The Learn, Explore, and Practice (LEAP) Intelligent Tutoring System: a demonstration project incorporating instructional design theory in a practical tutor.

Franklyn N. Linton
University of Massachusetts Amherst

Follow this and additional works at: https://scholarworks.umass.edu/dissertations_1

Recommended Citation
https://scholarworks.umass.edu/dissertations_1/5200

This Open Access Dissertation is brought to you for free and open access by ScholarWorks@UMass Amherst. It has been accepted for inclusion in Doctoral Dissertations 1896 - February 2014 by an authorized administrator of ScholarWorks@UMass Amherst. For more information, please contact scholarworks@library.umass.edu.
THE LEARN, EXPLORE, AND PRACTICE (LEAP)
INTELLIGENT TUTORING SYSTEM:
A DEMONSTRATION PROJECT
INCORPORATING INSTRUCTIONAL DESIGN THEORY
IN A PRACTICAL TUTOR

A Dissertation Presented

by

FRANKLYN N. LINTON, JR.

Approved as to style and content by:

Klaus Schultz, Chair
Beverly Woolf, Member
George Forman, Member

Bailey W. Jackson, Dean
School of Education
DEDICATION

To learners:

May intelligent tutoring systems optimize portions of your path.
ACKNOWLEDGMENTS

U S WEST Advanced Technologies had the foresight to fund this intelligent tutoring project, the insight to hire an instructional designer, and granted me the prerequisite of using all data and information related to the LEAP tutor in this dissertation. Scott Wolff laid the groundwork for intelligent tutoring at U S WEST; he conceived the project, tirelessly promoted it to the eventual stakeholders, and ultimately obtained the funding. Larry Lefkowitz, of Bellcore, and the WITS tutor provided an initial impetus to the project. Charles Bloom ably led the LEAP team to its success. Brigham Bell programmed virtually every line of code. Each brought an interest in intelligent tutors to the project and the resulting tutor was the product of a cooperative effort by the whole team. Edwin Norton, Rob Cuthbertson, and Mark Haines each played the role of hardware or unix guru at some time during the project.

The Communications Workers of America (CWA) represented on the project by Cathy DeSmet-Harrington provided several dozen domain experts and an equal number of novices over the course of the project. Most of these were promised anonymity and confidentiality, and cannot be named; many were from the Loveland CO office of U S WEST’s Home and Personal Services. Dora Valdez and Shirley Preuitt, supervisors there, offered the site for formative evaluation and were unstinting with their personnel, time, and space. Many, many people throughout U S WEST were extremely generous with their help and support; to name just a few: Kent Runge, Germaine Guttierrez, Pat Harkinson, Pat Tobie, Ruthie Felix, and Jan Biga. Marilyn Burlingame, of Telcom Training Corporation, Vancouver BC, provided a view of the domain from a perspective that evolved into the domain representation.
This dissertation could not have been written without the continuing long-term support and encouragement of Rosalia Chow, Frank Linton Sr., Cecile Linton, and many friends.

This dissertation has been much improved by the advice from and repeated close readings of my committee: Klaus Schultz, Beverly Woolf, and George Forman.

Beverly Woolf's laboratory has provided support, experience, peer discussions, and entertainment. IBM's T. J. Watson Laboratory has also provided some experience. The people of AED Inc. and Brattle Systems, instructional design businesses, have given me experience and income for my graduate work. Discussions with my fellow researchers at MITRE Corporation's Advanced Information Systems Technology group have contributed to my perspective on this work.

The College of Public and Community Service of the University of Massachusetts, Boston, has given me an outlook on learning and teaching that can be found in all my intelligent tutoring systems. Innumerable other students and teachers have given me experience in individualized teaching and learning over the years.
ABSTRACT

THE LEARN, EXPLORE, AND PRACTICE (LEAP)
INTELLIGENT TUTORING SYSTEM:
A DEMONSTRATION PROJECT
INCORPORATING INSTRUCTIONAL DESIGN THEORY
IN A PRACTICAL TUTOR

MAY 1995

FRANKLYN N. LINTON, JR.

B.A., UNIVERSITY OF MASSACHUSETTS BOSTON
Ed.M., HARVARD UNIVERSITY
Ed.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Klaus Schultz

Intelligent Tutoring Systems (ITS) can provide individualized instruction in problem-solving skills, a kind of instruction that until recently only humans could perform. While ITS have been an active area of research for nearly twenty-five years and researchers have convincingly demonstrated that ITS can instruct in various ways, few ITS are in actual use and their potential benefit to learners is unrealized.

This research is predicated on the notion that ITS research has three closely related but distinct foci: artificial intelligence research in tutoring, instructional research in tutoring, and research on practical tutoring; and on the notion that investigation and evaluation in the latter two areas has been lacking.

With respect to instructional research in tutoring, this work examines the extent to which conventional instructional design theory can usefully inform the design of intelligent tutors, the means of incorporating instructional methods into an intelligent tutor, and the range of instructional skills necessary in a practical intelligent tutor. It examines how ITSs push instructional design theory in the
area of computational instructional design and presents a new instructional method: Focused Practice. Evaluation of tutoring skills focused on trainees' usage of the tutor and the resulting learning, and on measuring the extent to which the tutor was capable of individualizing instruction.

With respect to research on practical tutoring, this work examines the extent to which it is feasible to simulate a work environment, represent the expertise of a non-formal domain, construct a large knowledge base, build a functional student model, supply a shell and authoring tools, incorporate a variety of instructional skills, instructional activities, and instructional materials into a cohesive tutoring package that integrates well into a training program; and gain support from the variety of stakeholders affected by the tutor. Evaluation of practicality focused on trainees' and instructors' affective responses toward the tutor, their perceptions of usability and instructional value; and on other stakeholders' (instructional designers, managers from research, production and training) perceptions of value.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>v</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>xi</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xiii</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 The Research Question</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Significance of This Work</td>
<td>8</td>
</tr>
<tr>
<td>1.3 Basic Plan of the Paper</td>
<td>12</td>
</tr>
<tr>
<td>2. OVERVIEW OF THE RESEARCH</td>
<td>14</td>
</tr>
<tr>
<td>2.1 LEAP Overview</td>
<td>14</td>
</tr>
<tr>
<td>2.2 Design, Implementation, and Evaluation Objectives</td>
<td>21</td>
</tr>
<tr>
<td>2.3 Design Considerations</td>
<td>23</td>
</tr>
<tr>
<td>2.4 Development and Evaluation</td>
<td>26</td>
</tr>
<tr>
<td>3. LITERATURE REVIEW</td>
<td>32</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>32</td>
</tr>
<tr>
<td>3.2 Representation</td>
<td>34</td>
</tr>
<tr>
<td>3.3 Instructional Expertise</td>
<td>38</td>
</tr>
<tr>
<td>3.4 Practical Intelligent Tutoring Systems</td>
<td>43</td>
</tr>
<tr>
<td>3.5 Evaluation</td>
<td>48</td>
</tr>
<tr>
<td>3.6 Summary</td>
<td>52</td>
</tr>
<tr>
<td>4. DESIGN OF LEAP</td>
<td>54</td>
</tr>
<tr>
<td>4.1 The Training Task</td>
<td>54</td>
</tr>
<tr>
<td>4.2 The Domain Representation</td>
<td>62</td>
</tr>
<tr>
<td>4.3 The Student Model</td>
<td>85</td>
</tr>
<tr>
<td>4.4 How LEAP Makes Tutoring Decisions</td>
<td>95</td>
</tr>
<tr>
<td>4.5 Authoring in LEAP</td>
<td>146</td>
</tr>
<tr>
<td>4.6 Design: Summary</td>
<td>160</td>
</tr>
</tbody>
</table>
5. EVALUATION ............................................................................................................... 162
   5.1 Methodology ........................................................................................................ 163
   5.2 Initial Conditions .............................................................................................. 168
   5.3 Affective and Usability Results ..................................................................... 175
   5.4 Instructional Value .......................................................................................... 192
   5.5 Command Usage ............................................................................................ 233
   5.6 Achievement Versus Self- and LEAP-directed Study .................................. 262
   5.7 Individualization of Instruction .................................................................. 274
   5.8 Instructors’ Reflections ................................................................................ 289
   5.9 Summary of Evaluation Procedures ............................................................. 291

6. CONCLUSIONS and RECOMMENDATIONS ...................................................... 294
   6.1 Tutor-related Research Areas ...................................................................... 294
   6.2 Design ............................................................................................................... 297
   6.3 Evaluation ........................................................................................................ 312
   6.4 Authoring ......................................................................................................... 314
   6.5 Cost/Benefit Analysis .................................................................................. 318
   6.6 Summary .......................................................................................................... 320

7. RESULTS, CONTRIBUTIONS, FURTHER WORK ............................................... 321
   7.1 Results .............................................................................................................. 321
   7.2 Contributions .................................................................................................. 327
   7.3 Further Work ................................................................................................... 345
   7.4 Summary .......................................................................................................... 362

APPENDICES
A. LEAP WALKTHROUGH ....................................................................................... 363
   A.1 Introduction .................................................................................................... 363
   A.2 Top Level ........................................................................................................ 363
   A.3 Study the Guide ............................................................................................. 370
   A.4 Rehearse Conversation ............................................................................... 371
   A.5 Examine Contact Flow ................................................................................ 387
   A.6 Revising LEAP’s Decision-making Strategies ........................................... 394

B. LOGGED DATA: SUMMARY AND SAMPLE ...................................................... 401

BIBLIOGRAPHY ..................................................................................................... 415
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Tutor Development Tasks</td>
</tr>
<tr>
<td>2.2</td>
<td>Tutor Development Timeline</td>
</tr>
<tr>
<td>4.1</td>
<td>Voice Messaging Dialogue</td>
</tr>
<tr>
<td>4.2</td>
<td>Script for Adding Voice Messaging Service (VMS)</td>
</tr>
<tr>
<td>4.3</td>
<td>Topics in Conversations: Add VMS</td>
</tr>
<tr>
<td>4.4</td>
<td>Topics in Conversations: Change Ring Cycle</td>
</tr>
<tr>
<td>4.5</td>
<td>Topics in Conversations: Remove Voice Messaging</td>
</tr>
<tr>
<td>4.6</td>
<td>Conversation Exercises in LEAP</td>
</tr>
<tr>
<td>4.7</td>
<td>SA Pair Score Counts</td>
</tr>
<tr>
<td>4.8</td>
<td>Proficiency versus Percent Tried and Average Score</td>
</tr>
<tr>
<td>4.9</td>
<td>VMS Service and Due Date</td>
</tr>
<tr>
<td>4.10</td>
<td>Focused Practice Instructional Activities</td>
</tr>
<tr>
<td>4.11</td>
<td>Instructional Activity versus Topic Status and Student Score</td>
</tr>
<tr>
<td>4.12</td>
<td>Instruction and Instructional Theory in LEAP</td>
</tr>
<tr>
<td>4.13</td>
<td>Instructional Methods Employed in LEAP</td>
</tr>
<tr>
<td>4.14</td>
<td>Summary of LEAP's Tutoring Decisions</td>
</tr>
<tr>
<td>4.15</td>
<td>Explore and Practice: Different Ways of Learning</td>
</tr>
<tr>
<td>4.16</td>
<td>Authoring in LEAP: Steps and Tools</td>
</tr>
<tr>
<td>5.1</td>
<td>Trainee Experience with Voice Messaging</td>
</tr>
<tr>
<td>5.2</td>
<td>Trainee Affective Response Toward Self</td>
</tr>
<tr>
<td>5.3</td>
<td>Trainee Affective Response Toward LEAP</td>
</tr>
<tr>
<td>5.4</td>
<td>Trainee Perception of LEAP's Usability, 1 of 2</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>LEAP, Rehearse Conversation entry screen</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>LEAP, Full Practice mode, process for recording a conversational response</td>
<td>18</td>
</tr>
<tr>
<td>2.3</td>
<td>LEAP, Full Practice mode, SONAR action expected</td>
<td>19</td>
</tr>
<tr>
<td>4.1</td>
<td>Top Level Screen: Topics presented</td>
<td>75</td>
</tr>
<tr>
<td>4.2</td>
<td>The topic Close Contact as defined in the Domain Knowledge Base</td>
<td>81</td>
</tr>
<tr>
<td>4.3</td>
<td>The topic Determine Class of Service as defined in the Domain Knowledge Base</td>
<td>83</td>
</tr>
<tr>
<td>4.4</td>
<td>Top Level Screen: Conversation Table of Contents</td>
<td>91</td>
</tr>
<tr>
<td>4.5</td>
<td>Top Level Screen: Recommend presented</td>
<td>98</td>
</tr>
<tr>
<td>4.6</td>
<td>Top Level Screen: Topics Table of Contents</td>
<td>101</td>
</tr>
<tr>
<td>4.7</td>
<td>Study the Guide: opening screen</td>
<td>104</td>
</tr>
<tr>
<td>4.8</td>
<td>Study the Guide: typical page</td>
<td>105</td>
</tr>
<tr>
<td>4.9</td>
<td>Examine Contact Flow: Observe</td>
<td>108</td>
</tr>
<tr>
<td>4.10</td>
<td>Examine Contact Flow: Full Practice</td>
<td>110</td>
</tr>
<tr>
<td>4.11</td>
<td>Examine Contact Flow: feedback screen</td>
<td>111</td>
</tr>
<tr>
<td>4.12</td>
<td>Rehearse Conversation: entry screen</td>
<td>116</td>
</tr>
<tr>
<td>4.13</td>
<td>Rehearse Conversation: response recording process</td>
<td>118</td>
</tr>
<tr>
<td>4.14</td>
<td>Rehearse Conversation with History window open</td>
<td>119</td>
</tr>
<tr>
<td>4.15</td>
<td>SONAR-related level 1 hints</td>
<td>123</td>
</tr>
<tr>
<td>4.16</td>
<td>SONAR-related level 2 hints</td>
<td>123</td>
</tr>
<tr>
<td>4.17</td>
<td>Conversation-related level 1 hint</td>
<td>124</td>
</tr>
<tr>
<td>4.18</td>
<td>Conversation-related level 2 hint</td>
<td>125</td>
</tr>
</tbody>
</table>
4.19 Rehearse Conversation: feedback screen ........................................... 135
4.20 Authoring in LEAP: defining the Domain Knowledge Base .................. 149
4.21 Study parameters: “Topic Choice” selected ....................................... 152
4.22 Study parameters: “Conversation Choice” selected ............................ 154
4.23 Study parameters: conceptual structure for current topic .................... 155
4.24 Study parameters: “In-Topic Question Probabilities” selected ............... 156
4.25 Study parameters: conceptual structure for other topics ...................... 157
4.26 Study parameters: “Out of Topic Question Probabilities” selected .......... 158
5.1 Trainee age, years ........................................................................... 169
5.2 Trainee education, years ................................................................... 169
5.3 Trainee experience with any telephone company, years ....................... 170
5.4 Trainee experience with U S WEST, years ......................................... 170
5.5 Trainee Service Representative experience, years .............................. 171
5.6 Trainee sales skills ........................................................................... 171
5.7 Trainee experience with computers and software .................................. 173
5.8 Example of questionnaire format ....................................................... 176
5.9 Trainees’ preferred medium for practicing customer contacts, 1 of 3 ...... 180
5.10 Trainees’ preferred medium for practicing customer contacts, 2 of 3 ...... 180
5.11 Trainees’ preferred medium for practicing customer contacts, 3 of 3 ...... 181
5.12 Trainee perception of LEAP’s interactivity ........................................ 185
5.13 Trainee perception of LEAP’s Proficiency ratings ............................... 185
5.14 Trainee perception of LEAP’s conversation straightforwardness .......... 189
5.15 Trainee perception of LEAP’s conversation realism ............................ 189
5.16 Trainee perception of LEAP’s dialogue voice quality .......................... 190
5.17 Trainee comfort at hearing own voice .............................................. 190
5.18 Trainee perception of LEAP’s ease of use ........................................191
5.19 Instructions for Instructional Value questionnaire ..............................193
5.20 Trainee confidence in doing a real change order .................................195
5.21 Instructional value of the feature: Recommend .................................196
5.22 Instructional value of the feature: Topic Proficiency ............................197
5.23 Instructional value of the feature: Conversation Challenge .................198
5.24 Instructional value of the method: Guide to Voice Messaging ...............200
5.25 Instructional value of the method: Rehearse Conversations ..................201
5.26 Instructional value of the method: Examine Contact Flow ....................202
5.27 Instructional value of the method: Observe .......................................204
5.28 Instructional value of the method: Focused Practice .........................205
5.29 Instructional value of the method: Full Practice ...............................206
5.30 Is the changing situation presentation confusing? ..............................209
5.31 Is the number of response types confusing? ....................................210
5.32 Instructional value of conversation feedback ....................................211
5.33 Instructional value of SONAR feedback ..........................................212
5.34 Instructional value of conversation hint ..........................................213
5.35 Instructional value of SONAR hint ..................................................214
5.36 Instructional value of the conversation history ...................................216
5.37 Instructional value of the Talk-only and Talk & SONAR views ................217
5.38 Instructional value of commands: Hear Original and Hear Myself ..........218
5.39 Instructional value of the video conversation summary .......................219
5.40 Instructional value of the command: Repeat Conversation ....................220
5.41 Instructional value of doing an entire conversation ..............................222
5.42 Instructional value of doing an abbreviated conversation .....................223
5.43 Instructional value of Examine Contact Flow ........................................ 226
5.44 Instructional value of the command: Back Up ........................................ 227
5.45 Instructional value of the command: Audio On/Off ................................. 228
5.46 Preference for Observe versus Practice .................................................. 229
5.47 Instructional value of the command: Repeat the Contact Flow ............... 230
5.48 Instructional value of hearing same response by different reps ............. 231
5.49 Study time, total ............................................................................... 235
5.50 Conversations tried one or more times ................................................ 235
5.51 Guide, times entered ........................................................................ 236
5.52 Guide, average time in ...................................................................... 236
5.53 Guide, total time in ........................................................................... 237
5.54 Rehearse, times entered ...................................................................... 237
5.55 Rehearse, average time in .................................................................. 238
5.56 Rehearse, total time in ....................................................................... 238
5.57 Explore, times entered ........................................................................ 239
5.58 Explore, average time in ...................................................................... 239
5.59 Explore, total time ............................................................................. 240
5.60 Model, percent tried ........................................................................ 243
5.61 Model, score .................................................................................... 243
5.62 VMS knowledge, percent ................................................................... 244
5.63 VMS knowledge as score versus percent of SA pairs tried .................... 245
5.64 Knowledge per minute of Rehearse time ........................................... 246
5.65 Trainee confidence as a predictor of performance ............................... 247
5.66 Knowledge per total minutes in Guide + Rehearse + Explore ............. 248
5.67 Rehearse, total time versus score ...................................................... 249
A.3 LEAP Top Level screen - Conversations presented .................................. 367
A.4 LEAP Top Level screen - Help presented .................................................. 368
A.5 LEAP Top Level screen - Recommend presented ........................................ 369
A.6 LEAP "Study the Guide" entry screen ....................................................... 370
A.7 LEAP study "page" screen ........................................................................... 371
A.8 LEAP Rehearse Conversation entry screen ................................................ 373
A.9 LEAP Rehearse Conversation response recording process ................................ 376
A.10 LEAP Rehearse Conversation screen with History window open ................ 378
A.11 SONAR-related level 1 hints ................................................................. 379
A.12 SONAR-related level 2 hints ................................................................. 380
A.13 Conversation-related level 1 hint .......................................................... 381
A.14 Conversation-related level 2 hint .......................................................... 382
A.15 Rehearse Conversation top level Help screen .......................................... 383
A.16 Rehearse Conversation "Windows" Help window ..................................... 383
A.17 Rehearse Conversation "Commands" Help window ................................... 384
A.18 Rehearse Conversation "Instruction Styles" Help window .......................... 384
A.19 Rehearse Conversation "SONAR Simulation" Help window ....................... 385
A.20 Rehearse Conversation "History" Help window ......................................... 386
A.21 Rehearse Conversation feedback screen ................................................ 387
A.22 Examine Contact Flow screen #1 ............................................................. 389
A.23 Examine Contact Flow: Observe ............................................................ 390
A.24 Examine Contact Flow: Full Practice ....................................................... 391
A.25 Examine Contact Flow: Help screen ....................................................... 392
A.26 Examine Contact Flow: feedback screen ................................................ 393
A.27 LEAP study parameters - "Topic Choice" selected .................................. 395
A.28 LEAP study parameters - "Conversation Choice" selected .......................... 397
A.29 LEAP study parameters - "Student Modeling" selected .......................... 398
A.30 LEAP study parameters - "In-Topic Question Probabilities" selected .................. 399
A.31 LEAP study parameters - "Out of Topic Question Probabilities" selected .................. 400
CHAPTER 1
INTRODUCTION

1.1 The Research Question

This research examines the extent to which it is possible to build a practical working intelligent tutoring system (ITS) with a broad spectrum of tutoring capabilities.

Few ITS are in actual use and their potential benefit to learners is unrealized. This research attempts to build a practical ITS, tackling the implementation problems as they arise and either 1) solving them or 2) exposing and characterizing them for future work.

Intelligent tutoring systems can provide individualized instruction in problem-solving skills, a kind of instruction that until now has been teachable only by humans. While ITS have been an active area of research for nearly twenty-five years, and researchers have demonstrated many ITS that instruct with few methods, there are few ITS that instruct with many methods (for example, modeling, coaching, scaffolding and fading, and encouraging articulation, reflection, and exploration). It appears that ITS researchers do not select tutoring strategies consciously, after systematically reviewing instructional design theory, but instead simply apply whatever tutoring strategies they happen to be familiar with. Yet, a suite of tutoring skills is a necessary part of a practical tutor.

1.1.1 A Demonstration Project and Its Rationale

This research demonstrates a practical ITS by building an ITS to solve a genuine training problem for a real organization with a substantive training need and a diverse group of trainees. In this case, U S WEST was the organization and providing service representatives (reps) with new product knowledge was the training task. The training task is described in detail in the section of Chapter 4
titled *The Training Task*, and trainee demographics are reported in the section of Chapter 5 titled *Initial Conditions*.

A practical tutor must perform in the world. It must operate within the hardware, software, and network constraints of its users’ environment. It must be desired by, or at least acceptable to, the organizations it impacts upon, namely the trainees, their union, and their management, as well as the trainers, the instructional designers, and their management.

A practical ITS must be an improvement over existing instructional systems. At minimum, trainees must learn from it. In contrast to ITS developed for research, a practical ITS must focus on student acquisition of knowledge. A tutor that trainees learn from must provide a simulated work environment and a model of the expert’s skills (expert domain knowledge).

A review of the literature reveals that the ITS field has focused primarily on expert domain knowledge, paying little attention to tutoring skills. This situation is perhaps a result of the field’s history: ITS research grew out of expert systems research; expert systems were believed to embody the knowledge of human experts. Using this knowledge for training humans to expertise was almost an afterthought. Clancey’s (1986) early work discredited the assumption that any expert system capable of arriving at an expert solution would be a model of expertise suitable for instruction, but the focus on expert domain knowledge has continued (Anderson, 1988).

In short, besides the lack of practical tutors, there is also a lack of tutors with tutoring skills. These tutoring skills should be based on theories of instruction, address a wide variety of learning situations, and be applied when appropriate.
In the course of fielding a practical ITS, then, several kinds of issues must be addressed: technical, organizational, and instructional. These are now described in more detail.

**Technical Issues:** Most research ITS have a stand-alone design; tutors run on a computer that literally 'stands alone'. In contrast, a practical tutor will deliver its tutoring over a network to the computer or terminal on users' desks. A practical tutor may also interact with or monitor the trainee's use of other application software (or a simulation of it). The tutor must itself be accessible by the training and software maintenance departments over the network. A practical tutor must be designed to work within the constraints imposed by the technical realities of the workplace.

**Organizational Issues:** As mentioned, at least two departments within an organization must accept and support an ITS: the trainees' department, and the training department itself: trainers, instructional designers, and managers. Also, in the trainees' organization, training is likely to be viewed as a commodity; as long as it appears as needed and produces the desired results, it will attract little attention. At most, an improved delivery mode (for example, training delivered to the worker's desktop on demand) might provoke a positive response. In the Training Department itself, however, the culture must change and adapt if it is to employ a new technology. Thus the main organizational issue is how the training department must change to deliver training on ITS.

Among other things, this project describes how the training organization and its customer, the department that uses the tutor for learning, accept it. Will the tutor be accepted as an integral part of a training program? Will instructional designers be able to develop instruction for ITS? Can adequate authoring tools be created for them? How do trainers respond when a portion of training is done with an ITS?
Instructional Design vs. Tutoring Design and Tutoring: Instructional design addresses an instructional need. For example, the instructional need may be to create a lisp 101 course to meet the programming skills requirements of subsequent courses for students with certain prerequisite knowledge. The major elements of the design might consist of: lectures presented by an instructor, section meetings led by a Teaching Assistant, readings from a lisp textbook, and exercises on a lisp ITS. Any attempt to design a lisp tutor in isolation from the accompanying course design is doomed to failure (Singley, Anderson, Gevins, and Hoffman, 1989; McKendree, Radlinski, and Atwood, 1992) because not only must the content and problem-solving methods of the tutor coincide precisely with those of the course in which it will be used, but essential instructional activities must be allotted to the different course elements; what one does not do, another must. A practical tutor must be integrated into the larger instructional design. In this project, the tutor is integrated into an existing course.

Building a practical ITS makes clear the differences between conventional instructional design, tutor design (which is not now something instructional designers engage in), and tutoring. Instructional design is designing an overall training package that solves a specified training problem. Tutor design is designing an ITS within the context of the whole training package; tutor design consists of determining which training tasks the ITS should do, and designing an ITS with the capability to do them. (Or, if using a tutoring shell, it means selecting the portions of the training best taught by the ITS, and creating the files the ITS needs for tutoring the materials.) Finally, tutoring is what the ITS does. Tutor design may become a subspecialty of instructional design.

Currently, the distinction between instructional design and instruction, or tutoring, may often be unclear; especially if the instructional designer happens to be the instructor (or tutor) as well. Even if they are not the same person, instruc-
tors often modify (redesign) the materials as they go along. An ITS, however, makes the distinction clear: whatever the ITS does is tutoring; creating the materials and methods for the ITS to do its task is tutor design.

1.1.2 A General Architecture

The architecture of a practical tutor should permit new domain knowledge, or content, to be input with a minimum of recoding. One design strategy to minimize recoding is to build a tutoring shell. In adapting a tutor for a new domain, nearly everything could be rewritten; however the minimum change in a tutor that is being adapted to a new domain is the domain knowledge itself. This tutor has two characteristics built-in to make it a shell: 1) a representation for encoding domain knowledge, and 2) a set of tutoring methods that operate on that representation, independent of the domain-specific knowledge. The tutoring knowledge is built-in (but is modifiable as necessary). Domain knowledge is input for each domain using the representation supplied. Supplying a representation for domain knowledge: 1) limits the kinds of domain knowledge that can be put into the tutor, but 2) enables the construction of built-in tutoring knowledge. Together, these features define the tutoring shell.

The Domain Representation: As desirable as it may be to build a general-purpose tutoring shell, tutoring actions are dependent on the domain representation. (See the Literature Review chapter for more on representation.) That is, tutoring actions are implemented for a particular domain representation, and any ITS shell is limited in tutoring the content areas that can be reasonably instantiated in its domain representation. In this tutor, the domain representation is a transition network of nodes and links, with each node of the network containing a situation-action pair (roughly akin to a stimulus and response), and with the links of the network indicating paths through the domain. This representation is particularly well suited for certain job tasks, such as those of customer service.
representatives, whose tasks consist of conversations made up of short exchanges and simultaneous interactions with a database containing the customer's account information.

