,
and…I’m assuming that’s is…that would be the send, that would be the send

ANALYTICAL-SOFTWARE

(scanning all A1-2s trying to find it),

TARGETED SEARCH

so, …I would need to route this aux one, which uh, which ends up being the reverb to the
output.

STRUCTURED MODEL CONCEPT
How I would do that I’m not sure…um…(scanning output options on aux track).

[Actually this could] go on forever, um…

   ANALYTICAL-PROCEDURAL / TARGETED SEARCH / ANALYTICAL-SOFTWARE

   Well, what I could do is bring up the master fader and take a look in there.

   PROCEDURAL STRATEGY CHANGE / ANALYTICAL-SOFTWARE

   I would go to new tracks, bring up a master fader, that would be in stereo, create one of them. Ok, so there we have our master, and…

   CORRECT PROCEDURE

   I’m assuming that automatically goes out, this says A1-A2, I’m assuming that’s the output of the master fader, so…

   ANALYTICAL-SOFTWARE

   I’ll go with A1-A2 here (on aux track), that’s the output of that, and…

   CORRECT PROCEDURE

   my guess is that when we would have playback, we would get some sort of reverb happening,

   ANALYTICAL-SOFTWARE

   um, I just adjusted my sends so they would be going to the aux right here, inserted the uh, reverb there,

   CORRECT PROCEDURE

   and I’m assuming that the aux will now go out through the master fader and out to the analog world somewhere…so done.
ANALYTICAL-SOFTWARE / STRUCTURED MODEL CONCEPT / RECALL

EXPERIENCE

INCORRECT RESULT-CORRECT CONCEPT
APPENDIX G. CODE LIST WITH DEFINITIONS

PROCESSING

• SMC -- STRUCTURED MODEL CONCEPT – describing/explaining concept of current procedure—knows the model principles that govern the procedures being performed. Although may sometimes be procedural, indicates a somewhat higher-level awareness of what conceptually is happening in a given situation.

• PMC -- PARTIAL MODEL CONCEPT – Model is not well remembered, fully understood, and/or misapplied, but portions are present.

• IMC -- INADEQUATE MODEL CONCEPT – Existing conceptual approach is insufficient to allow understanding and completion of the task.

• A-Conc -- ANALYTICAL-CONCEPTUAL– Figuring out the big picture, the model is not solid to inform the procedures.

• A-Proc -- ANALYTICAL-PROCEDURAL– Figuring out how to accomplish the task in Pro Tools. May or may not understand the relevant model principles.

• A-Soft -- ANALYTICAL-SOFTWARE– Attempting to figure out various software interface functions and operations, what they mean, what they are used for, how the software is routing signals internally, etc. Applies when the user is unfamiliar with or unsure of the software and especially when it is not intuitive.

• CSC -- CONCEPTUAL STRATEGY CHANGE – Change of approach from a conceptual perspective, which guides relevant procedures.
• PSC -- PROCEDURAL STRATEGY CHANGE – Change of approach from a procedural perspective. Guiding model concept remains.

• INSTRUCTIONS – not used for analysis

ACTION

• CP -- CORRECT PROCEDURE – Completion of a correct action or sequential series of actions in Pro Tools.

• IP -- INCORRECT PROCEDURE – Incorrect action or series of actions in Pro Tools.

• IP-CC -- INCORRECT PROCEDURE-CORRECT CONCEPT -- Correct model and approach, but did not understand the software sufficiently, resulting in an incorrect action or series of actions.

• TS -- TARGETED SEARCH – Looking for a desired function in the software in targeted, likely places.

• RS -- RANDOM SEARCH – Does not know where to find desired function. Resorts to searching anywhere in the interface.

• CR -- CORRECT RESULT – Task is completed correctly.

• IR -- INCORRECT RESULT – Task is not completed correctly. Any minor error results in an incorrect result.

• IR-CC -- INCORRECT RESULT-CORRECT CONCEPT – Task is not completed correctly, but the fault is due to issues understanding the software. Subject applies correct model and procedures from a generic model perspective.
INTEGRATION/TRANSFER

- **RE** -- **RECALL EXPERIENCE** – Additional information applied to the task beyond what is required to complete the setup based on prior experience using Pro Tools or other recording systems.

- **TC** -- **TASK COMPARISON** – Comparison of current task concept and/or procedures with another task in the study exercise.

COGNITIVE STATE

- **UE** -- **UNEXPECTED** – Software presents an unexpected function, result, etc in a situation where the subject was confident in their approach.

- **UC** -- **UNCERTAINTY** -- Subject unsure of themselves, begin doubting their procedures, their model, the software, etc.