The authoring task, that is, the task of creating new instructional content for this ITS, consists of inputting domain knowledge using the representation supplied, and modifying the tutoring methods and strategies as necessary for the characteristics of the new domain knowledge. The tutor must have authoring tools for both tasks.

The Tutoring Knowledge: A tutoring shell must contain a number of tutoring skills supporting a broad range of learning activities. A general-purpose tutor must have a comprehensive and integrated set of tutoring methods, in contrast to a laboratory tutor that can legitimately aim at demonstrating or validating a single tutoring method. A tutor with a number of teaching methods must also have strategies or heuristics for choosing among those methods as appropriate.

A range of built-in tutoring methods and of strategies for selecting among them, together with the ability to modify both the methods and strategies, increases the generality of the tutor by reducing the need for future reprogramming.

This tutor (known as LEAP, for Learn, Explore, And Practice) incorporates a full spectrum of tutoring skills. The tutor recommends learning activities rather than demand that trainees perform in specific ways. The tutor selects a topic for study and selects an appropriate study method, such as practice, explore, or view multi-media presentations. For each topic, the tutor recommends specific exercises, and selects a method for doing the exercise, such as observe, focused practice, or full practice. For each step in an exercise, the tutor determines whether to skim or scaffold, or require the trainee to practice or review.
It seems redundant to say that an Intelligent Tutoring System should be based on theories of tutoring, teaching, or instructional design. Nevertheless, many tutors embody only their designer's intuitions of instruction. The reader may open ITS-92 Proceedings (1992) at random for examples, and see Jones (1988) for a critique of the ITS field on this point.

LEAP draws on many sources of instructional theory and design. The three main theories are: first, *Characteristics of ideal learning environments* (Collins, Brown, and Newman, 1989), second, *Principles of instructional design*, (Gagne and Briggs, 1979), and third, the theory of direct instruction, as modeled in textbooks of procedural knowledge, such as computer programming. These texts are characterized by short presentations, followed by examples and exercises on a particular topic. Many other learning theories and models of instruction (Joyce and Weil, 1986) influence the design indirectly. These are discussed further in the literature review.

Briefly, Collins, Brown, and Newman’s ideal learning environments include the ability to: demonstrate the skill, coach the learner, scaffold and fade as required; and provide means for and encourage articulation, reflection, and exploration. Gagne and Briggs’ principles of instructional design include the ability to: gain the learner’s attention, inform the learner of the objective, stimulate the recall of prior learning, present the stimulus, provide learning guidance, elicit the performance, provide feedback, assess performance, and enhance retention and transfer. Textbooks of procedural knowledge provide examples of how to organize instruction in a manner learners, instructors, and managers are familiar with.

To summarize, the architecture must include in the authoring tools a means of modifying the tutoring methods and strategies, as well as means for adding and deleting instructional content, or domain knowledge.
1.2 Significance of This Work

This tutor, LEAP, is part of a new phase of tutor development. LEAP is among the first tutors to attempt to perform successfully outside AI research laboratories in the field of education and training. LEAP's users are not merely subjects but learners who have a genuine need for the knowledge they acquire from the tutor. In one respect LEAP's success can be judged by the extent to which these learners perspicuously acquire the knowledge they need.

Earlier research has shown that various tutoring skills can be embodied in software; one of LEAP's contributions is to show that an ensemble of tutoring skills can be put together to form a complete package that has the potential to provide significant benefits over existing media, methods, and materials for education and training; another contribution is to demonstrate evaluation methodologies for ITS; finally, LEAP is an attempt to determine whether a practical implementation of tutoring systems is feasible.

This is a demonstration project: the task is to build and evaluate a tutor in the context of a training program, a tutor to do real training for real trainees, specifically customer service representatives. In the course of building and testing a practical tutor, I found out which tutoring strategies were needed and how well they worked. I assessed tutoring effectiveness by field testing (see the Evaluation Procedures chapter for more detail).

The results of this demonstration project are generalizable to the extent that: the tutor addresses typical skills and learning problems, trainees are typical learners, and the learning situation characteristics are widespread.

The architecture of the tutor is generalizable to skills for which its representation is suitable, and to similar theories of instruction. This tutor uses a representation common in ITS, namely transition networks, and uses a number of theories of instruction. It seems reasonable to infer that if these theories of in-
struction are applicable to one representation, then they are, in general, applicable to representations in general.

Besides the research contributions of a demonstration project, there is a practical value. Tutors have been research projects for nearly 25 years, but few tutors are in use. The presumed benefits of individualized tutoring accrue to no one. The presumption of benefits has hardly been tested.

1.2.1 Instructional Expertise

LEAP is one of the first tutors to incorporate a wide variety of instructional knowledge into a single tutor. Breadth of instructional knowledge is of course a requirement if a tutor is to be practical, but more importantly, the very attempt to incorporate concepts from instructional design theory and practice into the design of a tutor is unusual. This attempt to do so brings to light both the potential contributions of instructional design theory to ITS work, and the shortcomings of instructional design theory as applied to ITS development. These will be discussed further in the Literature Review chapter.

The strength of ITS-based instruction is not that ITS replace human tutors, but that ITS guide students through procedural tasks (exercises), a part of studying where they traditionally flounder along unsupervised by any tutor (human or otherwise). Individualized human tutoring is not available because it is too expensive, and all other forms of mechanized or mediated instruction are just not capable of individualized tutoring of complex tasks. By providing guidance at this stage of learning, ITS provide the potential for students to learn more in a given amount of time. The costs of education and training are increasing much faster than inflation, and ITSs are one way of reducing this cost.

1.2.2 Evaluation Results

Measuring student learning has rarely been an evaluation objective of ITS research. Many ITS research projects have never seen a student. The attempt in
LEAP to measure student learning as an effect of tutoring is a significant contribution to the field.

A second major evaluation omission from conventional ITS research is an evaluation of the effectiveness of the tutor's adaptivity to its individual students. In LEAP's evaluation, not only was the degree of adaptivity measured, but the effect of this adaptivity on student learning was evaluated.

One advantage of practical tutoring systems over laboratory tutoring systems is that AI-focused ITS researchers are able to see what portion of their work is valuable and what is not when intelligent systems interact with large numbers of trainees. The presumption (by funders) of benefits to learners and educators can finally begin to be tested. Another advantage is that education-focused ITS researchers begin to see the possibilities of ITS applications and to see what is possible, what is valuable, and what is not, given the state of AI research.

1.2.3 Practical Value

The construction and testing of practical tutors is a new phase in tutor research. Interest in tutors is spreading from the Artificial Intelligence (AI) community to the education and training community. The research question is shifting from "Can software be built that demonstrates any capacity for tutoring?" to "Can software be built that demonstrates the suite of tutoring skills required for tutoring complex tasks to diverse learners?"

Three powerful forces of change currently affecting all organizations are: increasing computer power, increasing communications power, and the convergence of computer and communications technologies. These forces make ITS possible and potentially practical\(^1\) by reducing the cost of tutors while simultaneously increasing the widespread need for them.

---

\(^1\) "Our own experience is that the time we take to develop instructional modules in our system is no different than the figures typically cited for conventional CAI." (Anderson, 1993, page 254)
The site for this dissertation project is a large multinational telecommunications corporation experiencing the simultaneous stresses of deregulation, increased competition, and downsizing. The same stresses currently apply to many of the world's organizations; speaking metaphorically, not only the playing field, but the players and the rules of the game are changing. Thus if the tutor proves useful to this organization, it may be beneficial to similar organizations.

Education is often thought of as a human-to-human process, yet few students can afford the luxury of a personal human tutor. In higher education, large classes are the norm, and individualized attention is rarely available. ITS have the potential of providing high-quality environments for learning.

McKendree (1992) characterizes ITS technology as an extension of computer based training (CBT). As an extension of CBT, development costs should be somewhat less than CBT, learning benefits somewhat greater. Development costs should be lower than CBT because, unlike CBT, detailed specification of each teaching action is unnecessary, teaching actions are specified as generalities. Learning benefits should be greater because the tutor, while providing whole-task practice in context, keeps students working at the edge of their competence, skimming sub-tasks the student knows, and scaffolding (Brown and Palincsar, 1989) sub-tasks the student is not yet able to perform.

Practical ITS are important because they have the potential to reduce the time required to become an expert. Reduced time to expertise means training costs are reduced, while productivity is increased. Reduced time to expertise also means the workforce can adapt to changing tasks faster, resulting in a more flexible workforce.

---

1 On the other hand, the savings may be offset by the requirement for more detailed specification of domain knowledge.
With the success of this project, the ITS field is one step closer to being able to assist human learning outside the research laboratory. The consequence is learning that is faster, of higher quality, and lower cost than heretofore possible.

1.3 Basic Plan of the Paper

The *Overview* chapter presents an overview of the research, including the research objectives, products, processes, and constraints. The *Literature Review* chapter compares ITSs with other forms of instruction, examines the characteristics of a *representation*, the key feature distinguishing ITSs from other forms of instructional media, it then considers the questions of why instructional expertise has for so long been lacking in ITS work, why few practical tutors have been attempted, and why ITS research is not evaluated, at least not from an educator’s perspective.

The *Design* chapter begins with a description of the training task, which is the context in which the research was performed. Next, the design details of the representation, upon which all else rests, are presented. The third section describes a key feature of the dissertation work: the tutoring decisions LEAP makes, the instructional skills and methods incorporated, and the principles of instructional design employed. The final section in the *Design* chapter describes the authoring process, how new instructional materials may be added to LEAP and how LEAP’s instructional strategies may be radically changed without re-programming.

The *Evaluation* chapter begins with a description of the evaluation methodology and initial conditions. The next section describes trainee affective responses and usability results. The third section reports trainee perceptions of the value of each of LEAP’s features in terms of its contribution to their learning. The section after that presents data describing how trainees actually used LEAP, how much they learned, and how usage affected learning. The last section de-
scribes how LEAP individualizes instruction, and instructors' attitudes towards LEAP felt after having it in their classes for a day.

The Conclusions and Recommendations chapter presents a general discussion of what I have learned about building intelligent tutoring systems. First, AI research in the domain of tutoring is different from instructional research in the domain of tutoring, and building practical tutors is different from them both. The remaining sections present a number of design points: first, some general instructional goals for tutor researchers, then merits of tutor versus trainee control, followed by a recommendation for several forms of feedback. Next some recommendations for multi-media and simulations. The final sections of the chapter deal with evaluation, authoring, and cost/benefit analysis.

The Results, Contributions, and Further Work chapter presents several major results of this work, including the attainment of the project objectives; several contributions, including a new instructional method, and a new research paradigm; and suggestions for further work, including several methods for improving the tutoring process.

Readers unfamiliar with LEAP may want to begin by reading the Appendix LEAP Walkthrough. The walkthrough displays a detailed view LEAP from a user’s perspective: the top-level screens, the three tutoring methods, and authoring.
CHAPTER 2
OVERVIEW OF THE RESEARCH

This chapter presents an overview of the completed research. The first section describes LEAP’s instructional functionality and instructional strategies. The second section describes the dissertation objectives. The third section describes some theoretical and design assumptions and real-world constraints upon which the work is based. The fourth section describes the development process and evaluation procedures.

2.1 LEAP Overview

2.1.1 LEAP’s Instructional Functionality

LEAP is a coached practice environment (Lesgold, Eggan, Katz, and Rao, 1992) for training customer service representatives of a large regional telephone company (US WEST). LEAP contains sufficient domain knowledge for about eight hours of training on the topic of voice messaging sales and service. LEAP has been tested with twenty-nine trainees in one-day sessions. This overview is a high-level description of LEAP’s functionality; the LEAP Walk-through section presents a more detailed description of many of LEAP’s screens and features; the section How LEAP Makes Tutoring Decisions presents mechanisms underlying the functionality described here.

LEAP has three study methods: Study the Guide to voice messaging, Explore the voice messaging knowledge base, and Rehearse voice messaging conversations. The first study method, the Guide to voice messaging, is a book-like set of presentations in text, audio and video on the subject of voice messaging. These reference materials on voice messaging are both menu-accessible and accessible from other parts of LEAP by pressing the Related Info button, which opens the Guide to the topic the trainee is currently exploring or rehearsing. The
Guide illustrates the possibilities of multimedia presentations on the delivery platform.

The second study method, Explore the voice messaging knowledge base, allows the trainee to study a selected topic in detail, and at a slightly abstract level, learn how to respond to all the situations that could occur at every point in the topic. In Explore mode, trainees can acquire a comprehensive, integrated view of a topic, outside of any specific conversation.

In the third study method, Rehearse voice messaging conversations, LEAP simulates the trainee’s work environment. Simulated customers call with requests of various sorts and the rep must converse with the customer appropriately while simultaneously retrieving, entering, or updating customer account information in a simulated database.

In LEAP, these customer calls, or conversations, are exercises. Exercises are grouped by Topic: LEAP’s top-level screen has Topic and Exercise Menus, and a Recommend button. The Recommend button presents LEAP’s recommendation for what to do next. A student can use the Recommend button or make her or his own study choices from the menus.

Exercises have two distinct types of activity: conversing with a customer and manipulating the customer account database. Trainees can perform each activity in any of three learning modes: Observe, Full Practice, and Focused Practice.

When a trainee runs an exercise in Observe mode, LEAP models a conversation between a customer and an expert rep. Mouse clicks by the trainee advance the conversation one step at a time. The trainee hears the conversation through the same headset she or he will wear on the job.\(^1\) Besides hearing the

\(^1\) Conversations are pre-recorded. Genuine expert reps recorded the reps speaking parts; LEAP’s research team played the customer roles.
conversation, the trainee sees the actions the expert rep takes in the database containing the customer's account information. The expert rep brings up database screens in synchrony with the conversation, refers to existing data, modifies it as appropriate, and adds new data based on the interaction with the customer.

When the trainee runs an exercise in Full Practice mode, LEAP performs only the customer's side of the conversation. The trainee must practice the rep's role, both the speaking part and the database manipulation part. The trainee records each spoken response for later review,¹ and works with the simulated database exactly as she or he will do with the real one. During the exercise, LEAP presents a situation, then observes and evaluates the action the trainee takes, provides feedback, and updates the student model. Figure 2.1 shows the Rehearse Conversation and the SONAR Simulation windows (SONAR is the database). Figures 2.2 and 2.3 illustrate Full Practice Mode. In general, if the trainee's action is correct, the conversation simply continues. If the action is incorrect, LEAP provides informative feedback² and gives the trainee another chance.

¹ As LEAP cannot interpret trainee's verbal responses, it presents a list of plausible responses from which the trainee selects the one closest to what s/he said.

² For database errors, LEAP describes the expected field and value. For verbal errors, LEAP only indicates the user has erred.
In Focused Practice, LEAP modifies the exercise by doing some of the steps for the trainee. LEAP modifies the exercise so that the trainee practices the material she or he is currently attempting to learn, i.e., material in the current topic, reviews (by observing and occasionally practicing) material she or he has already learned, and observes material that is not-yet-learned and not in the current topic.
LEAP, Full Practice mode, process for recording a conversational response

At the end of a conversation a brief video summarizes the salient points. The trainee can also review the conversation, comparing the responses she or he recorded with those of the expert, and either return to the top-level or repeat the same exercise in any of the three learning modes.
2.1.2 LEAP’s Tutoring Decisions

The LEAP intelligent tutoring system is a job task simulator with both domain and tutoring knowledge. Domain knowledge is organized by topics with declarative presentations and procedural exercises. Tutoring knowledge consists of numerous tutoring methods and a number of strategies for selecting the most appropriate domain knowledge and tutoring method for every state of the student model.
One set of tutoring strategies selects topics to study and recommends one of several tutoring methods for studying them, based on the trainee’s current knowledge and preferences, and on the topics’ prerequisite relationships.

Another set of tutoring strategies recommends one exercise to do next and a method for studying it. LEAP selects an exercise based on the trainee’s current domain knowledge, characteristics of the exercise, and factors determined by the author of the instruction. LEAP recommends one of three methods for studying an exercise, each emphasizing a different cognitive skill.

A third set of tutoring strategies determines within-exercise actions. Because exercises emphasize whole-task practice, within any given exercise a variety of actions may be required. LEAP keeps trainees practicing at the edge of their competence by selecting topic-related actions for practice, selecting some previously learned actions for review, skimming other previously learned actions and scaffolding unfamiliar unrelated actions.

The Author mode has tools for easily modifying the topic selection strategy, the exercise selection strategy, the action selection strategy, and the student modeling facility. Strategy modifications are likely to be needed whenever the characteristics of the intended users change or a different set of domain knowledge is put into the tutor.

Domain knowledge is stored in the tutor as presentations, and these presentations are accessed by the trainee when she wants to study information about the domain. This information is broken into small chunks corresponding to the topics of instruction. Each chunk is accessed independently. The media are: text, graphics, sound, speech, animation and video. The domain knowledge is authored on Macintosh computers and transferred to UNIX file formats for use. Domain knowledge presentation has not been emphasized in this dissertation. Many multimedia authoring tools exist, and the field is evolving rapidly.
2.2 Design, Implementation, and Evaluation Objectives

In brief, the objectives of this project are to determine the extent to which a complete, practical ITS with a suite of tutoring skills can be designed and implemented; to determine the extent to which the ITS can be used by learners, instructors, and designers; and to determine the extent to which the ITS will provide measurable benefits to learners, instructors, and designers.

2.2.1 Design Objectives

This section lists the design objectives of the tutor. In the construction of the tutor, the existence of each of these components is a result. For example, the existence of a student model and an expert model are results. The incorporation of tutoring methods and strategies are also results. Design details are discussed in the section How LEAP Makes Tutoring Decisions.

- Design a complete practical tutor
  - Simulate customers and database software
  - Model an expert service representative’s behavior
  - Track trainee progress using a student model
  - Make appropriate tutoring suggestions.
  - Embody one or more theories of instruction
- Design a set of authoring tools
  - Tools for inputting new domain knowledge
  - Tools for modifying tutoring skills
- Design a general-purpose tutoring shell

2.2.2 Implementation Objectives

This section lists the implementation objectives of the tutor. The first is to implement the items listed in the previous section. Beyond the ‘simple’ construction of a stand-alone version of a tutor, usable only for demonstration purposes,
is the objective that a large number of practical implementation problems be addressed, such as simulating within the tutor the database that trainees will use on the job, and obtaining agreement from all interested parties that the expert knowledge base is complete and correct. Success on the last objective -- Gain support of potential trainees -- means that people who might use the tutor actually want to (the histories of the GRACE tutor (McKendree, 1992), the WITS tutor (Lefkowitz, personal communication, July, 1993), Singley’s algebra tutor (Singley, Anderson, Gevins, & Hoffman, 1989), and many software projects indicate that this would be no small achievement). Indicators of the attainment of these objectives would be the functioning existence of the tutor in the environment, with discussions of the limitations, and statements by potential trainees indicating the tutor’s desirability.

- Build the items listed in design objectives
- Put an ITS into use in a work environment
- Demonstrate a practical ITS, or
- Identify why a practical ITS is not yet feasible
- Gain support of potential organizational and individual users

2.2.3 Evaluation Objectives:

This section lists evaluation objectives. Some of these objectives, like the first and second, are tested with qualitative and quantitative measures. Others, like the third, are tested with a single subject. The fourth objective is demonstrated by an existence proof. The achievement of the fifth objective can be inferred from the achievement of the first objective.

- Show that trainees learn from the tutor
- Show that trainers elect to use the ITS in their classes
- Demonstrate that instructional designers can use the tools
• Show the ability to apply instructional design theory in ITS construction
• Document the adequacy/inadequacy of tutoring methods and strategies

2.3 Design Considerations

2.3.1 Assumptions

This dissertation is written with several assumptions in mind: Two design assumptions are that tutoring takes place in the context of more complete instruction, perhaps a complete course, and that a cognitively plausible (Anderson, Boyle, Corbett, and Lewis, 1990) representation of domain expertise can be developed. The basic assumption of Artificial Intelligence (of which Intelligent Tutoring is a part) is the Physical Symbol System Hypothesis (Newell and Simon, 1976). Yet another assumption is that existing theories of instructional design (with their implicit assumption of a human teacher in a classroom) will provide adequate guidance for ITS construction. This last assumption is in contrast to the views of, for example, Elsom-Cook (1991), and others and will be discussed further in the Literature Review chapter.

A final assumption is the availability of resources: This work has been carried out in the context of a large organization, one that can afford to build a tutor. The work makes use of a number of resources that have a bearing on the research: Personnel used Macintosh computers as appropriate. Development of the tutor itself took place on UNIX workstations (SUN Sparc series). Multimedia was developed on Macintosh computers and ported to UNIX. Evaluation took place on networked UNIX workstations approximating trainees' actual working conditions.

2.3.2 Limitations

This study is formative and generative; results are illustrative rather than definitive and indicate how a suite of tutoring skills can be successfully combined with a widely used representation to produce a reasonably general-pur-
pose tutor. The physical product is a tutor with a specific representation into which similar domain knowledge may be put. The tutoring knowledge is general-purpose in the sense that it can tutor anything that can be instantiated in the representation.

2.3.3 Delimitations

This dissertation is education-focused ITS research rather than AI-focused ITS research: AI researchers try to make computers intelligent (the consultation program MYCIN, upon which Clancey (1986) based his tutor, only expected its users to provide it with facts that it was unable to acquire on its own). AI-focused ITS researchers' goal is to build ITS with the domain and tutoring skills of human experts. Education-focused ITS researchers try to build ITS from which humans can acquire domain expertise at an optimal rate. AI-focused ITS researchers try to build tutors that are self-contained. Education-focused ITS researchers are content to build ITS where the intelligence of the trainee or of a human instructor also plays a role in the tutoring process.

Issues of deployment, such as hardware and network configurations, are not addressed.

Issues of advanced authoring tools, such as tools for creating a workspace (an environment or a simulation), tools for knowledge engineering, and tools for multimedia authoring are not addressed. The multimedia presentations side of the tutor are not addressed. Budget is not addressed.

Much early ITS research was based on recreating the dialog of human tutors with their students. This tutor is based instead on applying the concepts of instructional design to the learning situation, and providing an optimal context for learning at each moment of time.

Many ITSs focus on building a cognitively plausible, runnable expert model of domain knowledge (Clancey, 1987; Anderson et al., 1990). This tutor fo-
cuses instead on constructing an expert model of tutoring knowledge, which makes intelligent tutoring decisions, decisions that optimize trainees' rate of learning. A cognitively plausible, runnable expert model of domain knowledge is included, but it is not the focus of the research.

2.3.4 Hardware Constraints and Tradeoffs

Hardware design, in the sense of selecting the hardware, or a configuration of hardware, that the tutor will run best on, is not really an option in this situation. The users of the tutoring work on UNIX X-windows terminals, and the tutoring must be delivered on those terminals. These terminals are linked by ethernet LANs and connected to mainframe computers by powerful WANs. Tutor development is constrained by hardware considerations: the tutor must run on the trainees' terminals, and the tutoring content must be deliverable in real time over the LAN and WAN networks. Within this design space there are many possible tradeoffs. The tutor software itself is runnable on any powerful UNIX box, such as a Sun Sparc 10 with two CPUs, one for the simulation and one for the multimedia. As mentioned in the Delimitations section, issues of deployment, such as hardware and network configurations, will not be addressed.

Multimedia especially eats up network resources, and since some media use more resources than others, this raises the issue of media type vs. quality and quantity of learning, an issue that has never been resolved definitively (and is, in my opinion, unresolvable). For example, the network resources required to transmit a few line drawings and some text are negligible compared to those required to transmit a few minutes of video; yet both media may instruct equally well in most circumstances. As mentioned in the Delimitations section, the multimedia presentations side of the tutor will not be addressed.
2.4 Development and Evaluation

2.4.1 Development Process

The ITS development process incorporates the development processes from two related disciplines: instructional design and software engineering, plus healthy doses of creativity, experimentation and iteration for those decisions the aforementioned disciplines are silent about. The procedure for developing LEAP was to build the tutor one stage at a time, trying it out on trainees at each stage -- as much as it could be tried out -- to make sure it was working as planned, revising as necessary to make it function well and tutor effectively. The tasks of developing a tutor are described in Table 2.1, Tutor Development Tasks. The timeline for performing these tasks is presented in Table 2.2, Tutor Development Timeline.
Table 2.1
Tutor Development Tasks

<table>
<thead>
<tr>
<th>Step</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Assemble group</td>
</tr>
<tr>
<td>2.</td>
<td>Select topic, select applicable theories</td>
</tr>
<tr>
<td>3.</td>
<td>Design overall instruction</td>
</tr>
<tr>
<td>4.</td>
<td>Design tutor in context of overall instruction</td>
</tr>
<tr>
<td>5.</td>
<td>Build, test, and demonstrate a functional prototype of tutor including: core software, core domain knowledge, core tutoring strategies, core interface functionality and interactivity. Determine knowledge representations, illustrate media possibilities, perform formative evaluation, refine overall design.</td>
</tr>
<tr>
<td>6.</td>
<td>Build and test a complete version of tutor including: integrate with the simulated database software, the trainee’s telephone headset, and realistic conversations, build overall interface with integrated look and feel, develop all declarative multimedia presentations, develop all procedural exercises. Test software functionality, test tutoring functionality, test interface design. Repeat as necessary.</td>
</tr>
<tr>
<td>7.</td>
<td>Transfer to experimental job-site, solve all hardware, software, and content transition problems. Run evaluation of tutor. Write up results.</td>
</tr>
</tbody>
</table>

Note. Steps 2 through 7 require extensive interaction with subject matter experts.
Table 2.2
Tutor Development Timeline

<table>
<thead>
<tr>
<th>Month</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Project Initiation: Commitment by: research, training, and customer service departments.</td>
</tr>
<tr>
<td>2</td>
<td>Staffing, goal refinement, and topic selection.</td>
</tr>
<tr>
<td>3-5</td>
<td>Front end analysis, instructional design, software prototyping, and formative evaluation.</td>
</tr>
<tr>
<td>6</td>
<td>Demonstration of prototype to all parties.</td>
</tr>
<tr>
<td>7-12</td>
<td>Full scale development and testing: hardware, software, multimedia declarative presentations, interactive tutorial exercises, deployment on network, and formative evaluation.</td>
</tr>
<tr>
<td>13-16</td>
<td>Evaluation and report.</td>
</tr>
</tbody>
</table>

2.4.2 Evaluation Procedures

The main purpose of evaluating the tutor was to measure the extent to which trainees learned from it. Detailed results of field trial evaluation are reported in the Evaluation chapter. Preliminary evaluation had the purpose of evaluating and modifying the characteristics of the tutor such that trainees would eventually learn from it at an optimal rate. Preliminary evaluation took place during the tutor’s development. Once the tutor was sufficiently developed, and an adequate knowledge base was installed, field trials of the tutor began. During field trials, the trainees’ learning experiences were evaluated. Developmental evaluation continued during the field trials, and the results were incorporated into the tutor’s design.
During development, several kinds of expert from outside the program examined the evolving tutor (Flagg's (1990) 'connoisseurs'): experts from the field of ITS, including William Clancey of the Institute for Research on Learning, Alan Lesgold of the Learning Research and Development Center, Beverly Woolf of the University of Massachusetts, Lawrence Lefkowitz of Bellcore, and Bob Radlinski of NYNEX provided comments and feedback on the design of the tutor; subject matter experts reviewed all content materials; and experts from the client organizations, both training and end users, examined the overall tutor¹.

During development, the tutor was tried out on trainees as early as possible and after every major revision in order to acquire information regarding the fidelity of the simulation, the functionality of the software, the usability of the software, the utility of the instruction as designed, and the reaction of the intended audience.

As trainees tried out each new prototype, they were observed in an informal way. During the session, both the use of the tutor by trainees and the required interventions by the developers were noted for use in refining the design.

As the tutor's design stabilized, increasing amounts of domain knowledge were confidently added to it. With substantial amounts of domain knowledge, with the interface simplified and clarified, and with bugs worked out of the code, field trials could begin; not only did developmental testing continue, but also student learning was measured. Ultimately, the goal of evaluating the tutor is to determine how good a means of instruction it is, as well as to determine ways it works well, and ways it needs improvement. ITS researchers and their funders believe that the potential of tutors to contribute to instruction is significant; the

---

¹ Demonstrations of LEAP to interested people took place at least weekly. LEAP was demonstrated to parties ranging from teammate's in-laws to US WEST's board of directors.
field has finally progressed to the point where this potential can begin to be demonstrated.

In the corporate setting in which LEAP was developed, large numbers of trainees are not available for field testing. Although several hundred individuals may be trained per year, they are distributed over 14 large western US states; not only would transportation of the tutor (and the ITS development team) have been prohibitively expensive, but the varying regulatory requirements among individual states would have required the tutor to have a domain knowledge base for each state. Thus practical considerations limited access to trainees to five local training courses averaging six trainees each.

Although the primary purpose of field trials was to test trainee learning, further developmental testing of the tutor to acquire information regarding the fidelity of the simulation, the utility of the instruction, the usability of the software, and the reaction of the intended audience, were profitably continued.

One way to evaluate the overall effectiveness of an ITS is a “horse race” (Winne, 1993, p. 323), that is, to compare it with another method of instruction, such as classroom instruction, one-on-one tutoring with a human coach, self-study, etc. While such testing produces a global measure of the tutor’s effectiveness and is desirable both from a funder’s point of view and from the point of view of the field of intelligent tutoring, it is a pointless test for a particular tutor until a fairly well-refined version has been developed, i.e., the software bugs have been worked out, and the interface features, the domain knowledge, and the set of tutoring skills have been through several iterations and are well-integrated and working as intended. Intended improvements can actually introduce errors or make things worse in other ways, thus the testing cycle can be laborious. This stage of development can be reached only after years of testing; the LEAP tutor did not have the opportunity for such refinement. Instead of a horse
race, LEAP was evaluated with the goal of testing: the functioning of various components, their integration, and the learning that resulted from using LEAP.

Evaluation of Authoring Tools: An important but much smaller evaluation took place as course materials were developed. As the design stabilized, authoring tools were also developed; they were evaluated by having a conventional instructional designer use them to author materials for the tutor's domain knowledge base. Whereas the end users are service representative trainees and the tutor must work for all of them, it is not required that the authoring tools work with all instructional designers. It is necessary, however, to determine the skills instructional designers must have in order to use the tools. Since the result of authoring -- a representation of domain expertise -- is running code, some experience in writing software that runs, whether with a CBT shell, a programming language, or some other format, was found to be helpful.

To summarize the evaluation process: First, the tutor was evaluated as it was developed. Representatives from all interested organizations contributed by reviewing the content, or domain knowledge, the usability, the tutoring interaction, and the design itself. Second, as the design stabilized, and sufficient content was input, the capability of the tutor to perform as an ideal learning environment was evaluated. Trainees used the tutor for relatively long periods of time, about three to six hours, filled out questionnaires and were observed by researchers during that time. Simultaneously, the tutor recorded trainees' actions while using the tutor, their student models, and their spoken responses. Third, as the tutor was used, its 'fit' to the classroom environment was noted. Fourth, the authoring tools were informally evaluated for adequacy.
CHAPTER 3
LITERATURE REVIEW

3.1 Introduction

Intelligent tutoring systems are a new medium of instruction. One feature that distinguishes ITSs, and artificially intelligent software in general, from other types of software is the concept of representation. Another feature that ITSs require is tutoring skill, though surprisingly, most tutoring systems do not have much of it, and there are some serious obstacles to be overcome before they will have enough tutoring skill to be practical. As a result, practical tutors are rare; only a handful of tutors have been constructed for actual use by students, and as a result of that, evaluation of ITS’ tutoring skills is also rare. This chapter begins with a comparison of ITS to other forms of instruction, particularly traditional computer-based instruction. The next section describes the characteristics an ITS’ representation must have to support instructional expertise, and summarizes the extent to which two outstanding tutors (and one design for a tutor) implement instructional expertise. The next section makes the claim that most existing ITS actually lack instructional expertise, considers the counterclaim that instructional design theories in their current form are of limited value to ITS research, and lists a number of theories that are relevant and informative to ITS design. The following section describes the few practical tutors ITS researchers have deployed. The final section of the chapter considers why ITS’ tutoring effectiveness is rarely evaluated as well as other salient evaluation issues.

3.1.1 How ITS Is Different from Other Instruction

The main features that differentiate Intelligent Tutoring Systems from other mediated instruction are their flexibility and adaptability to the individual trainee. An ITS presents topics of instruction as soon as their prerequisites have
been mastered, exits them as soon as their contents have been mastered and re-views them as soon as their contents have been forgotten. Within topics, domain knowledge is embodied in a variety of activities. The ITS selects for the student the most appropriate activity of a set, depending on the nature of the knowledge itself and on the nature and knowledge of the student. Within each activity, the student or the tutor will do each step, with the tutor providing demonstrations, feedback, hints, etc. for the steps the student does.

Although both ITS and computer-based training (CBT) use the computer as their medium, the ITS approach to the task of instruction is based on a completely different design perspective than traditional CBT. CBT is based on Skinner’s theory of Behaviorism and its text-based training approach: Programmed Instruction. CBT typically presents information to trainees one screen at a time, followed by multiple-choice questions about the material presented. If the student answers the questions correctly, the program advances, if not, it presents the information again. There are more advanced versions of CBT; these might, for example, simulate the screens of some software application, say, a word processor, and have the trainee take a specific action. If the trainee takes the correct action the simulation proceeds to the next step, otherwise it gives the trainee the chance to try again, perhaps with more explicit direction. Well designed CBT can be instructive, even entertaining, but it is hardly more flexible or adaptive than a textbook. Nevertheless CBT has two features ITS should aspire to: first, every CBT system embodies at least one theory of instruction, if only Behaviorism, and second, CBT software is practical, it is a commercial success, a certain percentage of the industrial world’s training is done with CBT.

The ability of ITSs to be flexible and adaptive derives from an altogether different approach to representing knowledge from that taken by CBT. In an ITS, domain knowledge is broken into tiny abstract chunks, typically IF ... THEN ...
rules, or *productions*. The ITS can itself apply these rules to exercises and solve them, often in multiple ways. Using this domain skill together with some tutoring skill, which is perhaps encoded in the same way, an ITS can demonstrate solutions or provide hints regarding the next solution step. It can evaluate student’s solution steps by comparing them with the expert rules, and so provide feedback on each solution step the student takes, and on the whole performance. If the ITS has rules mimicking common errors (so-called ‘buggy rules’) it can provide coaching on these specific errors as well. Furthermore, it can keep track of how well the student uses each rule, use this information to predict performance on new exercises, and thus intelligently select exercises, modify exercises, determine when a student knows all the information in a topic, etc. This ability to observe a student in action and respond appropriately as she or he practices some task is ITSs strongest point. Not even the most vigilant human instructor could monitor in step-by-step detail the progress of every student in a class; and in most cases instructors do not even have the opportunity to do so, since students typically practice their skills on exercises assigned as homework, which they do outside the presence of the instructor.

### 3.2 Representation

One of the fundamental concepts of artificial intelligence is *representation*. In order for an ITS to have *any* instructional expertise a suitable representation must be chosen:

A suitable knowledge representation is crucial for encoding the knowledge and the states of reasoning of intelligent agents that can understand natural language, characterize perceptual data, or learn about their world. This is because the representational primitives, together with the system for their combination, effectively limit what such systems can perceive, know, or understand. (Woods, 1987, pp. 44-79)
The characteristics of the representation chosen for LEAP must enable it to teach lessons that involve some skill on the learner's part. While other characterizations of the purpose of an ITS are possible, this skill-focused definition is interesting because it builds on the notion of expertise as the active application of knowledge by the learner.

3.2.1 A Suitable Representation Must Serve Several Purposes

LEAP teaches its trainees the skills of expert service reps. To be capable of attaining this goal, LEAP must 1) be capable of expert service rep performance, and 2) be capable of teaching its trainees that same expertise. To instruct well, the instructional expert needs to make sound instructional decisions, based on the above-mentioned theories of instruction or learning. The instructional expert also needs to access not only the domain expert's problem-solving expertise, but other domain knowledge as well, such as the problems or exercises themselves. Furthermore, the instructional expert needs information about the trainee's current knowledge, and information about the trainee's actions as they are performed.

For example, suppose the trainee is learning to respond to a (simulated) customer's call. The domain expert must be able to perform the task of, say, recognizing the type of call and responding appropriately by keying the right transaction type into the database. The instructional expert must be able to run the domain expert in order to demonstrate the task and concurrently describe it. The instructional expert also needs to access other domain knowledge, such as introductory text or visuals describing the task, and suitable examples for demonstration, practice and testing. The instructional expert also needs to know which parts of the task are already known to the trainee so as not to re-teach them (the trainee may already know how to key in the transaction type), and it needs to be able to monitor the trainee as she tries the task, step by step, in order to provide
learning guidance and feedback. In short, a suitable representation for an ITS must:

- Represent domain expertise
- Represent other domain knowledge useful for instruction
- Represent the trainee's current knowledge
- Represent tutoring expertise that:
  -- Implements a theory or model of instruction
  -- Is capable of running the domain expert
  -- Is capable of monitoring the trainee's learning activities
  -- Has principled access to other domain knowledge

The main representational formalisms found in AI texts are production systems, semantic networks, frames, and logic (Winston, 1984; Cercone and McCalla, 1987; Shadbolt, 1989; Waterman, 1986; and Minsky, 1991). Knowledge can be represented declaratively and procedurally, explicitly and implicitly, as fixed-text and in a machine-interpretable form. The most common representations found in ITS are production systems, semantic networks and frames (Frasson, 1988).

ITS typically work by repeatedly observing the trainee and classifying domain knowledge into two types: 1) known to the trainee, and 2) unknown to the trainee; then selecting, in some ad hoc manner, a chunk of unknown knowledge and presenting the trainee with some fixed-text relating to it.

Surprisingly, the domain representation in most tutors is limited to fixed-text domain knowledge only; the representation is not a representation of domain expertise in a form that enables tutors to perform tasks in the domain. As a result, there are many fundamental tutoring tasks they cannot perform such as: demonstrate skills, provide learning guidance, give feedback, and assess a learn-
er's performance. The representation of teaching expertise in most tutors is also modest at best: they have no sense of curriculum nor any explicit ability to instruct. In sum, their teaching ability is generally limited to presenting fixed-text domain knowledge to trainees.

3.2.2 Examples of Suitable Representations

While most ITS cannot teach, in the educator's sense of the word, the representations used by Anderson (1991, 1990, 1987) and Clancey (1987) in their tutors, and by O'Neil, Slawson and Baker (1991) in their design for a tutor, come closest to meeting all the representational requirements listed above:

Represent domain expertise Anderson and Clancey represent domain expertise with production systems that are specifically designed to meet the needs of learners, as well as to replicate expert performance.

Represent other domain knowledge useful for instruction Both come up short here: domain knowledge is limited to that appearing in the exercises themselves. Clancey assumes the student is already familiar with the knowledge, Anderson places it in a textbook.

Represent the trainee's knowledge Both Anderson and Clancey consider the student model to be a subset of the expert model and draw on the student model to instruct perspicuously, though neither uses it to motivate the student. Anderson also uses buggy rules to represent common, specific student errors.

Represent tutoring expertise that implements a theory or model of instruction Both Anderson and O'Neil base the tutor's actions on a theory; Anderson on his theory of cognition, O'Neil on Merrill's Component Display (Merrill, 1983) theory. Clancey's tutoring actions are ad hoc.

Represent tutoring expertise that is capable of running the domain expert Neither Anderson and Clancey runs the domain expert to demonstrate a solution process to the trainee (but see next item).
Represent tutoring expertise that is capable of monitoring the trainee’s learning activities. Both Anderson and Clancey run the domain expert to solve exercises, and both compare the student’s solution path with the expert’s for instructional purposes.

Represent tutoring expertise that has principled access to other domain knowledge. Neither Anderson nor Clancey intend their ITS to instruct complete lessons; Clancey assumes the trainee already knows most of what MYCIN knows, and Anderson intends that the trainee acquire the information beforehand, from a lecture or a book. Neither tutor is capable of sequencing exercises or topics.

3.3 Instructional Expertise

3.3.1 Problem: ITS Lack Instructional Expertise

An ITS should have a model of instructional expertise. Numerous models of instructional expertise are found in the instructional design and teaching literature (Reigeluth, 1983, 1987; Joyce and Weil, 1986; Simpson, 1980), yet ITS, in spite of their name: Intelligent Tutoring Systems, tend to ignore tutoring expertise, and few instructional design theories are cited in ITS research.

For example, a survey of ITS-88 (Frasson, 1988) reveals that only about a third of the ITS described have a model of instructional expertise. At a recent conference devoted to ITS, Artificial Intelligence in Education (Brna, Ohlsson, Pain, 1993) only two of sixteen sessions (five papers) and one invited speaker had an explicit focus on pedagogy or tutoring strategy. Examination of these presentations reveals that most of these researchers did not, in fact, turn to instructional design theory for guidance or inspiration. Two of these papers had no instructional design references, one paper mentioned one: Magar’s work on instructional objectives; another paper mentioned two: J. S. Brown, and L. Resnick; and a third paper was built around a particular instructional procedure: Conversation...
Rebuilding. In general, however, these ITS researchers cited other ITS work, cognitive science, psychology, and AI. There is little indication of widespread understanding or broad application of instructional design theory in ITS research.

Wenger (1987, pp. 402-415) confirms that ITS researchers tend to design their instructional materials in an arbitrary or ad hoc manner rather than use established instructional design theories and models. Many other observers of the intelligent tutoring field have made similar observations; as exemplified in the following comments by Halff; Tennyson and Park; Lepper, Woolverton, Mumme, and Gurtner; Elsom-Cook, Capell and Dannenberg; and Anderson.

Halff is one of the original sources of information regarding curriculum and instruction in ITS (Halff, 1988). Halff believes instructional design’s systematic decomposition of instructional problems, comprehensive coverage of instructional applications, and cognitive classification of instructional objectives are almost completely absent from the intelligent tutoring field. Halff declares that the intelligent tutoring field would greatly benefit by applying existing instructional design knowledge, although he further observes that instructional design knowledge in its current form is not helpful regarding two instructional design issues particular to intelligent tutoring: the specificity necessary for formalization and programming, and detailed recommendations for student-tutor interaction.

Tennyson and Park (1987) similarly criticize ITS research for its lack of attention to instructional design principles:

The overall performance of ICAI systems could be significantly improved by integrating learning theories and instructional design principles proposed by psychologists and instructional researchers. ... most systems have focused on the development of man-machine interaction capabilities ... rather than on issues of learning and instruction. Most problems and issues treated in ICAI systems are trivial and task-specific, with little generalizability. ... favorable future development of effective ICAI systems
should be based on instructional theories, rather than on intuition about and technical manipulation of man-machine interactive capability. (p. 327)

Lepper, Woolverton, Mumme, and Gurtner (1993) voice concerns similar to Halff’s: namely that much ITS research does not address even basic issues of pedagogy, and conversely, that current instructional design literature does not address the details of one-on-one tutoring, especially motivational, affective and social goals:

As these systems have become more sophisticated, however, the gap between their obvious intelligence (both in capturing the dynamics of expert understanding and in diagnosing the particular deficiencies in understanding of specific individual students) and their evident limitations as pedagogues has become increasingly apparent. In contrast to the strong principled models of expertise and diagnosis embedded in these systems, the decisions such programs make regarding even basic issues of pedagogy have remained largely ad hoc and underdeveloped.

Moreover, there is remarkably little direct guidance to be gained regarding these issues from the current educational literature. Perhaps because tutoring has been sufficiently infrequent in our schools in this century, until recently virtually no empirical research as specifically examined the dynamics of one-to-one tutorials. ...

Even among the small group of investigators who have recently begun to examine the actions of actual human tutors as a potential source of information relevant to the design of effective computer tutors (e.g., Fox, 1989; McArthur, Stasz, and Zmuidzinas, 1990; McKendree, Reiser, and Anderson, 1984; Putnam, 1987), the focus of analysis concerning expertise in tutoring has remained largely limited to the study of cognitive issues. Questions of motivational, affective or social goals and strategies, and the interaction of these factors with cognitive goals and strategies, have remained virtually unaddressed. (p. 76)

Elsom-Cook is one of the few ITS researchers who has applied instructional design strategies to his ITS work; in Elsom-Cook et al. (1988) he describes a
tutor with seven visibly distinct instructional styles. Nevertheless, Elsom-Cook (1991) notes the lack of theory relating to one-on-one tutoring and the paucity of work at the required level-of-detail for ITS of current instructional design theories:

Virtually [no work has been done] on one-to-one educational interaction.... [What has been done] is descriptive and statistical in nature.... The analysis is at the wrong level of detail for our purposes.... the gap between this level and the detailed level needed for AI approaches is too great to bridge.

The questions which are asked and answered by the literature on educational research are often meaningless from the perspective of AI. (p. 76)

Capell and Dannenberg (1993) claim their system, the Piano Tutor, is unusual simply because of their effort to use instructional design techniques as the basis of the system, an effort they feel is uncommon in intelligent tutoring systems. They note that many ITS systems "... were created without any particular method of instructional design [and when they do have an instructional design] ... the application of these principles is mostly unwitting" (p. 97).

Anderson (1993) also notes the ITS field’s lack of reference to instructional design theory, and the inevitable consequences:

Early work was driven by the challenge of bringing artificial intelligence techniques to bear on education, but often lacked a coherent, scientific theory of effective education. Interventions were motivated by intuition, as often happens in education, but unlike most educational interventions, there was almost no empirical evaluation. (p. 242)

Although the AI efforts were not without merit and did set the foundation for some of our applications, they reflect the fundamental flaws in pursing educational endeavors cut off from concern with a scientific account of human cognition. (p. 243)
To summarize the above perspectives: ITS researchers do not, as a rule, refer to or apply instructional design theory when building tutors. However, if they were to attempt to do so, they would not find a sufficient level of detail in current instructional design prescriptions (since they were written for humans, not computers). Nor would they find sufficient information about one-on-one tutoring. Nevertheless, as LEAP shows, ITS research would benefit from instructional design theory to the extent it can be applied.

Two further comments may be made regarding the relationship between ITS research and instructional design. First, the representation of tutoring expertise should be explicit and in a cognitively plausible form, as is the representation of domain expertise and for the same reason, i.e., acceptance by its users, who are instructors in the former case, students in the latter. Characterizing a tutor’s strategies in terms of instructional design constructs would be a step in the right direction. Second, since ITS researchers must of necessity extend the domain of instructional design, not only should ITS researchers learn from instructional design research, but also instructional design researchers should learn from ITS research. A cooperative endeavor would appear to be the most fruitful approach.

3.3.2 Instructional Theories Suitable for ITS

Instructional expertise consists of the principles of instruction as found in the various theories and models of instruction. Some theories seem particularly applicable to intelligent tutoring systems and are applied in LEAP’s design. For example, Carl Rogers’ theory (1982) focuses on enabling trainees to make their own educational decisions. Glasser’s theory (1969) deals with learners, not in isolation, but as members of a group, who need to consider their behavior in light of the group’s needs (as in the workplace) and make learning decisions that are, as Glasser calls them, real, responsible and right. John Dewey’s theory (1937) proposes a method for developing expert knowledge. Bloom (1971) and Carroll’s
(1971) mastery learning and direct instructional methods expand on the traditional methods many readers would find familiar. Collins, Brown, and Newman (1989) describe the characteristics of ideal learning environments including the ability to: demonstrate the skill, coach the learner, scaffold and fade as required; and provide means for and encourage: articulation, reflection, and exploration. Finally, Gagne (1979, 1985) has the oldest, most comprehensive and most detailed theory of instructional design. He describes instruction as a nine step process; the details of each step vary according to which of five kinds of learning are expected to take place. Thus, a number of established models of instructional expertise can be applied to intelligent tutoring systems, improving the fragmentary, ad hoc models that predominate in current systems.

Tutorial interaction with the student should not be based on emulating a human tutor's interactions (which conventional instructional design theories assume), especially not on natural language understanding, but on using the computer as a tool to optimize learning, which means applying not only instructional design theory, but also principles of computer-human interaction and cognitive psychology to the construction of tutors. Furthermore, for the benefit of generality, the separation of domain knowledge and tutoring knowledge should be maintained (tutoring knowledge is about how to teach domain knowledge).

3.4 Practical Intelligent Tutoring Systems

ITS has long been an AI research activity. Typically researchers attempt to make some contribution to one aspect of intelligent tutoring. Few ITS have seen more than a handful of trainees, and those trainees have been viewed as subjects

---

1 Of course, students can learn to learn at the same time they are learning some other domain that is the nominal subject of instruction. Instruction for teaching students to monitor and improve upon their own learning processes might someday reasonably be expected to be a part of every tutoring shell. The tutoring knowledge will teach itself to the trainee! Nor is it in the realm of pure fantasy that tutors monitor the effectiveness of their own tutoring processes and improve their instruction (Sleeman & Brown, 1982; Lesgold, 1992).
testing the software, not as learners whose goal is to acquire a skill. There have been even fewer attempts to build production-quality ITS applications. Notable exceptions, discussed next, are the efforts of Anderson, Johnson, Lesgold and Woolf. At the 1993 U S WEST workshop on practical tutors several more works in progress were presented, but no fielded systems. A review of the proceedings of a recent conference devoted to ITS, Artificial Intelligence in Education (Brna, Ohlsson, Pain, 1993) added only one more system to this short list.

Anderson’s tutors were developed to test his theory of learning (ACT-R), which is that learning can be characterized as learning production rules by the process of analogy (Anderson, 1993). Because his goal was not to do AI research, but to test a theory of learning, Anderson has for years run experiments on groups of college students as they learned elementary programming skills on his Lisp Tutor. In the process of developing and refining his theory of learning Anderson has demonstrated several things of interest to those who would develop practical tutors. First, he has found that the domain knowledge for a one-semester course takes two or more years to develop: “Our own experience is that the time we take to develop instructional modules in our system is no different than the figures typically cited for conventional CAI.” (Anderson, 1993, p. 254). Second he has focused the tutor on supporting the practice of procedural skills (i.e., exercises), confining related presentations to a textbook (Anderson, Corbett, and Reiser, 1987) or lecture. Third, he has shown that a reasonably general-purpose tutoring shell can be produced (i.e., a shell for programming languages). Finally, Anderson’s success with ITS has led him to begin a program of building and testing algebra and geometry tutors in Pittsburgh’s public schools. So far the program seems to be reasonably successful (J. R. Anderson, personal communication, August, 1994).
Anderson (1993) believes there are few practical tutors simply because building them was never a design goal; instead "... many researchers took up intelligent tutoring because it promised to be a challenging domain in which to explore AI issues" (p. 244).

Johnson and Norton (1992) have been working on practical tutors for over a decade. They characterize the issues and constraints of practical tutors as follows:

Development in applied training environments shifted the laboratory focus on experimental evaluation to a focus on delivery issues such as hardware, interface, knowledge-base development, and so on. This transition made the research team acutely aware that software design is often driven not solely by scientific findings, but also by such constraints as resources, existing computer hardware, organizational politics, and preconceived opinions of managers, instructors, and students. Knowledge engineering in operational environments is also very different than building "proof of concept" systems in the laboratory. (p. 201)

One of their systems, SB-3614, which taught troubleshooting of a tactical switchboard, was used by the Army for four years, until the equipment was retired. Another, DGSIM was fielded by EPRI in 1987 and is still in use. In all Johnson and Norton have been involved in fielding eight systems, with varying degrees of intelligence. "We know that intelligent tutoring systems (a.k.a. intelligent simulations) can be developed at a reasonable cost and delivered outside the laboratory" (Johnson and Norton, 1992, page 212)

Lesgold built Sherlock I and II as part of research into computer-based apprenticeship environments (Lesgold, Eggan, Katz, and Rao, 1992). In 1994, a program was under way, the Maintenance Skills Tutor Program, to build and field a number of similar systems for the Air Force. The main feature of these Sherlock-type systems is their ability to present a sequence of increasingly complex exercises to the trainee while building up a model of the trainee’s performance for
use in reflective follow-up. Lesgold observes that providing a simulated environment in which trainees can practice solving complex problems that rarely occur in the real world is itself worthwhile, an observation that has long been acted upon by astronaut and aircraft pilot trainers. From this perspective, an ITS could be defined as a simulation of some environment, a set of exercises for the student to perform in that environment, and some tutoring facilities that provide further coaching and feedback than what the student would receive naturally from the simulation itself.

One of the first commercially successful intelligent tutors was the Recovery Boiler Tutor, or RBT (Woolf, Blegen, Jansen, and Verloop, 1986). Built under guidance from Woolf, RBT trains paper mill recovery boiler operators to handle both normal and emergency situations. Now in use at hundreds of paper mills throughout the world, RBT has cut insurance costs and helped transition paper mills from the age of hands-on, direct control of the operation, with the process accessible to operators’ hands and eyes, to hands-off control-room operation where the operators must acquire a conceptual understanding of the process, and attribute meaning to the dials and gauges that display process variables (Zuboff, 1988).

The Workshop on Practical Tutors held at U S WEST Advanced Technologies in Boulder CO in July of 1993, attracted representatives from only nine locations. Besides the tutors already mentioned (RBT, Sherlock, and Johnson’s work) no one could make a strong claim for actually having fielded a practical tutor, though several tutors presented were at least attempts to do so, namely Bellcore’s WITS tutor (Lefkowitz & Farrell, 1991), NYNEX’s Grace tutor (McKendree, 1992), Loftin’s Physics Tutor (Loftin, Engelberg, & Benedetti, 1994), and U S WEST’s LEAP tutor.
At the most recent conference devoted to ITS, Artificial Intelligence in Education (Brna, Ohlsson, Pain, 1993), only one of sixteen sessions and one of six workshops were focused on practical applications of ITS. The session was devoted to authoring tools, which implies the intention of building applications, and the workshop was devoted to deployment issues.

The conference session on authoring tools had four papers: In one, the tools had been implemented but no usage was reported. In another, the authoring tools were used by ten subjects to author tutorials. In the third, the authoring tools were in the prototype stage. The fourth paper (Munro, Johnson, Surmon, & Wogulis, 1993) described the Rides authoring system. While the paper itself did not mention trainees, K. Warren (personal communication, September, 1994) stated that the Galaxy Corporation is using Rides to build a tutor as part of the Maintenance Skills Tutor Program mentioned above. In sum, only one of these systems is actively being used to create applications.

The conference workshop: Real-World Issues in Deploying Intelligent Tutoring Systems, was chaired by Bowen Loftin and Beverly Woolf. The workshop "... explore[d] why, in spite of years of effort in developing many intelligent tutors, only a modest number have found acceptance and been successfully deployed" (p. 609). Further details were not reported in the proceedings.

To summarize, few practical ITS have been attempted, perhaps because ITS research is dominated by AI researchers rather than psychologists or educators (Anderson and Lesgold are psychologists, Woolf is an educator as well as an AI researcher; their systems account for most of the practical ITSs). Given an ITS shell, the cost of acquiring domain knowledge may be within an order of magnitude of CBT authoring. These practical tutors focus on exercises in which trainees take actions that are interpretable by the tutor, they do not focus on natural language interaction or multi-media presentations. A few large corporations, and
the Air Force, are attempting to build practical tutors but as yet have little to show for their efforts.