- **Conf** -- **CONFIDENT** – Subject is sure of their approach and ability to complete the procedure/task successfully.
## APPENDIX H. SAMPLE EVENT LISTS

### Expert 1 Event List

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>CP / SMC</td>
<td>SMC / RE</td>
<td>RE</td>
<td>TC</td>
</tr>
<tr>
<td>SMC</td>
<td>RE</td>
<td>SMC / RE</td>
<td>RE / SMC</td>
<td>SMC / RE</td>
</tr>
<tr>
<td>CP</td>
<td>CP / SMC</td>
<td>CP</td>
<td>SMC</td>
<td>CP</td>
</tr>
<tr>
<td>Con / CR</td>
<td>CP / RE</td>
<td>TC</td>
<td>CP</td>
<td>Con / CR</td>
</tr>
<tr>
<td></td>
<td>Con / CR</td>
<td>CP / SMC</td>
<td>RE</td>
<td>Con / CR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RE / SMC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Con / CR</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Expert 2 Event List

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>RE</td>
<td>CP</td>
<td>SMC / CP</td>
<td>CP</td>
<td>CP</td>
</tr>
<tr>
<td>CP</td>
<td>RE</td>
<td>SMC</td>
<td>RE</td>
<td>SMC / RE</td>
</tr>
<tr>
<td>Con / CR</td>
<td>Con / CR</td>
<td>SMC / RE</td>
<td>CP</td>
<td>Con / CR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Con / CR</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Con / CR</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Student 1 Event List

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP</td>
<td>CP</td>
<td>CP</td>
<td>CP</td>
<td>CP</td>
</tr>
<tr>
<td>Con / CR</td>
<td>IP-CC / PMC</td>
<td>RE</td>
<td>RE</td>
<td>SMC</td>
</tr>
<tr>
<td></td>
<td>CP</td>
<td>SMC</td>
<td>A Proc</td>
<td>Con / CR</td>
</tr>
<tr>
<td></td>
<td>Con / IR-CC</td>
<td>CP</td>
<td>CP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CP / SMC</td>
<td>CP</td>
<td>CP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IP</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CP</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Con / CR</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A Proc</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PSC / CP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Con / CR</td>
</tr>
</tbody>
</table>
### Expert 3 Event List

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMC / A Proc</td>
<td>SMC / A Proc</td>
<td>A Proc</td>
<td>SMC / A Soft / RE</td>
<td>TC</td>
</tr>
<tr>
<td>TS</td>
<td>CP</td>
<td>UE / A Soft</td>
<td>A Proc / SMC / RS / PSC</td>
<td>SMC / A Proc</td>
</tr>
<tr>
<td>UE / A Soft / IP-CC</td>
<td>TS</td>
<td>TS / A Soft</td>
<td>IP / A Proc / A Soft</td>
<td>CP</td>
</tr>
<tr>
<td>SMC / A Proc</td>
<td>A Soft</td>
<td>A Proc / A Soft</td>
<td>A Proc / PSC</td>
<td>A Soft</td>
</tr>
<tr>
<td>UE / SMC / A Proc / IP</td>
<td>A Soft / SMC</td>
<td>PSC / A Proc / UC</td>
<td>CP</td>
<td>CP / SMC</td>
</tr>
<tr>
<td>PSC</td>
<td>A Soft</td>
<td>CP</td>
<td>A Proc</td>
<td>SMC</td>
</tr>
<tr>
<td>A Proc / SMC</td>
<td>A Soft</td>
<td>IP-CC</td>
<td>A Proc / UC</td>
<td>CP</td>
</tr>
<tr>
<td>TS / A Soft / SMC / RS</td>
<td>CP / A Soft</td>
<td>CP / SMC</td>
<td>CP</td>
<td>SMC / CP</td>
</tr>
<tr>
<td>A Proc / A Soft / IP</td>
<td>A Proc / SMC</td>
<td>A Soft</td>
<td>CR</td>
<td>Con / CR</td>
</tr>
<tr>
<td>SMC / TS</td>
<td>CP / A Soft</td>
<td>A Proc / A Soft / UC</td>
<td>CP</td>
<td>IP-CC / A Soft / UC</td>
</tr>
<tr>
<td>IP-CC / A Soft / UE</td>
<td>A Soft</td>
<td>A Proc</td>
<td>CP</td>
<td>A Soft</td>
</tr>
<tr>
<td>A Soft</td>
<td>TS / A Soft</td>
<td>CP</td>
<td>CP-CC</td>
<td>PSC</td>
</tr>
<tr>
<td>Con</td>
<td>CP</td>
<td>CP-CC</td>
<td>A Proc</td>
<td>A Soft</td>
</tr>
<tr>
<td>CP</td>
<td>CP</td>
<td>CP-CC</td>
<td>A Proc</td>
<td>A Soft</td>
</tr>
<tr>
<td>A Proc</td>
<td>CP / UE / A Soft</td>
<td>A Soft</td>
<td>A Soft</td>
<td>A Proc</td>
</tr>
<tr>
<td>A Proc</td>
<td>CP</td>
<td>CP</td>
<td>A Proc</td>
<td>A Proc</td>
</tr>
<tr>
<td>TS</td>
<td>SMC</td>
<td>A Soft / UC</td>
<td>UE / A Soft</td>
<td>A Proc / A Soft</td>
</tr>
<tr>
<td>UE / A Soft</td>
<td>TS</td>
<td>A Proc / A Soft</td>
<td>TS</td>
<td>A Soft / UC</td>
</tr>
<tr>
<td>IP</td>
<td>A Soft</td>
<td>A Soft / UC</td>
<td>CP</td>
<td>CP</td>
</tr>
<tr>
<td>RS / A Soft</td>
<td>TS</td>
<td>CP</td>
<td>SMC</td>
<td>SMC / CP</td>
</tr>
<tr>
<td>UC</td>
<td>SMC</td>
<td>CR</td>
<td>A Proc / TS / A Soft</td>
<td>SMC / CP</td>
</tr>
<tr>
<td>SMC</td>
<td>A Proc / TS / A Soft</td>
<td>CR</td>
<td>CP</td>
<td>IP-CC / A Soft / UE</td>
</tr>
<tr>
<td>UE</td>
<td>PSC / A Soft</td>
<td>CP</td>
<td>A Soft / IP</td>
<td>A Soft</td>
</tr>
<tr>
<td>RS</td>
<td>CP</td>
<td>RS-CC</td>
<td>A Proc / A Soft / SMC / IP</td>
<td>A Soft</td>
</tr>
<tr>
<td>UC</td>
<td>CP</td>
<td>IR-CC</td>
<td>A Soft / SMC / RE</td>
<td>CP</td>
</tr>
<tr>
<td>A Proc / A Soft / SMC</td>
<td>IR-CC</td>
<td>A Soft / SMC / UC</td>
<td>A Soft / IP-CC</td>
<td>A Soft / A Proc / A 5</td>
</tr>
</tbody>
</table>