3.5 Evaluation

The tutoring effectiveness of ITSs is rarely evaluated, ITS researchers are generally satisfied with creating software that addresses some difficult AI problem. Winne (1993) suggests researchers perform evaluations that reflect their funder's goals. He also suggests: evaluation of an ITS' power to individualize its instruction, evaluation of an ITS' ability to incorporate new instructional methods, evaluation of the reliability of the student model, and triangulation of evaluation data.

3.5.1 Tutoring Effectiveness of ITSs Is Rarely Evaluated

One of the first observations readers of the ITS literature might make is that there is little evaluation of ITS systems. Anderson (1993) noted:

At the most recent International Conference on Intelligent Tutoring Systems (Frasson, Gauthier, and McCalla, 1992) approximately 25% of the 72 (non-invited) papers included empirical evaluations. Only 5 of these papers, however, assessed the pedagogical effectiveness of a learning environment by comparing it to some other learning experience and examining student performance. Although many research topics do not call for such pedagogical evaluations, this is a minuscule percentage for the field as a whole. (p. 249)

Twidale (1993) points out that the issue is not necessarily one of no evaluation but of conflicting paradigms of evaluation:

Psychology and Education Research belong to the scientific paradigm which lays great stress on the formal objective summative experiment as a means of justifying theories. By contrast, parts of computing research are more closely associated with the engineering paradigm, which employs proof by construction; if the program works in the manner expected then the theory has been justified. ... The culture clash can lead to different expectations about the necessity of experimental evaluation. (pp. 159-160)
In fact, Anderson (1993) essentially concurs with Twidale's reasoning:

... many researchers took up intelligent tutoring because it promised to be a challenging domain in which to explore AI issues. There are at least two disadvantages inherent in such a situation. First, relatively few systems have been brought to the point that they actually work... Second, there has been little evaluation of the pedagogical effectiveness of the systems. Success is not measured in how well the system instructs, but in how well it handles some of the difficult problems of artificial intelligence. (p. 244)

To summarize, since exploring AI issues has been the predominant goal of ITS research, and for AI researchers in the engineering paradigm, running code is a suitable result, only rarely has student learning been an evaluation objective.

3.5.2 Salient Evaluation Issues

One of the few references on evaluation of ITS is a special issue of The Journal of Artificial Intelligence in Education (1993) with seven articles on evaluation.1 Winne (1993) contributed an article synthesizing work done to date and offering a critique of the field. The paragraphs that follow describe how LEAP addresses some of Winne’s points. First, Winne (1993) suggests that rather than bicker about the appropriate evaluation methodology, researchers ask evaluation questions that reflect their sponsor’s goals:

... competent judges will hold varying standards of worth and value to be applied to an evaluation of ALEs2 ... Such tensions ... are an inherent part of political processes that ... will determine whether and how ALEs will be used. ... a suggestion about how to approach usefulness: Ask the user about data that will make a difference, then adapt methodological tools to those purposes.” (p. 329)

---

1 An article by Littman and Soloway (1988) at least introduced the subject of ITS evaluation in a general foundational ITS text.

2 Winne uses ALE (Adaptive Learning Environment) in place of ITS.
In the case of LEAP, the sponsor’s questions were: Does the tutor function as designed? Without crashing? Do trainees and instructors like it? Can they, and instructional designers, use it? Does it individualize instruction? Do the trainees learn? LEAP’s evaluation addresses these questions. Neither management nor instructors at US WEST were interested in a ‘horse race’ comparing tutor-based instruction with the conventional method (Winne, 1993; Anderson, 1993). They did not see this sort of evaluation as a test of the tutor but as a test of the students (and perhaps, of the instructors) and as an attempt to treat groups of students differently.

Winne (1993) in reviewing the special issue of AI in Ed on Evaluation further commented:

... the power of an ALE to adapt its instruction to a student based on data it collects during tutorials. ... is a major topic of basic research into ALEs. Its relative omission ... highlights current limits in developing ALEs ... (p. 318)

In the evaluation of LEAP this topic was not omitted. Two measures of LEAP’s ability to adapt its instruction to individual students were employed: one determined the variation in the sequencing of exercises as a function of student performance, the other examined the similarities and differences in learning styles among three trainees who attained similar final scores.

Another issue that concerned Winne was the flexibility or adaptability of the tutoring module, in particular could it be revised to incorporate new findings about effective instruction? In fact, LEAP’s field trial revealed that a few of the trainees who ignored LEAP’s recommended study methods outperformed trainees who followed them. Analysis of these trainees’ methods revealed strategies that could be input into LEAP’s tutoring strategies by simply manipulating a few sliders on the authoring screens. On the negative side, some trainees
who ignored LEAP’s recommended study methods underperformed trainees who followed them. The more effective instruction in this case is to tell the trainees to follow LEAP’s recommendations when they are floundering. To incorporate this second finding would require a programming change, though not a major one.

Winne (1993) observed that because ALEs adapt to students:

... the reliability of an ALE’s measurements of students’ individual differences to which the system adapts -- motivation, knowledge, and skills -- is a quality that must be assured, and therefore, examined ... The reliability of data about students’ individual differences and the means by which an ALE takes account of reliability in adapting to a student’s learning should be prime targets in ALE evaluation studies. (p. 320)

To paraphrase Winne, first, the reliability of the measurements upon which the tutor’s adaptations are based should be examined and assured, and second, how the tutor deals with reliability when adapting should be evaluated. In LEAP, measurements are made repeatedly and adaptations are made conservatively. The main goal of LEAP, as an ITS for training, is to be certain that trainees can perform some task correctly; too much practice is a tolerable error, too little practice is not. One of LEAP’s adaptations, based on the student model, is to cease presenting an item for practice when a student demonstrates that she or he knows it. The method used in LEAP to ensure a reliable measure of student performance does not follow the nominal prescriptions for checking reliability: test-retest, equivalent forms, or split-half, (Rosenthal and Rosnow, 1984, pp. 81-82) but is similar to these. A trainee practices an SA pair until its score\(^1\) reaches a predetermined cutoff point. One component of the total score is the trainee’s average score for the SA pair, another is the trainee’s consecutive correct (or con-

\(^1\) The score is a weighted sum of three components as described in the section on LEAP’s student model.
secutive incorrect) score for the SA pair. While a high score on one measure can compensate to some extent for a low score on the other, both scores must be reasonably high for the student to 'know' the item. LEAP's scores are reliable because LEAP requires both extensive practice, and high scores on two different performance measures. Finally, LEAP takes reliability into account by being conservative in taking the decision that a student knows something.

There is another issue regarding reliability that neither Winne nor others in the ITS field seem to have considered: the possibility that the student's knowledge changes faster than the student model can reliably determine its state. If this is so (and I believe it is), then tutors shall always require 'too much practice' simply to ensure the student model can reliably state that the student 'knows' the item in question.

Finally, Winne recommends triangulation (Cook and Campbell, 1979) as a means of strengthening claims that the tutor has achieved an objective. The evaluation for LEAP contains substantial triangulation; the evaluation: asked questions of trainees and instructors; measured trainee perceptions of usability and measured actual usage of commands and functions; measured trainee perception of learning and measured actual learning; measured trainee perception of individualization of instruction and measured actual individualization of instruction.

3.6 Summary

ITS need representations that will support not only the performance of the task the trainee is learning, but also the performance of a variety of instructional actions as well. These instructional actions cannot be simple and ad hoc but must be comprehensive, theory-based, supported by research results, and must address learning issues the actual users of the tutor have. Only a few practical tutors have been attempted, and these have been outgrowths of the work of psychologists or educators, rather than AI researchers. Finally, neither the pedagogi-
cal effectiveness of ITSs, nor the unique abilities of ITSs, nor the functionality of the critical factors in ITS performance have been the focus of ITS evaluation.
CHAPTER 4

DESIGN OF LEAP

This research aims to demonstrate a practical intelligent tutoring system and to demonstrate the application of instructional design theory to intelligent tutoring system design. This chapter describes the workplace training environment in which LEAP is intended to function, the job the tutor provides training for, and other training considerations; these are the practical constraints into which LEAP’s design must fit. The chapter then describes the general architecture of the tutor, the domain representation, the student model, the tutoring strategies employed, and their theoretical support. Finally it describes authoring in LEAP, which gives an indication both of LEAP’s practicality and of its ability to be used for tutoring similar materials.

4.1 The Training Task

4.1.1 The Service Representative’s Job

LEAP’s trainees are customer service representatives, or reps, for U S WEST, a regional telephone company serving fourteen western states. Reps sell U S WEST’s products and services. Currently, U S WEST has several thousand employees in customer service or related jobs, and hires or transfers several hundred employees into customer service jobs every year. The training need is twofold: to train new employees, and to continuously update the knowledge and skills of existing reps as product and service changes are made.

The rep’s job is complex. First, service representatives deal with a vast amount of frequently-updated information about the products and services
available.\footnote{During the eighteen months LEAP was being developed and tested, the Service Rep’s foot-thick Desk Reference on the author’s desk received a stack of changes about 6 feet thick. The Desk Reference is printed single-spaced on both sides of 8 1/2 x 11 paper. The paper-based Desk Reference is currently (1995) being replaced by an on-line version.} Second, reps must be careful to comply with all regulations, since the company is highly regulated at both the federal and state levels. Third, reps must conduct a conversation with a customer while simultaneously inputting and retrieving information from various databases. Table 4.1, Voice Messaging Dialogue, is an example of a typical conversation with a customer.
Table 4.1
Voice Messaging Dialogue

<table>
<thead>
<tr>
<th>Actor/Action</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Situation:</td>
<td>The telephone beeps. A new customer is on the line.</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;U S WEST Communications, this is Josephina. How may I help you?&quot;</td>
</tr>
<tr>
<td>Customer:</td>
<td>&quot;Hello, my name is Pat Garrett and I want that messaging service you advertise on TV.&quot;</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;Certainly Pat, I can help you with that.&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>Telephone #: <em><strong>-</strong></em>-____</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;May I have your telephone number with area code please?&quot;</td>
</tr>
<tr>
<td>Customer:</td>
<td>&quot;It is 303 226-1354&quot;</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;Thank you Pat, it will take just a moment for me to get your records. That number was 303 226 1354.&quot;</td>
</tr>
<tr>
<td>Rep types:</td>
<td>&quot;303 226 1354&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>Customer's account is not past due. Credit information is up to date, etc..</td>
</tr>
<tr>
<td>Conclude:</td>
<td>All prerequisite order activity is complete.</td>
</tr>
<tr>
<td>Database:</td>
<td>Calling Party: ______________</td>
</tr>
<tr>
<td>Rep types</td>
<td>&quot;PAT&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>Telephone #:<em><strong>-</strong></em>-___</td>
</tr>
<tr>
<td>Rep types</td>
<td>&quot;303 226 1354&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>Enter selection: _</td>
</tr>
<tr>
<td>Rep types</td>
<td>&quot;10&quot; (10 is the menu selection for this kind of order)</td>
</tr>
<tr>
<td>Database:</td>
<td>_ SLF01</td>
</tr>
</tbody>
</table>

Continued, next page
Table 4.1, continued

<table>
<thead>
<tr>
<th>Actor/Action</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rep types</td>
<td>&quot;X&quot;, and presses the Enter key.</td>
</tr>
<tr>
<td>Database:</td>
<td>VOICE MESSAGING 6.95</td>
</tr>
<tr>
<td>Conclude:</td>
<td>Voice Messaging is available in the customer's area.</td>
</tr>
<tr>
<td>Database:</td>
<td>1FR (Customer has a private line with unlimited local calling.)</td>
</tr>
<tr>
<td>Conclude:</td>
<td>Customer's class of service is suitable for Voice Messaging.</td>
</tr>
<tr>
<td>Database:</td>
<td>CALL WAITING</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;I see you currently have Call Waiting. It will work for you as it does today. When you do not want to answer your Call Waiting and have your calls go directly into your Voice Messaging Service, simply dial *70 before making your outgoing call.&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>CALL FORWARDING VARIABLE; 1 LINE</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;I see you also have Call Forwarding. Call Forwarding will override Voice Messaging; when you forward your calls, Voice Messaging will not work. And, if you do not want your phone to ring, you can forward your calls directly into your Voice Messaging by dialing a special number. I will include those instructions for you. Do you have any questions on how Voice Messaging works with Call Waiting and Call Forwarding?&quot;</td>
</tr>
<tr>
<td>Customer:</td>
<td>&quot;No, just send me those instructions.&quot;</td>
</tr>
<tr>
<td>Database:</td>
<td>_ VOICE MESSAGING 6.95 8.50</td>
</tr>
<tr>
<td>Rep types</td>
<td>&quot;1&quot;, and press the Enter key.</td>
</tr>
<tr>
<td>Database:</td>
<td>RING OPTION: _</td>
</tr>
<tr>
<td>Service rep:</td>
<td>&quot;Voice Messaging is set up to answer your calls automatically after a certain number of rings. How many times does the phone normally ring before you pick it up?&quot;</td>
</tr>
<tr>
<td>Customer:</td>
<td>&quot;Usually 3 to 4&quot;</td>
</tr>
</tbody>
</table>

Continued, next page
Rep types

Service rep: "3", and presses the Enter key.

Service rep: "OK, I can set it to pick up after three to five rings. Now if that does not work out for you, you can change it once for free."

Database: OUR NEXT APPOINTMENT DATE IS: __/__/__

Service rep: "I can program the Voice Messaging for you in four working days; that would be Tuesday the 24th. Does that work for you?"

Customer: "That's great."

Rep types "04 24 94", and presses the Enter key.

Service rep: "OK, Pat, let's review. I will be adding Voice Messaging to your line, telephone number 226-1354, on Tuesday, the 24th. Your calls will forward to Voice Messaging after 3 to 5 rings. The monthly rate will increase by $6.95, and the set-up fee is a one-time charge of $8.50."

"Voice Messaging is optional, it is not required for basic phone service and there are other companies that do provide a service similar to our Voice Messaging. I'll mail you a brochure that will explain the Terms and Conditions of Voice Messaging and I'll also mail you the "User Guide" which gives complete instructions on how to use your Voice Messaging."

"Pat, let me give you some information on using Voice Messaging. Do you have a pen and paper handy?"

Customer: "Yes, go ahead."

Service rep: "To retrieve your messages simply call 225-6182. It will ask you for a security code. You pick your own, but to get you started, your temporary security code is 1234. And one more number, in case you have any problems the Help Center is available and their number is 1-800-669-7676. You know you have messages when you hear the stutter dial tone."

Service rep: "Did you get it all down?"

Continued, next page
Currently, training consists of about 10 weeks of traditional classroom instruction in groups of 5 to 7 trainees. Trainees spend about one quarter of class time practicing what they have learned by doing role-plays with each other, while the instructor observes and coaches them. Coaching role-plays is not easy for instructors; upon seeing LEAP for the first time, one instructor said:

I think it’s wonderful I won’t have to do role-plays anymore, day in, day out. I despise role plays. After the 551st role play it’s a little hard to be creative. What do you do with the rest of the class while you’re sitting there doing a role-play? They’re doing whatever they’re doing wrong, and they just keep doing it wrong, until you get over there and correct it, so this would give them the opportunity to practice ...

Trainees also visit the workplace to observe experienced reps and to practice what they have learned. Customers’ calls cannot be pre-sorted, so trainees rarely get to observe or practice something they have studied re-
ently. In fact, because customers call U S WEST for a wide variety of reasons, neither trainees nor new reps get sustained practice on any one topic; this is part of the reason they need one or more years to become competent.

A number of personnel in various roles have an interest in service representative training: U S WEST’s training department, called U S WEST Learning, has responsibility for initial service representative training. U S WEST’s Home and Personal Services department employs the trainees and is responsible for their continuing training. The service representatives’ union, Communications Workers of America (CWA), is of course, interested in the well-being of its members. Thus trainees, trainers, instructional designers, union officials, and managers from the training and customer service departments all have an interest in the tutor. U S WEST Learning (and its internal customers) are not technology averse; currently they provide instruction on interactive video-disk, deliver distance learning via 2-way video, have an electronic learning group building performance support systems, and exhibit a generally positive attitude toward technological change.

4.1.2 Role of an ITS in Training

A practical ITS must fit into and improve the existing training process (Reusser, 1993). Trainees must feel they learn more by using the tutor than they do without it. Instructors must feel training is better when they have the tutor in their classrooms. Instructional designers must believe they can input and modify the contents of the tutor. Training managers must see a positive cost/benefit analysis and a reliable technology. Union officials prefer to see their members becoming more skilled and receiving higher pay than to see

---

2 Many people in these varied roles began their U S WEST careers as service representatives.
them being replaced by machines. The service organization wants its employees to be well-trained on the latest products and services at the lowest cost.

Trainees and trainers should directly benefit from the instructional design principles built into the tutor, as described later in this chapter. Instructional designers should find the built-in abilities to their liking, should be able to modify them as needed, and should be able to add new instructional materials to the tutor via its authoring system. The other interested parties should benefit from improved training.

4.1.3 Overall Design of the ITS

LEAP's design must support the task of training customer service representatives. For example, the terminals, keyboards, and headsets the trainees use with LEAP must be identical to those used on the job. The tutor must fit into existing training processes for both initial and continuing training (which it does as an improvement on the role-play).

4.1.4 Hardware; Terminals

Service reps currently access U S WEST's databases via networked UNIX workstations with large (19 inch) color monitors. LEAP is designed to be used on the same monitors and keyboards reps currently use. However, the network used for database access cannot handle the increased load that instructional videos would place on it, nor can the workstations at the reps' desks handle training audio that simulates customer calls. As a result, LEAP is currently delivered on an independent network of Sun Sparc workstations.

LEAP was written in Lisp, and uses a commercial software package, VAPS, for multimedia. Video is displayed in Uniflix; audio is captured and played with the Sun Audiotool; the windowing environment is Openwindows. The network server is a Sun Sparc 10 Model 42, 128 MB memory, with 200 MB swap space; LEAP software is located on a 2 GB external
hard drive; the network is a twisted pair hub using 10Base T Ethernet connections. The local display machines for trainee use are 2 Sun Sparc IPXs, 1 Sun Sparc 2, 2 Sun Sparc 10 model 30s, and the server.

4.1.5 A General Architecture

LEAP's architecture roughly approximates the classic ITS architecture. Domain or expert knowledge is stored in a knowledge base independent of the remainder of the tutor and can be replaced with other domain or expert knowledge, allowing the tutor to instruct similar domains with little reprogramming. The student model is an overlay on the domain knowledge. Instructional knowledge is hard-coded (but tutoring decision-making parameters are adjustable through an authoring interface). There are two tutoring interfaces, or simulations: one, for verbal conversations, is general; the other, for databases, is specific to the database being trained; it can be modified with the help of a programmer to simulate other mainframe databases. Authoring of the domain knowledge base is done primarily with a text editor, individual conversations or exercises are input with a special editor, and as mentioned, instructional parameters are adjusted using an authoring interface.

4.2 The Domain Representation

LEAP simulates the trainee's work environment. Simulated customers call with requests of various sorts and the rep must respond to the customer appropriately while simultaneously retrieving, entering, or updating customer account information in a simulated database. The simulated work environment is built using a domain representation whose elements are described in this section. The main items in the domain representation are Situation-Action pairs, conversations, the domain knowledge base, and topics.
The core item in LEAP's domain representation is the *Situation-Action pair* (SA pair). SA pairs are analogous to rules in expert systems. The situation side of the SA pair contains a situation LEAP will present to the service rep trainee at some point in a conversation, and the action side of the SA pair contains the action(s) an expert rep would take in response. Here are two SA pairs:

**Situation:**
Database shows that customer has a private line.

**Action:**
Conclude that customer's service is suitable for Voice Messaging.

**Situation:**
Database shows that customer has a party line.

**Action:**
Explain that Voice Messaging requires a private line, & options.

Notice that these SA pairs are rather abstract. They do not describe precisely what the database shows or what the expert rep would say. These details vary from conversation to conversation and are presented in specific conversations. Here are two *instances* of SA pairs taken from a conversation:

**Situation:**
Customer: "I'd like to add Voice Messaging to my service."

*and*
Database Screen01: Name _____ Number ( ) ___-___

**Action:**
Rep replies: "I can help you with that; may I have your name and number with area code please?"

**Situation:**
Customer: "My name is John Smith and my number is 303 541-6412."

**Action:**
Rep enters: John (303) 541-6412
From an analysis of the domain, it was determined that three main types of Situation occur:

- a database field providing or requiring information
- a mental recollection by the rep

Likewise, three main types of Action occur:

- talk with the customer
- enter information into the database
- conclude something about the situation

Compound situations and actions may also exist: For example, when the situation is the Due Date database field, the rep's actions are to simultaneously

- tell the customer when the service can be provided, and
- type the due date into the field.

A conversation is a sequence of SA pairs. Conversations are instantiated with the text and speech of 'customers' and expert service reps, customer account information and database I/O activity. On the job, reps spend most of their time conversing with customers, and all of LEAP's exercises are conversations. Conversations are two-layered objects. First, they are a sequence of SA pairs. Second, they are the spoken, mental, or database events associated with the situations and actions of the SA pairs. While the SA pairs are used in many conversations, the spoken, mental, or database events associated with them are unique to each conversation. The spoken events are stored as text and as audio, the mental events are stored as text, and the database events are stored as executable code. A conversation, with SA pairs labeled, and the type of situation or action indicated for each situation and action, is presented in Table 4.2.
Table 4.2
Script for Adding Voice Messaging Service (VMS) (to an account with Call Waiting and Call Forwarding)

<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Customer Contact</td>
<td>1</td>
<td>Situation:</td>
<td>The telephone beeps. A new contact is on the line.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Telephone</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action:</td>
<td>Say &quot;U S WEST Communications, this is Josephina. How may I help you?&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Talk</td>
<td></td>
</tr>
<tr>
<td>Service change is Add Voice Messaging</td>
<td>2</td>
<td>Situation:</td>
<td>&quot;Hello, my name is Pat Garrett and I want that messaging service you advertise on TV.&quot;</td>
</tr>
<tr>
<td>Get Name and Number</td>
<td>3</td>
<td>Situation:</td>
<td>Telephone #: <strong><strong>-</strong></strong>-____</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BOSS</td>
<td>Say &quot;May I have your telephone number with area code please?&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Talk</td>
<td></td>
</tr>
<tr>
<td>Topic</td>
<td>SA pair</td>
<td>Type</td>
<td>Instance of Situation or Action</td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>--------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>Situation Customer</td>
<td>“It’s 303 226-1354”</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action Talk</td>
<td>Say “Thank you Pat, it will take just a moment for me to get your records. That number was 303 226 1354.”</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AND</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action BOSS</td>
<td>Type “303 226 1354”</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Situation BOSS</td>
<td>Customer’s account is not past due.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action Conclude</td>
<td>All order prerequisite activity is complete.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Situation SONAR</td>
<td>Calling Party:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action SONAR</td>
<td>Type “PAT”</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Situation SONAR</td>
<td>Telephone #:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action SONAR</td>
<td>Type “303 226 1354”</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Situation SONAR</td>
<td>Enter selection:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Action SONAR</td>
<td>Type “10” (Menu selection for Change Orders)</td>
<td></td>
</tr>
<tr>
<td>Topic</td>
<td>SA pair</td>
<td>Type</td>
<td>Instance of Situation or Action</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------</td>
<td>------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td>Check VMS Availability</td>
<td>9</td>
<td>Situation</td>
<td>SONAR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action</td>
<td>SONAR</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Situation</td>
<td>SONAR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action</td>
<td>Conclude</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action</td>
<td>Conclude</td>
</tr>
<tr>
<td>Determine Class of Service</td>
<td>11</td>
<td>Situation</td>
<td>SONAR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action</td>
<td>Conclude</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verify Feature Compatibility</td>
<td>12</td>
<td>Situation SONAR</td>
<td>...&lt;br&gt;CALL WAITING ...&lt;br&gt;Action Talk Say &quot;I see you currently have Call Waiting. It will work for you as it does today. When you don't want to answer your Call Waiting and have your calls go directly into your Voice Messaging Service, simply dial *70 before making your outgoing call.&quot;</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>Situation SONAR</td>
<td>...&lt;br&gt;CALL FORWARDING VARIABLE; 1 line ...&lt;br&gt;Action Talk Say &quot;I see you also have Call Forwarding. Call Forwarding will override Voice Messaging; when you forward your calls, Voice Messaging will not work. And, if you don't want your phone to ring, you can forward your calls directly into your Voice Messaging by dialing a special number. I will include those instructions for you. Do you have any questions on how Voice Messaging works with Call Waiting and Call Forwarding?&quot;</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>Customer</td>
<td>“No, just send me those instructions.”</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Conclude</td>
<td>OK to continue.</td>
</tr>
<tr>
<td>Order VMS</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Situation</td>
<td>_ VOICE MESSAGING 6.95 8.50</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>SONAR</td>
<td>Type &quot;1&quot;, and press the Enter key.</td>
</tr>
<tr>
<td>Determine Ring Cycle</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>Situation</td>
<td>RING OPTION: _</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Talk</td>
<td>Say &quot;Voice Messaging is set up to answer your calls automatically after a certain number of rings. How many times does the phone normally ring before you pick it up?&quot;</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>Situation</td>
<td>“Usually 3 to 4”</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>SONAR</td>
<td>Type &quot;3&quot;, and press the Enter key.</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>Situation</td>
<td>Legal guidelines require ‘free ring cycle change’ notification.</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Recall</td>
<td>Say “OK, I can set it to pick up after three to five rings. Now if that doesn’t work out for you, you can change it once for free.”</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negotiate Four-Day Due Date</td>
<td>19</td>
<td>Situation SONAR</td>
<td>OUR NEXT APPOINTMENT DATE IS: <strong>/</strong>/__</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action Talk</td>
<td>Say &quot;I can program the Voice Messaging for you in four working days; that would be Tuesday the 24th. Does that work for you?&quot;</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>Situation Customer</td>
<td>&quot;That's great.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action SONAR</td>
<td>Type “04 24 94”, and press the Enter key.</td>
</tr>
<tr>
<td>Verify Order:</td>
<td>21</td>
<td>Situation Recall</td>
<td>Contact Standards require order to be verified.</td>
</tr>
<tr>
<td>Add VMS</td>
<td></td>
<td>Action Talk</td>
<td>Say &quot;OK, Pat, let's review. I will be adding Voice Messaging to your line, telephone number 226-1354, on Tuesday, the 24th. Your calls will forward to Voice Messaging after 3 to 5 rings. The monthly rate will increase by $6.95, and the set-up fee is a one-time charge of $8.50.&quot;</td>
</tr>
<tr>
<td>Provide Full-Disclosure</td>
<td>22</td>
<td>Situation Recall</td>
<td>Full disclosure requires 'optional service' disclosure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action Talk</td>
<td>Say &quot;Voice Messaging is optional, it is not required for basic phone service ...&quot;</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>23 Situation Recall</td>
<td></td>
<td>Recall</td>
<td>Full disclosure requires 'competitive service' disclosure.</td>
</tr>
<tr>
<td>Action Talk</td>
<td></td>
<td>Talk</td>
<td>Say &quot;... and there are other companies that do provide a service similar to our Voice Messaging. &quot;</td>
</tr>
<tr>
<td>24 Situation Recall</td>
<td></td>
<td>Recall</td>
<td>Full disclosure requires a contract.</td>
</tr>
<tr>
<td>Action Talk</td>
<td></td>
<td>Talk</td>
<td>Say &quot;I'll mail you a brochure that will explain the Terms and Conditions of Voice Messaging ...&quot;</td>
</tr>
<tr>
<td>25 Situation Recall</td>
<td></td>
<td>Recall</td>
<td>Customer requires the &quot;User Guide.&quot;</td>
</tr>
<tr>
<td>Action Talk</td>
<td></td>
<td>Talk</td>
<td>Say &quot;... and I'll also mail you the &quot;User Guide&quot; which gives complete instructions on how to use your Voice Messaging. &quot;</td>
</tr>
<tr>
<td>Describe Voice Mailbox Setup</td>
<td>26 Situation</td>
<td>Recall</td>
<td>Customer needs voice mailbox setup information.</td>
</tr>
<tr>
<td>Action Talk</td>
<td>Recall</td>
<td>Talk</td>
<td>Say &quot;Pat, let me give you some information on using Voice Messaging. Do you have a pen and paper handy?&quot;</td>
</tr>
</tbody>
</table>

Continued, next page
Table 4.2, continued

<table>
<thead>
<tr>
<th>Topic</th>
<th>SA pair</th>
<th>Type</th>
<th>Instance of Situation or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>Situation</td>
<td>Customer</td>
<td>“Yes, go ahead.”</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Talk</td>
<td>Say “To retrieve your messages simply call 225-6182. It will ask you for a security code. You pick your own, but to get you started, your temporary security code is 1234. And one more number, in case you have any problems the Help Center is available and their number is 1-800-669-7676. You know you have messages when you hear the stutter dial tone.”</td>
</tr>
<tr>
<td>28</td>
<td>Situation</td>
<td>Recall</td>
<td>Need to confirm customer got all information.</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Talk</td>
<td>Say “Did you get it all down?”</td>
</tr>
<tr>
<td>29</td>
<td>Situation</td>
<td>Customer</td>
<td>“Yes, I did.”</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Conclude</td>
<td>OK to continue.</td>
</tr>
<tr>
<td>(Full-Disclosure, continued)</td>
<td>30</td>
<td>Situation</td>
<td>Recall</td>
</tr>
<tr>
<td></td>
<td>Action</td>
<td>Talk</td>
<td>Say “I know you will love your Voice Messaging, but just to let you know, if for any reason it doesn’t meet your needs you have seven days to remove the service, in other words, seven days to try it out at no charge.”</td>
</tr>
<tr>
<td>Topic</td>
<td>SA pair</td>
<td>Type</td>
<td>Instance of Situation or Action</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------</td>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>(Verify Order: Add VMS, continued)</td>
<td>31</td>
<td>Recall</td>
<td>Contact standards require PERM note.</td>
</tr>
<tr>
<td>Action</td>
<td></td>
<td>Playthrough</td>
<td>(LEAP enters standard PERM note.)</td>
</tr>
<tr>
<td>32</td>
<td>Situation</td>
<td>Recall</td>
<td>Contact flow requires issuing order now.</td>
</tr>
<tr>
<td>Action</td>
<td></td>
<td>SONAR</td>
<td>Press the PF-7 key.</td>
</tr>
<tr>
<td>Close Customer Contact</td>
<td>33</td>
<td>Recall</td>
<td>Contact standards require that you ask the customer if there is anything else you can do for them.</td>
</tr>
<tr>
<td>Action</td>
<td></td>
<td>Talk</td>
<td>Say “Is there anything else I can help you with today Pat?”</td>
</tr>
<tr>
<td>34</td>
<td>Situation</td>
<td>Customer</td>
<td>“No, I can’t think of anything.”</td>
</tr>
<tr>
<td>Action</td>
<td></td>
<td>Talk</td>
<td>Say “Well, it’s been very nice talking with you, and thank you for doing business with U S WEST Communications.”</td>
</tr>
</tbody>
</table>

**Note.** The types of situation are:

- Customer: the customer says something
- BOSS: the BOSS database presents something
- Recall: the trainee remembers something
- SONAR: the SONAR database presents something
- Telephone: the telephone system beeps

**The types of action are:**

- Talk: the trainee or expert says something
- BOSS: the trainee types something into BOSS
- Conclude: the trainee concludes something
- SONAR: the trainee types something into SONAR
- Playthrough: LEAP always performs the action
The domain knowledge base is a set of abstract conversations in an AND/OR tree, where nodes are SA pairs, and branches are different possibilities based on the customer's situation, e.g., whether they have a private line or a party line, whether they accept a suggestion from the rep or rejects it, etc. Every path through the domain knowledge base, if instantiated, would be a valid conversation. Thus the domain knowledge base may also be called the conversation space, the space of all conversations.

A topic is a portion of the domain knowledge base that is conceptually related. A topic has a name that appears in LEAP's top-level menu, as shown in Figure 4.1. Each type of conversation (e.g., Order Voice Messaging, Remove Voice Messaging, etc.) is a major topic. Each part of a conversation (e.g., Check Voice Messaging Availability, Provide Set-Up Information, etc.) is a sub-topic. Topics are connected by sequential and prerequisite links. Each conversation contains domain knowledge from several sub-topics.
Figure 4.1

Top Level Screen: Topics presented

The three tables below, Tables 4.3, 4.4, and 4.5, show how different conversations between a service representative and a customer might touch upon different topics. Nearly all conversations share the opening, closing, and customer identification topics: Open Customer Contact, Get Name and Number, and Close Customer Contact, respectively. Many conversations share one or more additional topics.

Exercises in LEAP are in the form of conversations; the conversations in these tables illustrate how, while practicing rarely-occurring topics, even to
the minimum level, trainees may waste time in repeating frequently-occurring, already-well-learned topics. Conversely, these tables illustrate how a conversation that is timely, in the sense that it addresses precisely the portion of a topic that a trainee needs to study, may also touch upon several other topics that the trainee has not yet even been introduced to. It is for these reasons that LEAP modifies conversations in Focused Practice exercises (described in the *Focused Practice* section).
Table 4.3

Topics in Conversations: Add VMS

<table>
<thead>
<tr>
<th>Topic Name</th>
<th>Conversation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Open Customer Contact</td>
<td>•</td>
</tr>
<tr>
<td>Service Change is Add VMS</td>
<td>•</td>
</tr>
<tr>
<td>Get Name and Number</td>
<td>•</td>
</tr>
<tr>
<td>Check VMS Availability</td>
<td>•</td>
</tr>
<tr>
<td>Determine Class of Service</td>
<td>•</td>
</tr>
<tr>
<td>Verify Feature Compatibility</td>
<td>•</td>
</tr>
<tr>
<td>Describe How VMS Works</td>
<td>•</td>
</tr>
<tr>
<td>Order VMS on SLF01</td>
<td>•</td>
</tr>
<tr>
<td>Determine Ring Cycle</td>
<td>•</td>
</tr>
<tr>
<td>Negotiate Four-Day Due Date</td>
<td>•</td>
</tr>
<tr>
<td>Verify Order: Add VMS</td>
<td>•</td>
</tr>
<tr>
<td>Provide Full-Disclosure</td>
<td>•</td>
</tr>
<tr>
<td>Describe Voice Mailbox Setup</td>
<td>•</td>
</tr>
<tr>
<td>Service is ONA Restricted</td>
<td>•</td>
</tr>
<tr>
<td>Close Customer Contact</td>
<td>•</td>
</tr>
</tbody>
</table>
Table 4.4

Topics in Conversations: Change Ring Cycle

<table>
<thead>
<tr>
<th>Topic Name</th>
<th>Conversation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Open Customer Contact</td>
<td>•</td>
</tr>
<tr>
<td>Service Change is Change Ring Cycle</td>
<td>•</td>
</tr>
<tr>
<td>Get Name and Number</td>
<td>•</td>
</tr>
<tr>
<td>Discuss Ring Cycle Change</td>
<td>•</td>
</tr>
<tr>
<td>Negotiate Two-Day Due Date</td>
<td>•</td>
</tr>
<tr>
<td>Determine Charge: Ring Cycle Change</td>
<td>•</td>
</tr>
<tr>
<td>Verify Order: Change Ring Cycle</td>
<td>•</td>
</tr>
<tr>
<td>Close Customer Contact</td>
<td>•</td>
</tr>
</tbody>
</table>
### Table 4.5

**Topics in Conversations: Remove Voice Messaging**

<table>
<thead>
<tr>
<th>Topic Name</th>
<th>Conversation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Customer Contact</td>
<td>•</td>
</tr>
<tr>
<td>Service Change is Remove Voice Messaging</td>
<td>•</td>
</tr>
<tr>
<td>Get Name and Number</td>
<td>•</td>
</tr>
<tr>
<td>Attempt Save: Voice Messaging</td>
<td>•</td>
</tr>
<tr>
<td>Branch to: Change Ring Cycle</td>
<td>•</td>
</tr>
<tr>
<td>Negotiate Same-Day Due Date</td>
<td>•</td>
</tr>
<tr>
<td>Change Sales Code to Office Sales Code</td>
<td>•</td>
</tr>
<tr>
<td>Verify Order: Remove Voice Messaging</td>
<td>•</td>
</tr>
<tr>
<td>Transfer Customer Contact</td>
<td>•</td>
</tr>
<tr>
<td>Close Customer Contact</td>
<td>•</td>
</tr>
</tbody>
</table>

LEAP currently has seventeen exercises or conversations. Descriptive names for the exercises are listed in Table 4.6. Exercises are grouped by the main topic they teach. In the Add Voice Messaging ... exercises, emphasis is on how voice messaging (VMS) interacts with line features such as Call Waiting and Custom Ringing, and on VMS availability. The purpose of the remaining exercises is evident.
### Table 4.6

**Conversation Exercises in LEAP**

<table>
<thead>
<tr>
<th>Add Voice Messaging to an Account with:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call Waiting and Call Forwarding</td>
</tr>
<tr>
<td>Custom Ringing and a 1AESS central office switch.</td>
</tr>
<tr>
<td>Custom Ringing and a 5E or a DMS central office switch.</td>
</tr>
<tr>
<td>Enhanced Call Waiting, Three Way Calling, and Caller ID</td>
</tr>
<tr>
<td>Speed Calling and Intracall.</td>
</tr>
<tr>
<td>Measured Service and Vacation Service</td>
</tr>
<tr>
<td>Voice Messaging is not yet available; explain how VMS works</td>
</tr>
<tr>
<td>Voice Messaging is not yet available; customer will call back</td>
</tr>
<tr>
<td>Voice Messaging is not available</td>
</tr>
<tr>
<td>ONA Restriction</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Change Ring Cycle:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer decides to keep original ring cycle</td>
</tr>
<tr>
<td>Customer decides to lengthen ring cycle</td>
</tr>
<tr>
<td>Customer decides to shorten ring cycle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Remove Voice Messaging:</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMS is broken</td>
</tr>
<tr>
<td>VMS is hard to use</td>
</tr>
<tr>
<td>Save the sale, explain how to use VMS</td>
</tr>
<tr>
<td>Save the sale, refer customer to Help Center</td>
</tr>
</tbody>
</table>
4.2.1 Example Topics in the Domain Knowledge Base

The internal structure of topics varies widely, from short and linear to long and multi-branching. Thus the number of conversations that a trainee must practice to master a particular topic also varies widely. Depending on its complexity, it might take from one to six or more conversations just to visit all the SA pairs in a topic. The number of exercises a student must practice in order to master a topic, then, could vary from a few, to a few dozen, depending on the topic and the student.

Some Topics in the Domain Knowledge Base are simple and straightforward, for example, the topic Close Contact consists of two SA pairs as shown in Figure 4.2.

(define-grammar CLOSE-CONTACT
(SEQ
S: Contact standards require that you ask caller if there is anything else you can help with
   A: Ask caller if there is anything else you can help with
      S: Customer is satisfied
      A: Say thank you and good-bye
      (END))
)

Figure 4.2

The topic Close Contact as defined in the Domain Knowledge Base

In one exercise in LEAP, this appears as:

Rep: "Is there anything else I can help you with today, Pat?"

Customer: "No, I can't think of anything."
Rep:

"Well, it's been very nice talking with you and thank you for doing business with U S WEST Communications."

These lines appear in every conversation, yet after practicing them a few times, an occasional review is all that is necessary.

Other portions of the Domain Knowledge Base are more complex; consider, for example, the topic: Determine Class of Service. When a customer calls with a request for VMS, the service rep must respond appropriately, based on the customer's class of service, i.e., the customer may have a private line, or measured service, or a party line. If the customer has a private line (with unlimited service) the rep need only note this and continue to the next portion of the conversation. If the customer has limited service, the rep must inform the customer that all calls to VMS will affect the customer's allowance. The customer may decide not to add VMS, may decide to bear the costs, or may decide to upgrade his or her line\(^1\). In all three cases, the trainee must learn to respond appropriately. Finally, if the customer has a party line, the rep must inform the customer that VMS requires a private line. Again, the customer may decide not to add VMS, or may decide to upgrade his or her line, and the trainee must learn to respond appropriately. In the domain knowledge base, the topic, with these branches, is represented as shown in Figure 4.3.

---

\(^1\) The VMS lesson in LEAP does not teach the trainee what to do if the customer decides to upgrade his or her line since line upgrading is a different skill, and is taught in another part of the course. This is an example of how instruction can improve on real-life experience by providing an opportunity for concentrated practice on one topic at a time.
(define-grammar DETERMINE-CLASS-OF-SERVICE
  (OR
   (SEQ S: Customer has a private line
     A: Conclude: Customer's service is suitable for Voice Messaging)
   (SEQ S: Customer has measured service
     A: Explain all calls to Voice Messaging will affect allowance, & options
     (OR
      (SEQ S: Customer decides to upgrade class of service
        A: Consider service upgraded to unlimited private line)
      (SEQ S: Customer decides not to add Voice Messaging
        A: Begin the Close-contact sequence
        (SEQ S: Customer understands costs
          A: Conclude: Customer's service is suitable for VMS))))
   (SEQ S: Customer has a party line
     A: Explain that Voice Messaging requires a private line, & options
     (OR
      (SEQ S: Customer decides to upgrade class of service
        A: Consider service upgraded to unlimited private line)
      (SEQ S: Customer decides not to add Voice Messaging
        A: Begin the Close-contact sequence))))))

Figure 4.3

The topic Determine Class of Service as defined in the Domain Knowledge Base

How many exercises must the student practice in order to master this topic, Determine Class of Service? The topic has six branches, and the student must learn to handle all of them. There are two factors to consider, first, the number of different branches the student must actually practice, and second, the number of repetitions required for mastery of each branch. The two branches where the customer decides not to add VMS, deal with an event that could occur in any topic (i.e., beginning the close-contact sequence), and as long as they are addressed somewhere, need not be considered further here.
Two other branch endings are nearly identical -- in both cases the customer decides to upgrade their service to an unlimited private line -- but the trainee must learn that the customer has this option in two situations, and both must be taught explicitly. Thus, a total of four branches must be taught.

The second factor to consider is how many conversations must be created to enable the student to master these four branches. A minimum of one conversation per branch would seem to be required, but the designer may choose not to address a branch at all since, for example, customers with party lines rarely attempt to order voice messaging, because party-line customers tend to be either too poor to afford VMS or to live in rural areas where VMS is not available. On the other hand, the designer may decide to instantiate several conversations on the same branch, so trainees practice those SA pairs in a variety of contexts. To determine the minimum number of conversations, the entire domain knowledge base must be reviewed, since both within-topic and between-topic branching must be considered.

One final point about LEAP's domain representation: Instruction in LEAP addresses a domain where expertise requires natural language understanding, a task in which computers have only rudimentary skill. Thus, one way in which this tutor differs from others, particularly Clancey- and Anderson-style (Anderson, Boyle, Corbett, & Lewis, 1990; Clancey, 1986) tutors, is that LEAP's expert model cannot actually perform the task itself. Nevertheless, LEAP can model the task well enough for trainees to learn it, and in a practical tutor, that is what is important. A second 'weakness' of the expert model is that conversations are linear, LEAP does not model training conversations with the flexibility of real conversations. Nevertheless, flexible

2 Perhaps I give these systems too much credit. Neomycin cannot interact with patients and the LISP tutor does not start with a requirements document.
conversations are a non-problem as far as training is concerned. As long as trainees practice each SA pair in several conversations, they will be reasonably prepared to apply it whenever it occurs. In sum, domain knowledge in LEAP is adequate; it is not the main focus of the research.

4.2.2 Domain Representation Summary

The key item in LEAP's domain representation is the SA pair, analogous to the expert system rule. Conversations are sequences of abstract SA pairs and the concrete spoken, mental, or database events associated with them. The domain knowledge base is the space of all conversations in the domain. A topic is a conceptually related portion of the domain knowledge base. The internal structure of topics varies from short and linear to long and multi-branching. The number of conversations in the tutor must be adequate to cover all the important branches of the domain knowledge base, and must provide enough variety for the SA pairs to be learned by the trainee. The student model, outlined in the following section, calculates scores indicating the student's current knowledge of each of the domain representation elements described in this section.

4.3 The Student Model

LEAP is a coached practice environment that optimizes the learning process by maintaining and consulting a detailed student model. LEAP calculates a score for every action, exercise, and topic the student tries, then uses the scores both to provide feedback and to select topics, exercises and actions for the student to practice. Using the student model to focus practice means exercises can be realistic and complex instead of artificial and simple, thereby enabling instructional designers to address issues of motivation, transfer, and
apprenticeship learning. The student model is temporally sensitive, enabling LEAP to adjust its instruction as trainees learn or forget.

If an intelligent tutor is to optimize the learning process, it must concentrate the student's effort at the boundary dividing the already-learned skills from the not-yet-learned. It must eliminate from students' activities those skills they have already learned and have no further need to study, and those they are not yet prepared to learn. The information a tutor needs to optimize the learning process can come only from a detailed student model.

LEAP refers to its student model when making tutoring decisions. For example, LEAP refers to the topic score when deciding which topic the student should study next and how they should study it. If the student should be doing exercises, LEAP refers to the exercise scores when deciding which exercise to give a student and in which mode it should be presented. If the student is studying an exercise in Focused Practice mode, LEAP refers to the SA pair scores when selecting SA pairs for the student to practice, review, skim, and scaffold. The student model records performance data for each SA pair that the trainee encounters, and uses the data to compute a score for the SA pair. The scores are aggregated to obtain a Proficiency measure for each topic and a Challenge measure for each exercise. SA pair scoring, conversation challenge and topic proficiency are discussed in sequence.

4.3.1 SA Pair Score

On the SA pair level, LEAP's student model is patterned after Newell & Rosenbloom's (1981) 'ubiquitous law of practice.' In particular, it is based on the observation that the probability of a correct response increases as a func-

---

3 How these issues are addressed is described in the next section.

4 The specific meanings of these terms will be defined in the next section.
tion of the number of previous correct responses (Underwood, as cited in Postman, 1972). LEAP’s student model goes beyond the law of practice in that it considers the weighted values of:

- **times seen**: the number of times the trainee has seen the SA pair performed by an expert
- **average score**: the trainee’s average score on each SA pair
- **consecutive (in)correct**: the strength of the trainee’s knowledge (or ignorance) based on the number of consecutive correct (or incorrect) responses for the SA pair

These values are calculated from student actions. As mentioned, exercises in LEAP are sequences of SA pairs. When a student encounters an SA pair in an exercise, the Student Model records several student actions regarding that SA pair. These observations are saved using counters as described in Table 4.7:
Table 4.7

SA Pair Score Counts

<table>
<thead>
<tr>
<th>Counter</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seen</td>
<td>Increment if the student sees the expert perform the SA pair.</td>
</tr>
<tr>
<td>Total</td>
<td>Increment if the student tries to take the action.</td>
</tr>
<tr>
<td>Correct</td>
<td>Increment if the student takes the correct action.</td>
</tr>
<tr>
<td>Consecutive (In)Correct</td>
<td>Increment if this action and the preceding action were correct.</td>
</tr>
</tbody>
</table>

From these counts LEAP calculates three values:

- The **Average value** is Correct divided by Total of all attempts at a particular SA pair. It provides a record of the student’s total performance. An average value weights all attempts equally.

- The **Seen value** is calculated by inserting the number of times the student has seen the SA pair performed into a function that increments the value by a slightly smaller amount each subsequent time the SA pair is seen. The rationale is that students can learn by watching an expert, but that each observation of the expert is a bit less instructive than the previous one.

- The **Consecutive (In)Correct value** is calculated by inserting the number of consecutive correct or consecutive incorrect attempts at the action into a function that increments the value by a slightly smaller amount for each subsequent attempt. In contrast to the
Average value, which looks at all attempts, Consecutive (In)Correct looks only at the most recent attempts, rewarding (penalizing) students for consistent performance. The rationale for the function is that students learn the most on their first few attempts and that subsequent practice, while important, eventually has diminishing returns.

The three values are weighted and summed to get an overall score for the SA pair. Current default weights are:

- 10% Seen
- 25% Average
- 65% Consecutive (In)Correct

These weights were chosen because they produce ‘reasonable’ values for correct/incorrect sequences of answers indicating:

- initial errors, then learning
- learning followed by occasional slips
- random guessing with no learning
- correct action already known to trainee; etc.

Although the value produced seems to be a reasonable estimate of how well the student will perform on the next occurrence of the SA pair, it is only an estimate. Since student modeling is a crude art (Hativa & Lesgold 1991; Katz, Lesgold, Eggan, & Gordin, 1993), LEAP never entirely ceases to present an SA pair for practice, no matter how high the student’s score. Instead, the probability of having to practice an SA pair is initially set to 100% and decreases as the trainee’s score on the SA pair increases. This point is discussed further in the next section.

Each SA Pair in the domain knowledge base has its predictive value attached. The values of SA Pairs are aggregated to provide ratings for each con-

5 In the section below on Authoring, one figure shows an authoring screen, where the weights of the values can be adjusted by moving the slider beside each one.
versation and each topic. LEAP uses these numbers, along with other information, to make several tutoring decisions, including SA pair presentation, exercise selection, and topic recommendation.

4.3.2 Conversation Challenge

As mentioned, each exercise in LEAP is a conversation, and conversations are sequences of SA pairs. Every SA pair occurs in more than one exercise, and some SA pairs occur in many exercises. By averaging the student's scores on the SA pairs in an exercise, LEAP can estimate the student’s proficiency on the exercise as well as the work remaining to master it -- the exercise’s current Challenge. Each exercise’s Challenge is recalculated whenever the student finishes any exercise and is displayed at the top-level. The current default exercise challenges are: a lot, some, a little, and none. The student model also records and displays how many times the student has practiced each exercise. LEAP uses both the Challenge and Times-practiced measures when selecting the next exercise for the student to practice. The Conversation Table of Contents with Challenge and Times-practiced measures is shown in Figure 4.4.
Using the student model as a basis for selecting exercises, and SA pair presentation within exercises, has two important effects: first, exercises can be designed to be realistic instead of being artificially simplified to be within the student's ability. Realistic exercises enable instructional designers to address critical issues of motivation (Lepper, Woolverton, Mumme, & Gurtner, 1993; Keller, 1983), transfer (Larkin, 1989), and apprenticeship learning (Suchman, 1987; Lave & Wenger, 1991). Second, LEAP's ability to select and modify exer-
cises based on an instructional strategy and the state of the student model helps optimize learning. In brief, LEAP takes unique advantage of the student model to provide realistic exercises and to focus instruction within each exercise.

4.3.3 Topic Proficiency

As mentioned, a topic is a conceptually related portion of the domain knowledge base, e.g., Provide Set-Up Information. The student model records an overall performance measure for each topic based on two factors: the percentage of SA pairs in the topic that the student has tried, and the student’s average score on those SA pairs. These values are combined and assigned a symbolic Proficiency label for each topic. The student’s proficiency is recalculated and displayed each time they return to the top-level and is used by LEAP when recommending the next topic for instruction. The current default settings are shown in Table 4.8
Table 4.8

Proficiency versus Percent Tried and Average Score

<table>
<thead>
<tr>
<th>Percent of Topic Tried</th>
<th>Average Score*</th>
<th>Proficiency Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>95</td>
<td>75</td>
<td>Excellent</td>
</tr>
<tr>
<td>65</td>
<td>40</td>
<td>Good</td>
</tr>
<tr>
<td>50</td>
<td>30</td>
<td>Almost!</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>Practicing...</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>Untried</td>
</tr>
</tbody>
</table>

Note. * Scores are lower than conventional grades.

For example, if a trainee has tried 55 percent of the SA pairs in a topic and the trainee's score is 33, then the trainee's Proficiency for the topic is "Almost!". If the score were 29 for the same percent of SA pairs, the trainee's Proficiency would be "Practicing...".

4.3.4 Related Work

In an experiment reported in Corbett & Anderson (1992), students practiced in the Lisp Tutor until it determined that they had acquired each rule. Posttest results did not support the hypothesis that students had acquired all rules. The student model design assumed that rules simply transition from unlearned to learned with no transition period and no forgetting. In contrast, in LEAP, realistic exercises mean that previously 'learned' SA pairs will be encountered in later exercises and LEAP never entirely ceases to present individual SA pairs for practice. If a student should err on or forget a previously
learned action, its consecutive (in)correct score will drop sharply, increasing the odds the SA pair will be practiced at the next opportunity.

Sherlock II is another coached practice environment (Katz, Lesgold, Eggan, & Gordin, 1993). In Sherlock a sequence of three errors followed by three correct responses on a rule will yield a distribution (Sherlock’s analog to LEAP’s score) that is the same as a sequence of three correct responses followed by three errors. In LEAP the two sequences yield scores that are quite different, and LEAP responds accordingly.

4.3.5 Student Model Summary

LEAP is a coached practice environment that optimizes the learning process by maintaining and consulting a detailed student model. LEAP calculates a score for every SA pair, exercise, and topic the student tries, then uses the scores both to provide feedback to the student and to select topics, exercises and SA pairs for the student to practice, as well as the methods for practicing them. The SA pair Score is a weighted sum of the Average, Seen, and Consecutive (In)Correct values. The conversation Challenge is a measure of the further learning required to master a conversation. The topic Proficiency is a measure of how much of and how well a topic has been learned. Challenge and Proficiency are recalculated at the end of each exercise and given a symbolic label that is displayed on LEAP’s top-level tables of contents.

Using the student model to focus practice means exercises can be realistic and complex instead of artificial and simple, thereby enabling instructional designers to address issues of motivation, transfer, and apprenticeship learning. The student model is temporally sensitive, enabling LEAP to adjust its instruction as trainees learn or forget. The next section describes how LEAP makes tutoring decisions, including the role of the student model in the decision-making process.
4.4 How LEAP Makes Tutoring Decisions

Most of us can recall having an instructor who was an expert in her field but who could not teach. In developing LEAP a conscientious effort has been made to include both domain expertise and tutoring expertise in the tutor. To provide LEAP with tutoring expertise, a body of instructional design theory has been applied to the training task. A side benefit of including tutoring expertise in LEAP is the generality of the result: Once defined, the tutoring expertise can teach similar domains (i.e., other service rep tasks). For LEAP to teach another domain requires only the insertion of the new domain expertise; the tutoring expertise is already there.

LEAP’s tutoring decisions consist of determining what the trainee should study next and how the trainee should study it. These decisions are based on the state of the student model and on characteristics of the material to be studied. LEAP makes its tutoring decisions on three levels (an earlier version of this section appears in Linton, Bell, Bloom, & Norton, 1994). LEAP recommends:

- a topic of study and a method for studying it
- an exercise (a conversation) to practice and a method of practicing it
- for each step within an exercise, LEAP determines whether the trainee or the system should perform it

In LEAP, trainees practice role plays with simulated customers and a simulated database, focusing on one topic at a time until they have mastered it. LEAP is a coached practice environment where trainees can master customer service skills before interacting with real customers. Joyce and Weil (1986 p. 372) attribute similar advantages to learning from simulations of all sorts. As mentioned in the Assumptions section, LEAP’s design assumes that a human instructor is available to: teach the use of LEAP, coach trainee’s
learning activities on LEAP, and act as subject matter expert (SME) when trainees have questions whose answers are not in LEAP.