### Student 7 Event List

<table>
<thead>
<tr>
<th>Task 1</th>
<th>Task 2</th>
<th>Task 3</th>
<th>Task 4</th>
<th>Task 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP / SMC</td>
<td>IMC</td>
<td>SMC</td>
<td>TC</td>
<td>SMC / A Proc</td>
</tr>
<tr>
<td>Con / CR</td>
<td>RS</td>
<td>A Proc</td>
<td>PMC / A Proc</td>
<td>IMC</td>
</tr>
<tr>
<td>IF</td>
<td>CP</td>
<td>IP</td>
<td>A Proc / RS</td>
<td>SMC / A Proc</td>
</tr>
<tr>
<td>A Soft / IMC</td>
<td>IR</td>
<td>A Proc</td>
<td>A Soft / PSC</td>
<td>A Soft</td>
</tr>
</tbody>
</table>
| IR      | A Soft | PMC / A Proc | CP | IP-
| A Soft | A Soft / PSC | PSC | CP |
| UC      | A Soft | IP-CC | A Soft | IP-CC |
| SMC      | RS / A Proc | A Proc | A Soft | IP-CC |
| UC       | A Soft | CP | PSC | CP |
| A Soft   | A Soft / IP-CC | CP | RS / A Soft / A Proc / TS | A Soft |
| IR-CC    | PMC | C / A Proc / A 5 | FMC / UC | IR |
| RS / A Soft / A Proc | IR | IR-CC | A Soft / IP-CC |
Vita

Barry R. Hill

Education

2006  Doctor of Education in Instructional Systems  
The Pennsylvania State University, State College, PA

1996  Master of Music in Music Technology  
Studies in Interactive Media, ITP Program  
New York University, New York, NY

1989  Bachelor of Science: Music with Recording Arts  
University of North Carolina—Asheville, Asheville, NC

Professional Experience

1993-  
current  
Associate Professor of Music  
Director Music Recording Technology Program  
Lebanon Valley College, Annville, Pennsylvania

1990-93  
Director Music Engineering & Technology Program  
Elizabeth City State University, Elizabeth City, NC

1989  
Horizon Music, Asheville, NC  
Recording Engineer

1987-  
current  
Independent recording engineer and consultant

Professional Organizations

National Academy of Recording Arts and Sciences (Grammy)  
Music & Entertainment Industry Educators’ Association  
Audio Engineering Society

Publications


2004  In the Studio. Self-published textbook with CD-ROM.