Instruction in LEAP is similar to what Brown and Palincsar (1989) refer to as *proleptic teaching* teaching in informal apprenticeships, where teaching is a minor part of the total activity. "... novices are encouraged to participate in a group activity before they are able to perform unaided ... The novice carries out simple aspects of the task while observing and learning from an expert, who serves as a model for higher level involvement" (p. 410). LEAP is similar to proleptic teaching in that trainees respond to portions of the conversation before they are able to handle a whole conversation unaided. The novice carries out those aspects of the conversation she or he is learning (or reviewing), while LEAP's expert carries out the remainder. One difference between LEAP's practice environment and an informal apprenticeship is that LEAP's environment is simulated rather than genuine. While it might be a good learning experience to have a trainee and an expert rep jointly converse with a genuine customer, the customer would find it disconcerting, and the expense of human one-on-one tutoring makes it infeasible. However, with a simulated customer and a simulated expert, the trainee can, as in proleptic instruction, carry out some portions of the task, while observing and learning from an expert. Brown and Palincsar (1989) contrast proleptic instruction with traditional instruction as follows:

Consider tried and true educational procedures such as easy-to-hard sequences ... one way of making the task easier it to divide it into manageable subcomponents and to provide practice on these in isolation until they are perfected. This increases the likelihood that the easy tasks will not resemble the complex target, and it is often the case in educational settings that the role of recombining the subcomponents
(vertical transfer) or using them flexibly in tasks of which they are elements (lateral transfer) is left up to the student with disastrous results.

In proleptic teaching, by contrast, the integrity of the target task is maintained; components are handled in the context of the entire task; skills are practiced in context. ... the novice's role is made easier by the provision of expert scaffolding ... until the novice can take over more and more of the responsibility. The task, though remains the same, the goal the same, the desired outcome the same. There is little room for confusion about the point of the activity, thus finessing to some extent problems of metacognition and transfer. (p. 415-416)

LEAP's tutoring decisions consist of determining what the trainee should study next and how the trainee should study it. These decisions are based on characteristics of both the study material and the trainee. Study material is divided into topics, with presentations and exercises for each topic. LEAP recommends specific topics, presentations and exercises. LEAP also contains a variety of learning methods for studying the material in the exercises. LEAP recommends the most appropriate learning method for the trainee at any given moment (Collins, Brown, & Newman, 1989; Reigeluth, 1983). In Figure 4.5, LEAP is recommending a topic and a study method. The recommendation mechanisms are described in detail below.
4.4.1 Varying the Degree of Active Tutoring

LEAP takes a cue from Rogers (1982) and respects trainees’ capacity to direct their own educational decisions. In general, LEAP makes tutoring decisions but does not force them on the trainee, instead they are put forth as recommendations. In Figure 4.5, the trainee has pressed the Recommend button and LEAP is making a recommendation. Reigeluth & Stein (1983) hypothesize: “Instruction generally increases in effectiveness, efficiency, and appeal to
the extent that it permits informed learner control by motivated learners." Merrill (1979) elaborates the following learner control factors:

- Content control: Selection and sequencing of the course content
- Pace control: The rate at which the trainee will learn
- Display control: The instructional strategy components the trainee selects and the order in which they are used. (In LEAP, these are study methods.)
- Conscious cognition control: the particular cognitive strategies the learner employs when interacting with the instruction

In LEAP, the learner has control over each of these factors. LEAP serves as a resource and guide while responsibility and control for choosing what and how to study remain in the hands of the trainee. Permitting trainees to make their own instructional choices of both what to study and how to study in LEAP is one means of encouraging exploration (Collins, Brown, & Newman, 1989).

Trainees can elect the degree of active tutoring they receive from LEAP, from no tutoring to full tutoring. At the no tutoring end of the scale, trainees make all the instructional decisions. They are in charge of their learning process. They select the Conversations or Topics they wish to practice and the methods they wish to practice them by. LEAP provides SA pair feedback, but acts mostly as a bookkeeper, updating the Proficiency and Challenge scores.

At the full tutoring end of the scale, LEAP makes all the instructional decisions, freeing the trainee to concentrate on learning the subject matter itself. LEAP selects the topic for the trainee to study and determines when the topic has been mastered and it is time to move on. LEAP selects the Conversations most suitable for the current topic and sequences the trainee through them, choosing fresh conversations or revisiting previously seen ones as necessary. LEAP selects the method by which the trainee studies each
conversation, increasing the trainee's involvement in the conversation until the topic is mastered. Within each conversation LEAP decides when to skim and when to scaffold each SA pair for learning to take place at an optimal rate.

Most trainees will probably elect some intermediate degree of active tutoring. They can accept or reject any tutoring recommendation LEAP makes, change the conversation study methods at any time, and exit any exercise at any time. While LEAP is capable of making all the tutoring decisions for a trainee, permitting a trainee to make his or her own tutoring decisions gives the trainee control and reinforces the point that ultimately, responsibility for one's learning lies with oneself.6

4.4.2 How LEAP Selects Topics

LEAP recommends both a topic to study and a method for studying it. Figure 4.6 displays the Topics Table of Contents. Topic selection is based on a consideration of three factors: the topic sequence in the Table of Contents7, the topic last studied, and the trainee's proficiency on each topic. As mentioned in the section on student modeling, topic proficiency ratings are: untried, practicing..., almost!, good, and excellent. These ratings are determined by the percentage of SA pairs in the topic the trainee has tried and his or her

6 At the same time, there is evidence that learners do not necessarily make optimal instructional decisions (Tobias, 1987; Merrill, 1983). A second domain of instruction that most tutors should be capable of engaging in is how to learn, including the self-monitoring of learning progress, selection of topics for learning, etc.

7 As in textbooks, some topics are prerequisite to others, the instructional designer must place them earlier in the sequence. Sequencing topics by considering their prerequisite relationships was first suggested by Gagne (Gagne & Briggs 1979), and is echoed by many instructional theorists.

As it turned out, in the Voice Messaging domain, no topics were truly prerequisite to others. The sequencing of topics in LEAP is first by frequency of occurrence (Add Voice Messaging, Change Ring Cycle, Remove Voice Messaging), and then by sequence of occurrence of topics within conversations.
average score on those SA pairs. Rating names are created and their values assigned by the domain (content) author.

Figure 4.6

Top Level Screen: Topics Table of Contents

LEAP's topic selection heuristic is based on the goal of having the trainee first attain minimum proficiency in all topics (to become productive as soon as possible) and later acquire expertise in all topics (for improved effectiveness). The overall topic selection heuristic is:

- begin at the first topic and practice it to a predetermined proficiency rating (currently set at 'good'). In the process, some other topics will have been encountered and studied to some extent
• For the next topic, pick the one which is nearest to a 'good' proficiency rating and study it

• Continue in this way until all topics have a proficiency rating of 'good'

• After the trainee has learned all topics to 'good,' revisit each topic in the same manner, practicing until the trainee reaches the 'excellent' rating

• If the trainee overrides LEAP’s recommendation regarding the topic to study, stay with the trainee-selected topic until the trainee reaches the target rating, chooses another topic, or asks for a recommendation

To summarize: If one topic precedes (or is prerequisite to) another, LEAP will present it first. Otherwise, LEAP selects the topic that the student has most nearly completed, and sticks with it until it is complete. LEAP has a spiral curriculum, meaning that topics are ‘completed’ twice. The first time through the curriculum, topics are considered complete when the student model reaches a proficiency of Good; at this point students know the basics well enough to begin working. The second time through the curriculum, topics are considered complete when the student model reaches Excellent; at this point students have mastered the topic.

4.4.3 How LEAP Selects a Study Method

After selecting the topic, the tutor selects the method of studying the topic: There are three top-level study methods: Study the Guide, Examine Contact Flow, and Rehearse Conversation. Figures 4.4 and 4.6 display the study method buttons.

---

8 Numerous authors, e.g., Gropper (1983), observe that tasks need not be learned in the sequence they are performed.

9 Bruner (1960), suggests a spiral curriculum; where topics are systematically reintroduced for review and elaboration.
The heuristic for selecting a study method is straightforward and mirrors conventional classroom and textbook instruction. Apart from the near-universal appearance of this method in textbooks, theoretical support for it is found in Gagne and Briggs (1979), and in other instructional design texts (Reigeluth, 1983; Joyce & Weil, 1986). In brief, LEAP recommends that trainees first observe a demonstration of the topic in use, then study knowledge related to the topic in multi-media presentations, then practice applying the knowledge in simulated conversations. Trainees may also elect to explore the structure of a topic or of the entire domain knowledge base. Each of the three top-level study methods has one or more sub-methods; these methods and their sub-methods will now be described in more detail.

4.4.3.1 Study the Guide

When trainees Study the Guide, LEAP presents multi-media displays of basic information about the topic: text, speech, graphics, photographs, animations and videos. Figures 4.7 and 4.8 illustrate the Guide. Currently multi-media presentations are created using a number of tools, saved in various formats, and presented using VAPS© (virtual applications). In selecting materials for the Guide, care was taken to omit materials already in the trainees’ Desk Reference, so that referring to their Desk Reference would be a skill trainees would have to practice while using LEAP.
After trainees exit the Guide, their student model is updated with the annotation that they have seen the Guide for that particular topic. No tutoring decisions are made within the Guide. For further details about trainee activities in the Guide see the appendix: *LEAP Walkthrough.*
In this chapter, you'll learn about our Voice Messaging Service and how it works. By understanding your customer's needs, and then telling them about benefits of Voice Messaging that meet those needs, you'll be cultivating loyal customers for US WEST—and helping to grow our business.

Voice Messaging answers the phone when the customer can't. Voice Messaging Service (VMS) is a service comprised of the Basic Elements (BSE) Call Forward, Busy Line, Don't Answer, and the Message Waiting Indication. Calls to the customer's line are transferred to a multi-line hunt group by adding a call forwarding number to the order. The information is transmitted over a private line between the central office switch and US WEST Voice Messaging equipment. Messages are stored in a voice mailbox.

Figure 4.8

Study the Guide: typical page

4.4.3.2 Explore Domain Knowledge Base

Explore and Practice modes are two fundamentally different ways of learning. In Explore mode, trainees observe and try (recognizing & selecting the text of) each SA pair as an abstraction in the context of its conceptually related materials. In Practice mode, trainees observe and try (recalling/generating, speaking, or keyboarding) each SA pair as an instance in the context of its application in a conversation. Practice mode will be discussed in detail later; here we focus on Explore mode.
Take, for example, the topic Negotiate Due Date; due dates vary by service; in Explore mode trainees directly compare and contrast the relationship between services and due dates as shown in Table 4.9:

Table 4.9
VMS Service and Due Date

<table>
<thead>
<tr>
<th>Service:</th>
<th>Due Date:</th>
</tr>
</thead>
<tbody>
<tr>
<td>add-voice-messaging:</td>
<td>four-working-day</td>
</tr>
<tr>
<td>change-ring-cycle:</td>
<td>two-working-day</td>
</tr>
<tr>
<td>remove-voice-messaging:</td>
<td>same-day</td>
</tr>
</tbody>
</table>

One of the SA pairs in this topic is:

**IF** context is: change-ring-cycle  
**AND** situation is: negotiate-due-date  
**THEN** action is: negotiate a two-working-day due date

In Explore mode, trainees can see SA pairs stated directly; the content and structure underlying the conversation space is revealed for study.

Each SA pair has several instances, which the trainee can hear in the History window of the Explore mode:

"You can have that changed on Friday or any time after that, when would you like it?"

"I can have your ring cycle changed for you by Tuesday, will that be OK?"
"We can have that done for you in two working days, that would be Saturday, OK?"

The trainee can explore topics, or Examine Contact Flow\(^1\), in three modes: Observe, Focused Practice, or Full Practice. In all modes, trainees explore the domain knowledge base by traversing it, choosing among multiple situations, and moving forward or backward through the domain knowledge base at will. In explore mode (Examine Contact Flow), whenever a conversation could branch, all branches are displayed and trainees pick the branch they want to explore. After traversing one branch, the trainee can back up through the conversation to the branching point and pick another branch to traverse, etc. For example, if the trainee chose to explore the topic Determine Class of Service, (described in the Domain Representation section) these three situations would appear on the screen simultaneously:

- S: Customer has a private line
- S: Customer has measured service
- S: Customer has a party line

Whichever one the trainee chooses, its action will then appear on the screen. If the customer has a private line, the topic ends at this point (the conversation continues onto the next topic). Customers with measured service, however, now have three options:

- S: Customer decides to upgrade class of service
- S: Customer decides not to add Voice Messaging
- S: Customer understands costs

Again, these all appear on the screen, and the exploration process can continue. Figures 4.9, 4.10, and 4.11 illustrate Examine Contact Flow. In

\(^1\) Reps refer to both the customer and the conversation with the customer as a contact.
Observe mode, trainees see the expert response to each situation they select. The two Practice modes are a little more adventurous; in general when Practicing, trainees attempt to respond to each situation themselves before seeing expert responses (the differences between the two Practice modes are described in detail below).

![Figure 4.9](image)

Examine Contact Flow: Observe
Once trainees have explored a branch, and put situation and action lines in the History window, they can click on the lines in the History and hear an expert's recorded voice speaking the line (similar to Observe, described below). Repeated clicking on a line in the History cycles through the various speaker's recordings of it as it is instantiated in different conversations and stored in the domain knowledge base. This ability to hear the same SA pair applied by several experts in the context of different conversations provides an example of how a line that appears to be unique when it occurs in a conversation has an underlying abstraction (the SA pair) that when learned can be generalized and the knowledge transferred to many conversations.
Figure 4.10

Examine Contact Flow: Full Practice

Exploration in LEAP is micro-level -- trainees explore the domain knowledge base -- rather than the macro-level exploration envisioned by proponents of exploration as a learning method (Perkins, 1986; Rogers, 1982; Collins, Brown, & Newman, 1989). Nevertheless, Explore mode (Examine Contact Flow) in LEAP permits trainees to select topics for examination, pursue the what-ifs of varying trainee situations, and acquire a comprehensive,
integrated view of a topic, independent of the context of a specific conversation.

Figure 4.11
Examine Contact Flow: feedback screen

4.4.3.3 Rehearse Conversation

LEAP’s Conversations correspond to the exercises in conventional textbook instruction. Rehearse Conversation is the most complex part of LEAP’s tutoring process. The complexity arises from the instructional design goal of having trainees work at the edge of their competence in the context of whole conversations. Working at the edge of one’s competence means practicing
mainly the topic that is the current focus of instruction while not redoing those topics one already knows how to do, nor doing those topics one is not yet ready to do, even though these topics appear naturally during the course of a conversation.

Instructional designers from behaviorist through cognitivist suggest top-down approach to instruction, putting more general before more specific objectives, (Gagne & Briggs, 1979), global before local skills, (Collins, Brown, & Newman, 1989). LEAP instantiates this approach by showing the global situation (the whole conversation) without requiring the trainee to perform the whole conversation.

After LEAP has selected the Rehearse Conversation study method for a topic, it must then select a particular conversation to study, and then select a study method for that conversation; these two selection processes are described next.

4.4.4 How LEAP Selects Conversations

When LEAP has selected the Rehearse Conversation study method for a topic, LEAP then selects a conversation to study. As with all instructional decisions, the objective is to optimize learning. LEAP selects a conversation by ranking each of the conversations according to several weighted factors and selecting the top-ranked one. Two of the six factors LEAP considers in ranking conversations are characteristics of the conversation, two factors are assigned by the author, and two are related to the current state of the student model; the resulting sequence of exercises is unique for each trainee. Factors pertaining to the conversation are its relation to the current topic and its complexity. Factors assigned by the author are the conversation's overall importance ranking and the increment of new material from exercise to exercise. Factors related to the current state of the student model are the number of times they
have already practiced the conversation, and their current skill level. These factors are weighted and summed for each conversation and the top-ranked conversation is selected.

Rationales for the above-mentioned factors are as follows: First factor: the conversation’s relation to the current topic; of course conversations are designed to address particular topics, but since conversations are complete and realistic, they contain SA pairs pertaining to many topics. For every SA pair, LEAP keeps a complete cross-reference of the topic it pertains to and the conversations it appears in. LEAP uses this information to find all conversations with one or more SA pairs pertaining to a topic. Second factor: the conversation’s complexity; the number of SA pairs in the conversation that pertain to the topic is used as a crude measure of the conversation’s complexity (doing ten things right is harder than doing five things right). The complexity measure is combined with the trainee’s current skill, described below, to get a Challenge rating for the conversation.

Third factor: overall importance ranking; this factor is assigned by the author. This factor permits any arbitrary ordering of conversations the author might choose. For example, the author might rank conversations depicting frequently occurring situations above those with equally important but rarely occurring situations. Or perhaps the author might rank conversations depicting rare critical situations above frequent unimportant situations. Factors that authors are found to apply consistently here can be made explicit in later versions of LEAP. Fourth factor: the increment of new material from exercise to exercise. This factor not only determines how quickly material is covered, but more importantly, flavors the trainee’s experience with LEAP. If too few new SA pairs are presented per exercise, the experience will be boring and repetitious; if too many new SA pairs are presented and trainees are overly chal-
lenged, they will become frustrated. The author must assign these two factors based on knowledge of the students and of their learning goals.

The last two factors are related to the current state of the student model. Fifth factor: the number of times the student has already practiced the conversation; for variety LEAP selects new (or less-frequently practiced) conversations in preference to repeating those previously practiced. Sixth factor: the student’s current skill level; the student’s scores for all the SA pairs that are both in the conversation and in the topic are averaged to get a current Challenge rating for each conversation.

In the case that the trainee prefers to select a conversation him- or herself, he or she can bring up the conversation-level Table of Contents shown in Figure 4.4. The conversation-level Table of Contents includes the conversation’s name and type, key words characterizing the conversation, the number of times the trainee has seen the conversation, and its current challenge or difficulty for the trainee.

4.4.5 Conversation Study Methods

After selecting a conversation, LEAP must select a study method for the conversation: There are three ways to study an individual conversation: Sequenced by increasing trainee involvement, they are: Observe, Focused Practice and Full Practice. In Observe, the tutor does all actions, the trainee observes the situations and the expert rep’s actions. In Focused Practice, LEAP selects some SA pairs for the trainee to do and does the remainder for the trainee. In Full Practice, the trainee practices all the rep’s actions -- LEAP provides step by step feedback and updates the student model. Each of these is described in more detail below. If the trainee has not yet observed any conversation in the topic, then LEAP recommends ‘Observe,’ otherwise LEAP recom-
mends 'Focused Practice,' which is the default. Trainees may also elect Full Practice. Figures 4.12 through 4.19 illustrate conversation study methods.

4.4.5.1 Observe

When the trainee elects Observe, LEAP demonstrates both sides of the entire conversation, presenting the customer’s verbal actions and the expert rep’s verbal actions, cognitive actions and database I/O actions. The trainee observes the expert at work and builds a conceptual model of the task. As the student observes the expert rep perform each step, the student model increments the Seen count for the corresponding SA pair. Rosenshine, (as cited in Joyce & Weil, 1986, p. 328), Collins, (Collins, Brown, & Newman, 1989) and Gagne, (Gagne & Briggs, 1979) call for the instructor to demonstrate the desired task performance to the trainee. While the instructor demonstrates, the trainee observes. Figure 4.12 illustrates Observe mode.
4.4.5.2 Full Practice

As mentioned, trainees have the option to elect Full Practice, but LEAP does not recommend it. The reason is that each conversation contains a lot of material, and it is likely that the trainee will encounter much material that is already well-known and does not need practicing. It is also likely that trainees will encounter much, probably too much, new material, more than they should begin practicing in any one exercise. For these reasons, LEAP recom-
mends Focused Practice, which is described in the next section. In both Full Practice and Focused Practice modes, trainees practice on individual SA pairs. This section will describe practicing with LEAP on the SA pair level, including feedback for each kind of SA pair, and hints.

When the trainee elects Full Practice, LEAP presents a conversation step by step. Each step is an SA pair. During the conversation LEAP presents a situation, then observes and evaluates the action the trainee takes. If the action is correct, the conversation continues. If the action is incorrect, LEAP provides some informative feedback\(^1\) and the student tries again. LEAP updates the SA pair score based on the student’s first response. Figures 4.13 and 4.14 illustrate Full Practice Mode.

\(^1\) For database errors, LEAP describes the expected field and value. For verbal errors, LEAP only indicates the user has erred.
Observe • Focused Practice • Full Practice
Click to record your response to the contact:
Start recording response

Observe • Focused Practice • Full Practice
Click to record your response to the contact:
Stop recording response
Recording...

Click one of the possible responses:
- Respond with willingness to help.
- Discuss 1 free ring cycle change with customer.
- Verify all details of order.
- Offer to inform caller when Voice Messaging becomes available.

Figure 4.13
Rehearse Conversation: response recording process
Figure 4.14

Rehearse Conversation with History window open
LEAP presents three main types of situation to the trainee: the first is a customer statement or question, the second is a database I/O requirement, the third is a contact standard or legal requirement. The trainee must respond to each situation with one of three main types of action: a verbal statement or question, a database command or input, or an observation or conclusion. There is no correlation between situation type and action type. Here are some example Situation-Action pairs showing each type of action:

**Situation:**
Customer: “I’d like to add Voice Messaging to my service.”
and
Database Screen01: Name ______ Number ( ) ______

**Action:**
Rep replies: “I can help you with that; may I have your name and number with area code please?”

**Situation:**
Customer: “My name is John Smith and my number is 303 541-6412.”

**Action:**
Rep enters: John (303) 541-6412

**Situation:**
Database: Services: ..., voice messaging, ...

**Action:**
Rep concludes: Voice Messaging is available in the customer’s area so it is OK to provide the service he has requested.

4.4.5.2.1 Feedback

After each action the trainee makes, LEAP provides feedback on the SA pair. SA pair level feedback in LEAP is quite distinct from conversation level feedback, which occurs after the conversation has been completed. Conversation feedback is described later. Usually both types of feedback are simply referred to as ‘feedback,’ since the type of feedback is clear from the context.
Many instructional design theories espouse the value of feedback, for example, Aronson and Briggs (1983) state: "Providing feedback is a crucial instructional event" (p. 92). Merrill (1983) states: "Feedback should always accompany practice at every performance level" (p. 322). And Keller (1983) states: "To maintain intrinsic satisfaction with instruction, use verbal praise and informative feedback rather than threats, surveillance, or external performance evaluation." (p. 426). As important as providing feedback is to these theorists, surely none envisaged the possibility of providing feedback as precise as the SA pair feedback LEAP provides its individual trainees.

LEAP provides different feedback for database actions and verbal actions. After each database command or entry the trainee makes, LEAP provides SA pair feedback. In general, if the trainee's database action is correct, the conversation simply continues. If the database action is incorrect, LEAP provides detailed feedback (actual versus expected field, value, command, etc.). The trainee may then retry the action or ask to see it demonstrated. As Collins, Brown, & Newman (1989) note, trainees reflect whenever they respond incorrectly.

When the trainee performs an action in the database, LEAP can directly observe, evaluate, and respond to the trainee's action. In contrast, when the trainee is conversing with a simulated customer, LEAP cannot understand the trainee's action. Figure 4.13 illustrates how LEAP handles conversational actions. Natural language understanding, especially the sort of speech understanding needed to interpret service representative trainee responses, is simply not on the horizon; nor is it realistic for trainees to type responses they would normally speak. Thus, LEAP has the trainee first record his or her action (for the purpose of creating, articulating, practicing and later, reviewing it), then presents the trainee with a list of plausible actions, from which the
trainee selects the nearest equivalent to his or her own. The distractors in the response set are selected from a list based on misconceptions, missing conceptions, near misses, blunders, and other empirically determined wrong answers trainees often make. This pragmatic design compromise enables the tutor to observe, evaluate, and respond to the trainee’s ‘spoken’ responses without using speech recognition or natural language processing.

After each verbal response the trainee makes, LEAP provides feedback. If the trainee’s response is correct, LEAP echoes the response for reinforcement and proceeds to the next situation. If the selection is incorrect, LEAP informs the trainee (verbally: “Sorry, wrong answer.”) and gives him or her another chance. Because the list of alternatives is short (at most four), trainees will quickly find the expected answer, and since they have made an error, will listen closely to the expert’s response. Feedback for an observation or conclusion is like that for verbal responses, except that correct responses are echoed in text rather than spoken.

4.4.5.2.2 Hints

When trainees are trying to respond, they may not know what to do. In that case they can ask for a Hint. Figures 4.15 through 4.18 illustrate Hints. Hints are part of coaching (Collins, Brown, & Newman, 1989), and learning guidance, (Gagne & Briggs, 1979). Behaviorists refer to hints as cues (Gropper, 1983). In LEAP, hints are multi-leveled. For database actions the first level hint (Figure 4.15) indicates the field from which data is read or into which data is input. The second level hint (Figure 4.16) for database actions adds to that the data itself, and the third level hint demonstrates the task to the

---

2 Actually, while the distractors should be selected from such a list, the problem is that the list must be empirically acquired. At the time of LEAP’s field trial, the distractors were either input by the instructional designer or selected by LEAP from other actions in the domain knowledge base; selection was based on the similarity of the actions to the correct action.
trainee. For conversational actions, the first level hint (Figure 4.17) describes the context and the situation. The second level hint for conversational actions (Figure 4.18) describes the appropriate response to the situation. The third level hint is a spoken example of the expert’s response.

Figure 4.15
SONAR-related level 1 hints

Figure 4.16
SONAR-related level 2 hints
The current context is: Verify that VMS is compatible with existing line features.

The situation is: SONAR CSR CALL WAITING, CONTROLLABLE

Figure 4.17

Conversation-related level 1 hint
4.4.5.3 Focused Practice

In Focused Practice, LEAP optimizes learning by modifying the selected conversation to focus the trainee’s learning efforts. From the trainee’s perspective, Focused Practice is simple: LEAP selects some of a conversation’s SA pairs for him or her to do -- thereby focusing the practice -- and does the rest of the SA pairs itself. From the tutoring decision-making perspective, however, Focused Practice is more complex.

When LEAP (or the trainee) has selected a conversation and determined that the most appropriate way to practice it is Focused Practice, LEAP
modifies the selected conversation, creating an exercise\(^3\) that is unique to the trainee’s current state and, presumably, optimal for learning. The three factors LEAP considers when modifying the conversation are the current topic, the state of the student model, and the element of chance (see the section \textit{Randomness}, below). LEAP modifies the conversation so that trainees \textit{practice} material they are currently attempting to learn and \textit{reviews} material they have already learned, while LEAP either \textit{skims} or \textit{scaffolds} the remaining material (these four terms are described in detail below).

Focused Practice is LEAP’s most complex study method. The purpose of Focused Practice is to provide what is variously described as situated learning (Lave & Wenger, 1991), proleptic instruction (Brown & Palincsar, 1989), or cognitive apprenticeship (Collins, Brown, & Newman, 1989). These concepts have in common a genuine task environment; an expert who performs the task and who models, coaches, scaffolds, fades, and evaluates the task as necessary; and novices who learn the task and gradually take it over as they become capable of doing so.

The need for Focused Practice arises from two design decisions: The first decision is to have the trainee practice in the context of whole tasks. Practice in the context of whole tasks means that each SA pair is \textit{situated} in a realistic context, not being studied in isolation, thus the learning is ‘situated learning’. It further means the trainee is \textit{applying} the SA pair, not reading or repeating a description of it. Finally, it means the trainee is applying an \textit{instance} of the SA pair in a specific situation, not verbalizing some general rule

\(^3\) Until this point, the terms ‘conversation’ and ‘exercise’ have been used as synonyms and were in fact nearly, if not totally, synonymous. In this section it may be helpful to consider conversations as the raw material from which exercises are constructed.
for what to do when certain circumstances occur. As Lave & Wenger (1991) state:

Knowing a general rule by itself in no way assures that any generality it may carry is enabled in the specific circumstances in which it is relevant. In this sense, any “power of abstraction” is thoroughly situated, in the lives of persons and in the culture that makes it possible. On the other hand, the world carries its own structure so that specificity always implies generality (and in this sense generality is not to be assimilated to abstractness): That is why stories can be so powerful in conveying ideas, often more so than an articulation of the idea itself. (p. 34)

In this case the whole task context is a conversation, and in realistic conversations almost anything could occur, including situations with which the trainee is already well-practiced, situations the trainee is practicing, and situations the trainee is unprepared to deal with (i.e., situations the trainee has studied, is studying, and has not yet studied, respectively). The second design decision is to have trainees work at the edge of their competence, neither wasting time re-doing tasks already-known, nor floundering with (and foundering on) tasks which they are not yet prepared to deal with. In sum, Focused Practice is the method LEAP employs to have trainees work at the edge of their competence practicing specific skills in the context of whole tasks.

Table 4.10 lists the instructional activities that occur in Focused Practice, and who performs their actions. There are four instructional activities in Focused Practice: practice, review, skim, and scaffold. The four instructional activities have only two functional distinctions: either the trainee per-

---

4 Learning a general rule also has its value. The general rule - the SA pair - can be studied in Examine Contact Flow.
forms the action or LEAP’s expert does, but each of the four activities has a distinct instructional purpose, described next.

Table 4.10
Focused Practice Instructional Activities

<table>
<thead>
<tr>
<th>Instructional Activity:</th>
<th>SA Pair’s Action Performed by:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practice</td>
<td>Trainee</td>
</tr>
<tr>
<td>Review</td>
<td>Trainee</td>
</tr>
<tr>
<td>Skim</td>
<td>LEAP’s Expert</td>
</tr>
<tr>
<td>Scaffold</td>
<td>LEAP’s Expert</td>
</tr>
</tbody>
</table>

4.4.5.3.1 Practice

Functionally, practicing an SA pair in Focused Practice is exactly the same as practicing an SA pair in Full Practice. The difference is not on the SA pair level but on the exercise level. In Focused Practice, trainees focus on the SA pairs in one topic at a time, practicing them until the topic is completed. As a result, trainees practice a few new SA pairs of a conversation in any given exercise. The benefit of focusing on one topic at a time is that practice is concentrated. Related SA pairs, in one topic, are practiced frequently and repeatedly until they are learned. Anderson (1993) establishes that productions (and SA pairs are similar to productions) are learned in about five repetitions, and LEAP requires the user to try about five repetitions of an SA pair before assigning a score of Expert and switching to Skim.
4.4.5.3.2 Review

Functionally, reviewing an SA pair in Focused Practice is exactly the same as practicing an SA pair in Full Practice. The difference is on the exercise level. In Focused Practice, trainees review SA pairs they have already learned in topics they are no longer focusing on. Both instructional designers (e.g., Gagne, Briggs, & Wager (1992); West, Farmer, & Wolff, (1991)) and cognitive psychologists (Anderson, 1990) note the importance of review in learning and retention.

4.4.5.3.3 Skim

Functionally, skim works exactly like observe -- LEAP presents the situation and performs the action -- but instructionally the purpose is quite different. A trainee observes in order to build a mental model of a new skill. LEAP skims when the trainee already knows how to perform the skill and has no need to practice or review it. The purposes of skimming are first to improve training effectiveness by establishing a realistic context for the practice and review taking place, and second to improve training efficiency by speeding trainees through SA pairs they already know, so they can dedicate their practice time (a limited resource) to those SA pairs they need to practice, in the context of a complete conversation. While many instructional design theories espouse scaffolding, to my knowledge, having the tutor modify exercises by skimming material already known to the trainee is an original idea (made possible by ITS technology).

4.4.5.3.4 Scaffold

Functionally, scaffold works exactly like observe -- LEAP presents the situation and performs the action -- while the trainee observes in order to build a mental model of a new skill. In Focused Practice, the trainee also observes and builds a mental model of the new skill, and scaffold serves other
purposes as well. LEAP scaffolds when the trainee does not know how to perform the skill and should not be concentrating on it at this time (because it is not the current topic). The purposes of scaffolding are first to improve training effectiveness by establishing a realistic context where trainees can practice the tasks they are learning within a conversation they would otherwise be unable to complete on their own, and second to improve training efficiency by saving trainees time and ego: not requiring them to practice something they are not yet ready to learn. Scaffolding is suggested by Brown & Palincsar (1989), and Collins, Brown, & Newman (1989).

4.4.5.3.5 Current Topic

As mentioned, in Focused Practice, trainees focus on the SA pairs in one topic at a time, practicing them until the topic is completed. The topic the trainee is currently studying is the current topic. In Focused Practice, LEAP treats the SA pairs in the current topic in one fashion, and those not in the current topic in another. In the current topic, LEAP will either have trainees practice SA pairs they do not know well or skim SA pairs they do know well.

As for SA pairs not in the current topic, if trainees have never studied the topic, LEAP will scaffold the SA pairs, doing them for the trainee. If trainees have studied the topic, LEAP will have the trainees review the SA pairs they do not know well and again, skim the SA pairs they do know well.

Besides the two factors already mentioned that determine whether the trainee will practice, review, or skim an SA pair: whether the SA pair is in the current topic, and how well-known the SA pair is, there is a third factor that affects the decision: the element of chance or randomness.

4.4.5.3.6 Randomness

The two decisions: whether trainees should practice or skim material in the current topic and whether they should review or skim material in the
previously studied topics, are not made deterministically, but randomly. In general in Focused Practice, trainees will practice or review SA pairs they do not know and skim SA pairs they do know, but LEAP makes the decision randomly for each SA pair: the odds of the trainee’s having to practice or review an SA pair decrease as the trainee’s score on the SA pair increases.

The element of randomness ensures that for each SA pair, practice tapers off gradually and review, though increasingly rare, is perpetual. The element of randomness also ensures that a Focused Practice conversation is different each time a trainee practices it. There are three pedagogical justifications for these uses of randomness. First, there is no theoretical way of determining exactly how many times a trainee must practice something in order to learn it (Anderson, Boyle, Corbett, & Lewis, 1990). Thus there is no way to determine when to stop practicing. Second, there is no theoretical way to determine how the passage of time and the performance of other tasks (related and unrelated) influence the retention and use of the knowledge. Thus it is necessary to practice or rehearse the material over time in order to ensure the retention and correct usage of the knowledge (Joyce & Weil, 1986). Third, a certain element of unpredictability or surprise contributes to maintaining the trainee’s attention (Lesgold, Eggan, Katz, and Rao, 1992)

4.4.5.3.7 Compared with Lisp Tutor’s Practice

LEAP’s Focused Practice stands in strong contrast to, for example, the Lisp Tutor (Anderson, & Pelletier, 1991). In the Lisp Tutor, programming knowledge is taught in sections, each containing about a half-dozen rules. Students practice exercises in a section until all its rules are mastered. Students must code the entire exercise; there is no avoiding rules already mastered. There is no focus on the hard-to-learn rules, nor systematic review of previously learned rules. There is apparently no assurance that a given un-
learned rule will even appear in a selected exercise. In contrast, LEAP’s exercise selection heuristic finds exercises with an appropriate challenge. Within an exercise, LEAP’s students practice only on the SA pairs they need to learn or review.

4.4.5.3.8 Focused Practice, Summary

To summarize, Focused Practice is an activity where the trainee performs some SA pairs and LEAP performs the rest. The purpose of Focused Practice is to provide what is variously described as situated learning (Lave & Wenger, 1991), proleptic instruction (Brown & Palincsar, 1989), or cognitive apprenticeship (Collins, Brown, & Newman, 1989). These concepts have in common a genuine task environment; an expert who performs the task and who models, coaches, scaffolds, fades, and evaluates the task as necessary; and a novice who learns the task and gradually takes it over as she becomes capable of doing so. LEAP creates a Focused Practice exercise from a conversation by determining whether each SA pair should be practiced, reviewed, skimmed, or scaffolded. LEAP makes the determination based on the trainee’s score for the SA pair, on whether the SA pair is in the current topic, and on an element of randomness. The benefits of Focused Practice are efficient, effective learning and transfer. Table 4.11 shows the instructional activity as a function of when the topic was studied and the student’s score on the SA pair.
Table 4.11
Instructional Activity versus Topic Status and Student Score

<table>
<thead>
<tr>
<th>SA Pair Score:</th>
<th>Topic Studied:</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Previously</td>
<td>Currently</td>
<td>Never</td>
<td></td>
</tr>
<tr>
<td>Low or Medium</td>
<td>Review&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Practice&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Scaffold</td>
<td></td>
</tr>
<tr>
<td>High</td>
<td>Skim&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Skim&lt;sup&gt;a&lt;/sup&gt;</td>
<td>Not applicable</td>
<td></td>
</tr>
</tbody>
</table>

Note. <sup>a</sup> Gradually switch from mostly practicing or reviewing to mostly skimming as trainee’s score increases on SA pair.

4.4.5.4 Conversation Review

When the trainee reaches the end of a conversation in Observe, Full Practice, or Focused Practice, LEAP automatically brings up the Conversation History and gives the trainee the options of seeing a video Summary of the exercise and of Repeating the exercise. Figure 4.19 displays the Conversation History, the Summary, and Repeat Conversation buttons. The end of the exercise is an appropriate moment for feedback and reflection, and LEAP has several opportunities for these learning activities.

The Conversation History shows the trainee a script of the entire conversation. The trainee can select any response and click to hear the expert version as well as his or her own recording of the response. Comparing one’s own responses to the expert’s provides a powerful form of feedback. Trainees can hear the differences in the two responses, and compare not only the content of the response, but also the quality of the response. After listening to the responses, reflecting (Collins, Brown, & Newman, 1989) on the similarities
and differences between themselves and the expert, and determining how to improve, trainees may choose to repeat the conversation.

Reigeluth & Stein, (1983) and Reigeluth & Curtis, (1987) characterize summarizers as systematic reviews of what has been learned that help prevent forgetting. In the video Summary an expert rep reviews the salient features of the conversation and the rationale for the expert rep’s responses. The description and rationale provide a context or a framework for the detailed SA pair responses the trainee has made (or observed) during the conversation.

If, at the end of an exercise, the trainee returns to the Topics or Conversations tables of contents, she will see any changes in her topic Proficiency rating and the conversation Challenge rating.

To summarize, at the end of a conversation or exercise, trainees receive feedback at the SA pair level, the conversation level, and the topic level. Trainees also have the opportunity to review and reflect on the SA pair level and the conversation level.
4.4.5.5 Varying the Conversation Study Methods

One final detail of the Rehearse Conversation setup should be mentioned: While LEAP recommends a conversation study method to the trainee, trainees can also adjust the conversation study method any time they are rehearsing a conversation; furthermore, the database side and the speaking side of the conversation take place in separate windows and are independently adjustable, for a total of nine (three study methods in each of two win-

Figure 4.19
Rehearse Conversation: feedback screen
windows, or \textsuperscript{3}^2 \textsuperscript{2} levels of trainee involvement. Figure 4.12 shows the Rehearse Conversation and the SONAR Simulation windows; note that each window has the set of study methods buttons across the top (Observe, Focused Practice, and Full Practice).

### 4.4.6 Tutoring Decisions Summary

One goal of the work described herein is to ground the design of an intelligent tutor in instructional design theory. As others have also noted (Tennyson & Park, 1987; Jones, 1988; Wenger 1987; Lepper, M. R., Woolverton, M., Mumme, D. L., & Gurtner, J., 1993), such grounding is uncommon in ITS but, if ITS are to teach more than one or two illustrative exercises, sorely needed.

Many instructional methods, activities, and actions applied in LEAP’s design have been espoused by instructional theorists. Several theories prescribe general approaches to instruction; for example, Gagne, Briggs, & Wager, (1992), describe a nine-step process for each of five kinds of learning outcome; Collins, Brown, & Newman, (1989), describe the characteristics of ideal learning environments for cognitive apprenticeships. Collections of more narrowly-focused instructional theories are found in Joyce & Weil, (1986), and Reigeluth, (1983), many of which apply to some aspect of customer service representative training. Finally several investigators have studied learning as an inherently social activity: Lave & Wenger, (1991), Brown & Palincsar, (1989), Collins, Brown, & Newman, (1989). All these theories have something to say about the kinds of activities learners should engage in, and the selection and sequencing of those activities. There are many similarities and overlapping prescriptions among these theories. The design of one portion of instruction might be seen to conform equally well to any of several theories. On the other hand, any substantive amount of instruction will require more in-
structional decision making, on a number of levels, than any one theory pro-
vides a basis for. Traditional instructional design theories, in particular, have
implicit assumptions of a human instructor, small numbers of students (but
not one-on-one instruction), and short class periods. Not one of these as-
sumptions is correct for intelligent tutoring systems. Table 4.12 summarizes
the instructional methods, activities, and actions in LEAP and the theories
that support them.
Table 4.12
Instruction and Instructional Theory in LEAP

<table>
<thead>
<tr>
<th>Instructional Methods, Activities, &amp; Actions:</th>
<th>Instructional Theorists:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spiral Curriculum</td>
<td>Bruner (1960)</td>
</tr>
<tr>
<td>Simulation</td>
<td>Joyce and Weil (1986)</td>
</tr>
<tr>
<td>Exploration</td>
<td>Perkins (1986)</td>
</tr>
<tr>
<td></td>
<td>Rogers (1982)</td>
</tr>
<tr>
<td>Practice</td>
<td>Anderson, Boyle, Corbett, &amp; Lewis (1990)</td>
</tr>
<tr>
<td></td>
<td>Joyce &amp; Weil (1986)</td>
</tr>
<tr>
<td></td>
<td>Newell &amp; Rosenbloom (1981)</td>
</tr>
<tr>
<td></td>
<td>Gagne, Briggs, &amp; Wager (1992)</td>
</tr>
<tr>
<td></td>
<td>Brown and Palincsar (1989)</td>
</tr>
<tr>
<td>Gaining Attention</td>
<td>Gagne, Briggs, &amp; Wager (1992)</td>
</tr>
<tr>
<td></td>
<td>Keller (1983)</td>
</tr>
<tr>
<td></td>
<td>Lesgold, Eggan, Katz, and Rao (1992)</td>
</tr>
<tr>
<td>Presenting Information</td>
<td>Gagne, Briggs, &amp; Wager (1992)</td>
</tr>
<tr>
<td></td>
<td>Bloom, in Joyce &amp; Weil (1986)</td>
</tr>
<tr>
<td></td>
<td>Carroll, in Joyce &amp; Weil (1986)</td>
</tr>
<tr>
<td>Demonstrating</td>
<td>Rosenshine, in Joyce &amp; Weil (1986)</td>
</tr>
<tr>
<td></td>
<td>Gagne, Briggs, &amp; Wager (1992)</td>
</tr>
<tr>
<td></td>
<td>Brown and Palincsar (1989)</td>
</tr>
<tr>
<td></td>
<td>Brown and Palincsar (1989)</td>
</tr>
<tr>
<td>Learning Guidance</td>
<td>Gagne, Briggs, &amp; Wager (1992)</td>
</tr>
<tr>
<td>Randomness</td>
<td>Anderson, Boyle, Corbett, &amp; Lewis (1990)</td>
</tr>
<tr>
<td></td>
<td>Joyce &amp; Weil (1986)</td>
</tr>
<tr>
<td></td>
<td>Lesgold, Eggan, Katz, and Rao (1992)</td>
</tr>
</tbody>
</table>

Continued, next page
Table 4.12, continued

<table>
<thead>
<tr>
<th>Instructional Methods, Activities, &amp; Actions:</th>
<th>Instructional Theorists:</th>
</tr>
</thead>
</table>
<pre><code>                                          | Brown and Palincsar (1989)       |
                                          | Brown and Palincsar (1989)       |
</code></pre>
| Hints and Feedback                          | Gagne, Briggs, & Wager (1992)  
| Summarizing                                 | Reigeluth & Stein (1983)  
                                              | Reigeluth & Curtis (1987)         |
| Reviewing                                   | Gagne, Briggs, & Wager (1992)  
                                              | West, Farmer, & Wolff (1991)      |
                                              | Anderson (1990)                   |

Of the myriad of instructional theories affecting LEAP's design, one was particularly influential; the Cognitive Apprenticeship approach described in Collins, Brown, & Newman (1989), with its teaching methods: modeling, coaching, scaffolding and fading, articulation, reflection, and exploration, was a major design influence.

LEAP *models* the performance of expert service reps, *coaches* trainees as they practice by providing hints and feedback on each step, *scaffolds* by completing for the trainee those portions of the task they are not yet prepared to practice, *fades* by ceasing to scaffold, and encourages *articulation* by having students verbalize the spoken parts of the task. Trainees *reflect* whenever they respond incorrectly, since they must respond correctly in order to proceed. They also reflect at the end of each exercise, both on the exercise and on the topic. LEAP encourages *exploration* by permitting trainees to make their
own instructional choices of both what to study and how to study. In particular, Explore mode (Examine Contact Flow) allows the trainee to explore the domain knowledge base in detail.

A second feature of Cognitive Apprenticeship applied in LEAP is the sociology dimension: situated learning, culture of expert practice, intrinsic motivation, exploiting cooperation, exploiting competition.

Learning in LEAP is situated in the sense that trainees practice tasks they will perform on the job. The tasks are genuine and trainees are practicing them, not talking about them. LEAP provides a culture of expert practice in two ways; first by its relationship to the external environment: human experts are required in conjunction with its use -- LEAP does not replace trainers, who are themselves expert reps; second, in LEAP, learners have readily available models of expertise-in-use, as well as access to the underlying representation of that expertise in the domain knowledge base. LEAP’s realistic tasks appeal to trainees’ intrinsic motivation. Trainees know they will be doing these tasks on the job. LEAP exploits cooperation, not in the usual way of having trainees work together to solve a problem, but by asking each trainee to work with a (simulated) customer to solve his or her problem. LEAP does not directly exploit competition. LEAP emphasizes mastery learning, and learning continues until all material is mastered. Thus all trainees end up at the same point. LEAP does post trainees’ topic-level scores on the top-level screen, visible to all passersby, so a trainee’s current status is known, not only to him- or herself, but also to the instructor and to other trainees, if they care to look.

Finally, LEAP makes the structure of the domain explicit in the tables of topics and conversations, and in the branches of the domain knowledge
base; and has the trainee act within the structure of the domain to select topics and exercises, and to explore situations & actions.

Table 4.12 lists instructional features of LEAP and their theoretical support. Table 4.13 concentrates on and summarizes the instructional methods, or kinds of instructional interactions, employed in LEAP.
Table 4.13

Instructional Methods Employed in LEAP

<table>
<thead>
<tr>
<th>A. Study the Book</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Read text</td>
</tr>
<tr>
<td>2. Hear text read</td>
</tr>
<tr>
<td>3. See video</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B. Practice an Exercise</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Observe</td>
</tr>
<tr>
<td>2. Full Practice</td>
</tr>
<tr>
<td>a. Feedback</td>
</tr>
<tr>
<td>b. Graduated hints</td>
</tr>
<tr>
<td>c. Help</td>
</tr>
<tr>
<td>3. Focused Practice</td>
</tr>
<tr>
<td>a. Practice</td>
</tr>
<tr>
<td>b. Review</td>
</tr>
<tr>
<td>c. Skim</td>
</tr>
<tr>
<td>d. Scaffold</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C. Explore the Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Same instructional methods as in B. Practice an Exercise)</td>
</tr>
</tbody>
</table>

Not only does LEAP embody a set of instructional methods, LEAP also makes instructional decisions about which method to use and which content
to use it on. LEAP makes a number of instructional decisions, as listed in Table 4.14.
LEAP begins by selecting a topic and a topic study method:

Select a Topic:
- Open Customer Contact
- Service Change is Change Ring Cycle
- Get Name and Number
- Discuss Ring Cycle Change
- Negotiate Two-Day Due Date
- Determine Charge: Ring Cycle Change
- Verify Order: Change Ring Cycle
- Close Customer Contact
- Etc.

Select a Topic Study Method:
- Study a Multi-media Presentation
- Practice an Exercise
- Explore the Domain Knowledge Base\textsuperscript{a} (trainee choice)

If the selected topic study method is Practice an Exercise, then:

Select an Exercise:
- Change Ring Cycle: Lengthen
- Change Ring Cycle: Shorten
- Change Ring Cycle: Never Mind!
- Etc.

Continued, next page
Table 4.14, continued

Select an Exercise Study Method:

• Observe Expert
• Focused Practice
• Full Practice (trainee choice)

If the selected exercise study method is Focused Practice, then:

Select a study method for each SA pair:

• Practice
• Review
• Skim
• Scaffold

Note. a The study method Explore the Domain Knowledge Base appears in the illustrations as Examine Contact Flow, a terminology that is more meaningful to LEAP’s trainees.

Explore and Practice modes are two fundamentally different ways of learning. In Explore mode, trainees observe and try each SA pair as an abstraction in the context of its conceptually related materials. In Practice mode, trainees observe and try each SA pair as an instance in the context of its application in a conversation. These differences are summarized in Table 4.15.
Table 4.15

Explore and Practice: Different Ways of Learning

<table>
<thead>
<tr>
<th>Learning Mode:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explore:</strong></td>
<td>Practice:</td>
</tr>
<tr>
<td>Focus:</td>
<td></td>
</tr>
<tr>
<td>Abstract statement of SA pair</td>
<td>Concrete instance of SA pair</td>
</tr>
<tr>
<td>Context:</td>
<td></td>
</tr>
<tr>
<td><strong>Topic:</strong> An and/or tree of conceptually-related SA pairs</td>
<td><strong>Conversation:</strong> A coherent sequence of SA pairs</td>
</tr>
</tbody>
</table>

To summarize, LEAP applies a spectrum of instructional design theory as it attempts to optimize learning by making several kinds of tutoring decisions: LEAP selects topics and topic study methods; LEAP selects exercises and exercise study methods; and LEAP decides which SA pairs students should practice, which they should review, and which SA pairs it will skim and scaffold. The LEAP tutor, then, is one example of how theories of instructional design can inform the design of an intelligent tutor.

4.5 Authoring in LEAP

If a tutor's design is to be practical, it must be made to teach a different domain or content areas with a minimum of effort. LEAP's design accomplishes this in two ways: first it provides a shell, into which new domain knowledge can be put with little need for further programming; and second, it provides a set of authoring tools for the input of the new or different domain knowledge.

LEAP's two-part tutoring shell consists of: First, a representation which can be used as a 'container' for domain knowledge, second, a set of tutoring
methods that act on the domain knowledge when expressed in the representation.

The details of LEAP’s domain representation have already been described. The key point about LEAP’s representation from an authoring perspective is that LEAP is capable of tutoring any domain knowledge that can be reasonably expressed in the representation. For example, LEAP can train customer service reps (of any organization) whose tasks consist both of talking with customers on the telephone and manipulating one or more databases. By way of contrast, Philosophy 101 would not be a good domain for LEAP to teach.

As part of the effort to keep domain knowledge separate from tutoring knowledge, domain knowledge in LEAP was not entered by a programmer, but developed and entered by an instructional designer. Also, domain knowledge is stored in files which are separate from the rest of LEAP’s code, and is loaded at run time.

The tutoring methods used in LEAP have already been described. The salient points about LEAP’s tutoring methods from an authoring perspective are first, that they are built into the tutor and do not have to be re-authored when the domain changes, and second, that they operate on LEAP’s representation of domain knowledge, but are independent of the content of any domain-specific knowledge. Certain tutoring methods may be more appropriate for some domains than for others; how LEAP’s tutoring methods may be modified is described below.

Besides providing a shell to contain new domain knowledge, LEAP provides a set of authoring tools for inputting new domain knowledge.¹ An

---

¹ Working independently (at another location) a second instructional designer was able to use the authoring tools to replicate a portion of LEAP’s domain knowledge base.
instructional designer using LEAP, like an instructional designer for CBT, would need to have some experience in writing running code.

Authoring in LEAP is a five-step process that consists of:

1. Creating a domain knowledge base describing the conversation space of the topics to be taught. The conversation space is an AND/OR tree of SA pairs.

2. Capturing the required database screens and desired account characteristics.

3. Instantiating conversations as paths through the conversation space.

4. Producing scripts for actors and reps to record the conversations.

5. Revising the tutoring decision-making strategies.

These steps are described below; the first four briefly, the last in detail.

The author first creates the domain knowledge base with a conventional text or programming editor. The domain knowledge base consists of a few key words: sequence, and, or, end, etc., the SA pairs, and topic names. For example, Figure 4.20 is the initial topic as an author would write it:

---

2 The process of creating multimedia materials for the Book is ignored here. It consists of the conventional instructional design process for the development of multimedia materials.

3 This project used both Macintosh and UNIX versions of the EMACS programming editor.
This topic consists of a sequence (seq) of one SA pair and a branch (or) to one of three sub-topics (add, change, or remove voice messaging). The situation side of the SA pair consists of two situations anded together, meaning that the trainee will see the simulated BOSS database on the screen when she hears the telephone ring (actually a beep) in her headset, announcing that another customer is on the line.

The author continues defining topics, sub-topics, and SA pairs in this manner until the domain is fully described. All topics and sub-topics must eventually bottom-out in SA pairs.

Second, the author captures database screens with desired customer account characteristics using Bellcore’s WITS Author©. The screen definition files are modified as necessary with a text editor.

Third, the author uses LEAP’s Conversation Editor to traverse the domain knowledge base, inputting the customer’s and rep’s spoken lines, and conversation-specific database information. For example, for the response:

Announce yourself and determine reason for call.

The author might input:

“U S WEST Communications, this is Frank; how may I help you?”
Using the Conversation Editor is, in one way, similar to exploring the domain knowledge base via Examine Contact Flow. The author steps through the domain knowledge base, choosing among multiple situations. Whenever a conversation could branch, all branches are displayed; the author selects a branch to instantiate and types the appropriate speech or database information into the window that accompanies each situation and action. When the author has instantiated a path through the domain knowledge base with a conversation, she or he can test it immediately by running the tutor. LEAP’s ability to switch effortlessly between author and tutor modes make it easy to develop and test conversations.

Creating the domain knowledge base and the conversations is ‘knowledge engineering,’ an iterative, time-consuming process. Achieving consensus among all interested parties regarding the form and structure of the domain knowledge base and the wording of conversations is vital to the acceptance of the tutor. The LEAP development team not only sought input from a group composed of members from the union, management and training departments (all former or current reps themselves), but also used a formal sign-off process.

After inputting a conversation, script generation is effortless; the author has only to push the [Generate Script] button. Dialogs are recorded directly onto disk using the UNIX Audiotool. To record the reps’ speaking parts, the LEAP project used genuine expert reps, who volunteered for the task. On one occasion when no expert reps were available, semi-professional actors were used. To record the customer speaking parts, we used the same expert reps and actors, as well as passersby snagged from the corridor outside the recording room. The file containing the recording of each spoken line is edited, if necessary, then linked to the appropriate situation or action. The
fifth authoring step, revising the tutoring decision-making strategies, is described in the remainder of this section.

4.5.1 Revising LEAP’s Decision-making Strategies

The final authoring step is to revise LEAP’s decision-making strategies (described earlier). These strategies are not hard-coded into LEAP, but are easily modified. Tutoring decisions are made by taking into consideration a number of factors. LEAP’s authoring interface has been designed so that the weight given each factor is easily varied. The contribution of each factor to the tutoring decision can vary from zero to one hundred percent. That is, the factor can be removed from LEAP’s decision-making altogether, or it can be the sole consideration in the decision, or be anywhere in between. The influence of each of these factors on the tutoring decision can be varied simply by moving a slider beside the factor’s name on an authoring screen.

In this section we will describe how to vary LEAP’s decision-making process for selecting topics, selecting conversations, and for practicing, reviewing, skimming and scaffolding SA pairs.

4.5.1.1 Selecting Topics

Figure 4.21 shows the Topic Choice screen. Topic selection is based on a consideration of three factors: the topic sequence in the Table of Contents, the topic last studied, and the trainee’s proficiency on each topic. Moving the sliders adjusts the weight of the factor’s contribution to the decision from 0% to 100%. Regarding the trainee’s proficiency factor and its effect on the spiral curriculum, the current design allows the author to select one of three predetermined sequences of priorities. A more flexible design for assigning and sequencing topic ratings is desirable.

When creating the course materials, the author must take care to sequence the topics according to accepted instructional design principles (e.g.,
prerequisites first), because one of the topic selection factors considers this author-defined sequence. As the trainee progresses through the course, LEAP will note his or her proficiency and preferences and resequence the topics accordingly. The current defaults are set so that trainee proficiency is weighted more heavily than topic sequence, with the effect that a topic, once begun, is mastered before going on to another. If a new, different, domain or content area, one with more prerequisite relations among the topics, were to be put into LEAP, the topic sequence factor would have to be given more weight (by moving the slider).

Figure 4.21
Study parameters: “Topic Choice” selected

152
4.5.1.2 Selecting Conversations

Figure 4.22 shows the Conversation Choice screen. As mentioned in the section *How LEAP Selects Conversations*, LEAP selects a conversation for study by ranking each conversation according to several weighted factors and selecting the highest-ranking one. Two of the six factors LEAP considers in ranking conversations are characteristics of the conversation, two factors are assigned by the author, and two are related to the current state of the student model. As with the topic selection factors, the relative weight given to each of these conversation selection factors is adjustable. Weights are adjusted by moving sliders beside each factor's name on the Conversation Choice screen. The author establishes the conversation sequencing process by setting the relative weight of each factor. Weighting should be based on a consideration of the domain, the trainees, the course objectives, and field testing. In the future it may be possible for the tutor to experiment and, to some extent, adjust the weights itself.
4.5.1.3 Skimming and Scaffolding SA Pairs

Figures 4.23, 4.24, 4.25, and 4.26 show how the author can revise the probabilities for skimming. The decision to skim or not to skim is made by chance, with the probability of skimming determined by two factors: 1) whether the SA pair is in the current topic; *ceteris paribus*, SA pairs in the current topic are more likely to be asked, and 2) how well trainees ‘know’ the SA pair; the better they know it, the more likely LEAP will skim it. The au-
Thor revises the skimming decision odds by, as usual, manipulating slider bars.

Figure 4.23 introduces the conceptual structure underlying the screen displayed in Figure 4.24, whose labels may be more confusing than clarifying. Similarly for Figures 4.25 and 4.26, respectively.

![Figure 4.23](image)

**Figure 4.23**

*Study parameters: conceptual structure for current topic*

Figure 4.23 is a diagram for the current topic. A trainee’s score for each SA pair will fall somewhere between zero and one. The graph shows the probability of a trainee being asked to Practice or Skim an SA pair, depending on its score.
Study parameters: “In-Topic Question Probabilities” selected

Figure 4.24 is the screen for the current topic. “In Topic Challenge” is the reciprocal of the student SA pair score. The “ask chance” is the probability of the trainee being asked to practice the SA pair. (If the trainee is not asked, LEAP will skim the pair.) Moving the sliders changes the probabilities as follows: The first slider “Question when not known?” moves the left portion of the line vertically. The second slider “Question when known?” moves the right portion of the line vertically. The third slider “Decrease questioning when known?” adjusts the slope of the right portion of the line.
Study parameters: conceptual structure for other topics

Figure 4.25 is a diagram for the other topics, the ones that are not current. Again, a trainee's score for each SA pair will fall somewhere between zero and one. The graph shows the probabilities that LEAP will Scaffold or Skim, or that a trainee will Review an SA pair, as a function of its score.
Figure 4.26

Study parameters: “Out of Topic Question Probabilities” selected

Figure 4.26 is the screen for the topics that are not the current focus of attention. It is identical to the current-topic screen except for two features: First, in general, the curve is lower than the curve for the current topic, indicating that the trainee is less likely to be asked to respond to any SA pair that is not in the current topic. Second, the left end of the curve shows how scaffolding is implemented. When the ‘challenge’ level of an SA pair is high, the ‘ask chance’ is nil. That is, trainees will not be asked to review an SA pair that they have not yet practiced and is out of topic.
4.5.2 Authoring in LEAP: Summary

LEAP is a practical tutor. It can be made to teach a different domain or content areas with a minimum of effort. LEAP provides a shell and a set of authoring tools for the input of new or different domain knowledge. The shell consists of a representation, used as a 'container' for domain knowledge, and a set of tutoring skills that act on the domain knowledge. Authoring is a five-step process and, as shown in Table 4.16, there are tools for each step:

Table 4.16
Authoring in LEAP: Steps and Tools

<table>
<thead>
<tr>
<th>Authoring Step</th>
<th>Tool:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create a domain knowledge base describing the conversation space of the topics to be taught.</td>
<td>Conventional text or programming editor</td>
</tr>
<tr>
<td>2. Capture the required database screens, and accounts with the desired characteristics.</td>
<td>Bellcore's WITS Author</td>
</tr>
<tr>
<td>3. Instantiate conversations as paths through the conversation space.</td>
<td>LEAP's Conversation Editor</td>
</tr>
<tr>
<td>4. Produce scripts for actors and reps, and record the conversations.</td>
<td>LEAP's Script Generator, UNIX Audiotool</td>
</tr>
<tr>
<td>5. Revise the tutoring decision-making strategies.</td>
<td>LEAP's Tutoring Decisions Adjuster</td>
</tr>
</tbody>
</table>

The final authoring step is to revise LEAP's decision-making strategies. LEAP's Tutoring Decisions Adjuster permits the weight given each decision-influencing factor to be varied by moving a slider beside the factor's name on
an authoring screen. The contribution of each factor to the tutoring decision can vary from zero to one hundred percent. Adjusting the weights of the factors varies LEAP's tutoring decision-making process for sequencing topics and conversations, and for practicing, reviewing, skimming and scaffolding SA pairs.

4.6 Design: Summary

In this section we have described LEAP's domain representation, student model, instructional design and tutoring decision making, and authoring tools. LEAP demonstrates the extent to which it is possible to build an ITS with its main focus on tutoring, incorporating instructional design theory and a range of tutoring skills, rather than a focus on domain knowledge.

The central item in LEAP's domain representation is the SA pair. Conversations are meaningful sequences of instantiated SA pairs. The domain knowledge base is made up of possible conversations. A topic is a conceptually related portion of the domain knowledge base. The conversations in the tutor must cover all the important branches of the domain knowledge base to be learned by the trainee.

LEAP optimizes the learning process by maintaining and consulting a detailed student model. LEAP calculates a score for every SA pair, exercise, and topic the student tries, and uses the scores both to provide feedback to the student and to select topics, exercises and SA pairs for the student to practice, as well as to select the methods for practicing them. Using the student model to focus practice means exercises can be realistic and complex, as in apprenticeship learning. LEAP adjusts its instruction as trainees learn or forget.

One goal of this work is to ground the design of an intelligent tutor in instructional design theory, uncommon in ITS research, but sorely needed. There are many instructional methods, activities, and actions in LEAP's de-
sign. They are supported by theories such as those in Gagne, Briggs, & Wager, (1992), Collins, Brown, & Newman, (1989), Joyce & Weil, (1986), Reigeluth, (1983), Lave & Wenger, (1991), and Brown & Palincsar, (1989). Traditional instructional design theories may have assumptions that do not pertain to intelligent tutoring systems; nevertheless, many of their prescriptions apply. One theory, Cognitive Apprenticeship, (Collins, Brown, & Newman, 1989), with its teaching methods: (modeling, coaching, scaffolding and fading, articulation, reflection, and exploration) was a major design influence. LEAP is an example of how theories of instructional design can inform the design of an intelligent tutor.

LEAP provides a shell and a set of authoring tools for the input of domain knowledge. The shell consists of a representation and a set of tutoring skills. Authoring is a five-step process: create a domain knowledge base, capture database screens and accounts, instantiate conversations, record the conversations, and revise the tutoring decision-making strategies.
CHAPTER 5
EVALUATION

The evaluation of LEAP focused on the following main themes: measuring trainees' affective responses toward LEAP and their perceptions of its usability and instructional value, measuring trainees' usage of LEAP, and the resulting learning, measuring how well LEAP was capable of individualizing instruction, and measuring instructors' response to having LEAP in their classrooms.

In the sections that follow I first describe the evaluation methodology, then I present initial conditions: trainee demographic data and instructors' initial notions of what using LEAP might be like. Next I present the results of questions characterizing trainee affective responses to using LEAP, including a review of how those responses changed over the course of time, and trainee perceptions of LEAP's general usability. In the fourth section I present trainee perceptions of the instructional value of LEAP's features; LEAP's top level and each of the three major study methods is examined in detail, with emphasis on Rehearse Conversation.

In contrast to trainee perceptions, in the fifth section I report on actual usage data as recorded by LEAP's logging facility: how much LEAP was used, how much was learned, which commands were used and how often, and how trainee self-directedness influenced achievement. In the sixth section I examine the extent to which LEAP managed to individualize instruction for each trainee by varying the selection and sequence of exercises each trainee received; this section also examines the extent to which three trainees who attain the same performance scores arrive there via paths unique to their preferred learning style.
In the seventh section I consider the effects of changes made in LEAP over the course of the field trials: improvements to the interface, dialog changes to accommodate different state laws, etc. In the eighth and final section I summarize instructors’ reflections after seeing their trainees use LEAP for the day-long session.

5.1 Methodology

This section describes the subjects, the materials, the apparatus, and the procedure used in the LEAP field trials. The subjects were five groups of 5 to 6 service rep trainees, data were gathered with logs and questionnaires administered before, during, and after the use of LEAP, and the procedure was a day-long training session.

5.1.1 Subjects

LEAP was used in a day-long learning session by twenty-nine service representative trainees, and five instructors. The trainees were in five groups, each with its own instructor, in U S WEST’s standard ten-week service rep training course. The courses had been typically been running a few weeks when LEAP was trialed. All the courses covered the same general curriculum; instructors had a wide latitude in the sequencing of materials and emphasis, or depth of coverage. In each course, LEAP was trialed within the range of time that training in Voice Messaging would be appropriate.

5.1.2 Materials

To record trainee data during the field trial a number of materials were used before, during and after the trial:

- Pre: Trainee questionnaire: Demographic data about the trainees, including characteristics that might be expected to affect their performance on the tutor, for example, years of computer experience, years of customer service experience, years with the company.
• Pre: Instructor questionnaire: Information was also collected from the instructor before the students tried the tutor, such as the instructor’s attitude toward the tutor, including the instructor’s visions of the possible good and bad outcomes of having this type of training tool available, and the related instruction that trainees have already had.

• During: Trainee reaction sheet: measured trainees’ general reactions to using LEAP. Collecting these measures periodically enabled the observation of any shift in perception as trainees became familiar with LEAP.

• During: LEAP not only created a student model, but also recorded changes in its state over time. Collecting the student model periodically enabled comparison of initial and final measures of knowledge, and paths and rates of learning.

• During: LEAP logged trainee activities during field trial. This enabled analysis of usage of the various instructional features.

• Post: Usability questionnaire: Trainees were asked about the usability of the various features.

• Post: Trainee reaction sheet: Measured trainees’ general reactions to using LEAP. While these measures were collected periodically during the trial period, they were also collected at the end of the training to capture the trainees’ final perceptions of LEAP.

• Post: Instructional impact questionnaire: Measured trainee responses to general instructional features of LEAP.

• Post: Instructor’s questionnaire: Instructors were asked to reflect on the effect(s) of having LEAP in their class.

• Post: Instructional Value questionnaire: Trainees reflected on the learning process and the quality of instruction. The features that attracted their attention, the features that caused them problems, their ability to interpret what they saw, and their degree of participation and interest were all noted.

5.1.3 Apparatus

For the field trials, a local ethernet network of six Sun Sparc workstations was set up, each with keyboard, mouse, and a large color monitor iden-
tical to the monitors reps use in the workplace. The apparatus for the trial was described more fully in the Design section.

5.1.4 Procedure

The field trial was a day-long event for each of five training groups. Two groups were located in Colorado Springs CO, one in Loveland CO, one in Phoenix AZ, and one in Denver CO. Groups consisted of five or six trainees and one instructor. Training sites were co-located with customer service centers and trainees often observed service reps at work. Each field trial took place as part of a regularly scheduled 8 to 10 week service representative training class on a suitable date (not too early or too late in the training) as determined by negotiation with the instructor. The field trial schedule was as follows:

<table>
<thead>
<tr>
<th>Time</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>8:00</td>
<td>Introduce the Field Trial to trainees</td>
</tr>
<tr>
<td>8:15</td>
<td>Train trainees on how to use LEAP</td>
</tr>
<tr>
<td>9:00</td>
<td>First session</td>
</tr>
<tr>
<td>10:30</td>
<td>Break</td>
</tr>
<tr>
<td>10:45</td>
<td>Second session</td>
</tr>
<tr>
<td>12:15</td>
<td>Lunch</td>
</tr>
<tr>
<td>1:45</td>
<td>Third session</td>
</tr>
<tr>
<td>3:15</td>
<td>Break</td>
</tr>
<tr>
<td>3:30</td>
<td>Post-questionnaires</td>
</tr>
</tbody>
</table>

Introduce the Field Trial to Trainees. The LEAP team\(^1\) and the trainees introduced themselves to each other. The researcher explained the objectives of the field trial: For the LEAP team: To observe how LEAP works with real

---

\(^1\) Typically, the team leader, the researcher/instructional designer, the programmer, and the UNIX/network guru would be on hand for a field trial.
trainees using it. For the trainees: To practice role plays for change orders, specifically the three Voice Messaging Service (VMS) change orders in LEAP: Add VMS, Change VMS ring cycle, and Remove VMS. The researcher compared LEAP to the role-plays trainees were accustomed to:

- LEAP provides simulated customers (all trainees are practicing).
- LEAP provides simulated SONAR and CSR accounts.
- LEAP makes sure trainees know every fact about VMS to a certain degree of expertise.
- LEAP keeps trainees working at the edge of their competence; not doing tasks that are too easy or too hard.

The trainees filled out a pre-trial demographic questionnaire, and the instructors filled out a pre-trial questionnaire regarding their perceptions of the technology and the VMS related knowledge trainees already had.

**Train Trainees on How to Use LEAP.** The researcher then trained trainees on how to use LEAP. The instruction consisted of walking trainees through the use of LEAP, approximately as in the Appendix *LEAP Walkthrough*. The training was hands-on, not a lecture or a demonstration. All LEAP team members joined in to keep trainees on track. First trainees tried all the features of the top-level screens, then they tried the features of the Book, then they worked through a short exercise, first in Observe mode, then in Full Practice mode. For consistency among trainee groups, the researcher always followed the same set of notes. At a convenient time, after trainees had become familiar with the basics of LEAP, the researcher introduced Examine Contact Flow.

**First Session.** Trainees used LEAP to learn VMS change orders in all three sessions. In the first session the LEAP team kept a close watch on the activities to ensure the hardware and software were running correctly, to en-
sure the trainees were using LEAP as intended (though trainees chose what and how to study), and to answer questions about LEAP and about computer use (trainees had varying degrees of experience with keyboard, mouse, and window environments). The instructor provided trainees with information about VMS as needed, and observed the learning process. The first session often involved a lot of discussion. Trainees filled out a ‘How are we doing?’ questionnaire at the end of all three sessions.

**Second Session.** In the second session, things were quieting down; discussion focused on VMS rather than LEAP usage, and discussion was among trainees or between trainees and the instructor (the instructor could plug her own headset into a trainee’s and listen to both sides of the conversation). The researcher had a chance to observe each trainee carefully and repeatedly. The LEAP team had a chance to make notes of which parts of LEAP were working well and which needed minor (occasionally major) fixing.

**Third Session.** By the third session trainees were engrossed in the role plays, notably oblivious to other events in the classroom. At first the LEAP team wondered among ourselves “How can they do this all day long and still be so into it?” The answer seemed to be that this was the job they were training to do. Their temperament and experience had prepared them for it, and now they had the opportunity to do it.

**Post-questionnaires.** After the third session trainees filled out a set of questionnaires, described above, and instructor filled out a post-training questionnaire.

---

2 "Who are all these people?" one trainee was heard to say in surprise as she suddenly became aware of the small crowd of training and H&PS managers and union officials that had gathered to observe LEAP in use at one trial site.
5.2 Initial Conditions

This section presents trainee demographics and instructor anticipations.

5.2.1 Trainee Demographics

Trainee demographic data is described in text and presented visually in Figures 5.1 through 5.7. The data characterize trainee age, education, related work experience, sales skills, prior use of voice messaging, and computer and software skills.

The modal trainee age range is 25-30 years (Figure 5.1), their mean age is 37, and their range is from 19 to 55. Trainees were well-educated: (Figure 5.2), while 8 of the 29 had completed only high school, the majority had from one to five years of college.

Eight of the trainees were new-hires. Ten had from 1 to 10 years of experience with a phone company (Figure 5.3), the remaining eleven, had from 10 to 23 years of experience. Most of the trainee who had experience with a phone company had it with US WEST (Figures 5.3 and 5.4).

Regarding service rep experience, 15 of the 29 have none (Figure 5.5). In contrast, four trainees had more than 15 years of experience. Presumably, these four trainees could have completed training much faster than the inexperienced trainees, had individualized instruction been available to them in the classroom as well as in LEAP. Trainees consider themselves to be of average sales ability (Figure 5.6).
Figure 5.1

Trainee age, years

Figure 5.2

Trainee education, years
Trainee experience with any telephone company, years

![Figure 5.3](image)

Trainee experience with U S WEST, years

![Figure 5.4](image)
Figure 5.5

Trainee Service Representative experience, years

Figure 5.6

Trainee sales skills
Most trainees were familiar with VMS (Table 5.1). Two-thirds (19/29) said they used voice messaging either at home or at work, and two-fifths (12/29) said they used it very often at one of those sites.

Table 5.1
Trainee Experience with Voice Messaging

<table>
<thead>
<tr>
<th>Question</th>
<th>Response Selected</th>
<th>Count</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use VMS at Home?</td>
<td>Yes</td>
<td>17</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>12</td>
<td>41</td>
</tr>
<tr>
<td>How often at home?</td>
<td>Very Often</td>
<td>11</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>4</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Very Little</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Use VMS at Work?</td>
<td>Yes</td>
<td>9</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>19</td>
<td>68</td>
</tr>
<tr>
<td>How often at work?</td>
<td>Very Often</td>
<td>5</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Average</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Very little</td>
<td>4</td>
<td>40</td>
</tr>
</tbody>
</table>

All but one trainee (28 of 29) reported having experience using computers. But of the 28, 14 reported no experience with word processing, accounting/finance, spreadsheet, or database software; so their experience was
probably minimal. The remaining trainees reported from 1 to 17 years experience with one or more of these types of program (Figure 5.7).

![Trainee experience with computers and software](image)

**Figure 5.7**

Trainee experience with computers and software

To summarize, trainees were a wide range of ages, with clusters in their late 20s and late 30s, had a wide range of formal education, clustering around high school and two-year degrees, a wide range of work experience, though many were new-hires, a wide range of voice messaging experience, though a third had none, and a wide range of computer and software experience, though half had very little or none.

5.2.2 Instructor Anticipations

During the introduction to the field trial, instructors filled out a questionnaire describing their feelings and their students’ feelings about using an ITS in their class, their beliefs about ITS’ possible positive and negative im-
pact on them and on their students, and the instruction the students had received relating to the task they would practice in LEAP. The results are summarized for each of the questions:

(1) In general, how do you feel about using LEAP in your class?

All instructors' responses about using LEAP in their class were positive, e.g., "Very excited, any new technology that will enhance the learning process, I'm all for."

(2) What three things do you think LEAP will help you most with?

Instructors mentioned LEAP would help them most with: practice, reinforcement, confidence building, consistency of training, individualization of instruction, and independent learning.

(3) In what three ways do you think LEAP might be most helpful for your students?

Instructors thought LEAP would help their students most with practice, confidence, reinforcement, mastery learning, providing a safe practice space, and providing individualized instruction.

(4) What are the three worst impacts that you could imagine LEAP having on you, the instructor?

The worst impacts LEAP could have on them, the instructors, were: replacing the instructor, frightening students, and taking too much time.

(5) What are the three worst effects that you could imagine LEAP having on your students?

The worst effects LEAP could have on their students were: loss of needed human interaction and team feeling, and trainees' fear of computers or software.

(6) What have your students already studied that could affect their performance on VMS change orders, and for how many hours have they studied each topic?
Related information that the students had already studied: VMS as product: 2 to 4 hours, depending on the class. Study typically consisted of reviewing the use of VMS materials in the Desk Reference. Study on other related materials varied widely, some had used SONAR, others had not; some had studied sales techniques, some had not; some had studied change orders (the type of sales order used in the field trial), others had not.

(7) How would you characterize students’ attitudes towards LEAP at this time?

The instructors characterized students’ pre-trial attitudes towards LEAP as mostly positive: "unknown," "curious," "positive," "very positive," "excited."

To summarize, instructors’ attitudes toward LEAP as a teaching technology were positive and they believed their students’ attitudes were also positive. They believed LEAP would help the training process by providing an opportunity for practice and reinforcement, building confidence, individualizing instruction and promoting mastery learning. They feared LEAP would replace needed human interaction with frightening machines. Their classes varied widely in the preparatory study of the material practiced in LEAP. Next, we examine trainees’ affective responses to LEAP.

5.3 Affective and Usability Results

The data reported in this section are from questions about trainee self-perceptions during LEAP use, trainee attitudes toward role plays with humans versus LEAP, and trainee perceptions of LEAP’s usability.

These data, and much of the other questionnaire data gathered in this evaluation were gathered using a semantic differential scale, a seven point graphic rating scale with anchor words; as originally developed by Osgood and his coworkers (Osgood, Suci, & Tannenbaum, 1957), and widely used in behavioral research (Rosenthal & Rosnow, 1984). A sample question is shown
in Figure 5.8. Trainees were instructed to first select the anchor word that most closely described their response to the question, and then to indicate the strength of their response by marking the scale.

15) I believe my proficiency is ____ LEAP's proficiency ratings.

| Above | 3 | 2 | 1 | 0 | 1 | 2 | 3 | Below |

Figure 5.8
Example of questionnaire format

The results are summarized and presented in Tables 5.2 through 5.6. Some of the more interesting response patterns, such as bimodal responses, are presented as bar graphs overlaying the original question (Figures 5.9 through 5.18). The significance test refers to the question of whether the responses are significantly different from a 'non-committal' response of 0, using an alpha level of .05. Because of the large number of questions (more than thirty) the reader might expect some of the tests to yield significant results owing to chance alone. The Bonferroni procedure for protecting against 'too many t tests' by dividing the alpha level by the number of tests performed, (.05/33 = .0015) provides a more conservative estimate of significance (Rosenthal & Rosnow, 1984). However, in most cases, this estimate does not change the significance of the results.

5.3.1 Trainee Affective Response Toward Self and System

A number of questions related to trainees' self-perceptions while using LEAP. These self-perception questions were asked on the questionnaire entitled Does the System Delight You? administered at the end of the day. Briefly, trainees who used LEAP reported they learned a lot, understood how to use
LEAP, and were happy, confident, in control, interested, and pleased. These results are presented in Table 5.2 below.

Questions 1-15 of the *Does the System Delight You?* questionnaire were also administered at the end of each session (using the *How are We Doing?* questionnaire). As a result, changes in trainee responses over the course of the day could also be measured. The significance test (a repeated measures ANOVA) refers to whether there was a significant movement in the mean of the responses over the course of the day. The results are, in short: Trainees reported feeling increasingly that they were learning a lot, understanding how to use the system, happy, confident, in control, and pleased. These results are in the last two columns of the Tables below 5.2 and 5.3.
Table 5.2
Trainee Affective Response Toward Self

<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean</th>
<th>Signif.</th>
<th>Movement during day</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.</td>
<td>I learned a lot (vs. a little).</td>
<td>1.44</td>
<td>.0002</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>4.</td>
<td>I felt I understood (vs. did not understand) how to use the system.</td>
<td>1.64</td>
<td>.0001</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>8.</td>
<td>I feel happy (vs. miserable).</td>
<td>1.30</td>
<td>.0003</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>9.</td>
<td>I feel confident (vs. doubtful).</td>
<td>1.13</td>
<td>.0012</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>10.</td>
<td>I feel in control (vs. resigned).</td>
<td>1.13</td>
<td>.0031</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>11.</td>
<td>I feel interested (vs. bored).</td>
<td>1.04</td>
<td>.0098</td>
<td>ns</td>
<td>--</td>
</tr>
<tr>
<td>12.</td>
<td>I feel pleased (vs. frustrated).</td>
<td>0.74</td>
<td>.042</td>
<td>Significant</td>
<td>Better</td>
</tr>
<tr>
<td>13.</td>
<td>I feel energized (vs. tired).</td>
<td>0.57</td>
<td>ns</td>
<td>ns</td>
<td>--</td>
</tr>
</tbody>
</table>

Note. a Responses were bimodal, one mode on zero, the other on energized.

Three of the trainees’ self-perception questions, 16, 23, and 27, had a common feature: each asked if trainees would prefer to practice conversations with LEAP or a human. Question 16 (Figure 5.9) offered the choice of practicing role-plays with LEAP versus a fellow trainee, question 23 (Figure 5.10) with LEAP versus an instructor, question 27 (Figure 5.11) with LEAP versus real customers. The means of the responses to the three questions were not significantly different from zero. However, the interesting feature of these responses is not their means but their shapes: as bar graph plots of the responses reveal, trainee responses spanned the spectrum of possible responses, with
modes at the extremes of the scale. Trainees either strongly preferred LEAP role plays or strongly preferred human role plays.

As might be expected, responses to these three questions are significantly correlated ($p < .01$). Responses to the three questions were summed to attain an overall rating of trainee preference for practice with LEAP versus practice with a human. Such a preference might reasonably be expected to influence trainee learning from LEAP, for example, one might expect that trainees who prefer practicing with LEAP to practice longer, learn faster, or learn more when using LEAP than those who prefer practicing with humans. However, such is not the case; there is no relation between trainee practice time, learning rate, or total learning (Figures 5.56, 5.64, and 6.62 respectively) and preference for practice with LEAP versus practice with a human ($r = .17, p < .45$), ($r = .035, p < .88$), ($r = .118, p < .61$) respectively. In other words, how long, how fast and how much trainees learned from LEAP is unaffected by trainee like or dislike of LEAP as a role-play medium.
16) I would prefer to practice customer contacts with ____________.

![Figure 5.9](image)

Trainees' preferred medium for practicing customer contacts, 1 of 3

23) I would prefer to practice customer contacts with ____________.

![Figure 5.10](image)

Trainees' preferred medium for practicing customer contacts, 2 of 3
Trainee affective responses toward LEAP: Three questions (questions 1, 5, and 7) pertained to trainees' affective responses toward LEAP. As before, the questions were asked on the Does the System Delight You questionnaire administered at the end of the day. The three items were also administered periodically -- at the end of each session during the day -- so that changes in trainee perception could be tracked as well. As before, trainee responses were significantly different from the noncommittal rating of zero, and in the desired direction (Table 5.3). Unlike trainees' self-perceptions, trainees' affective responses toward LEAP did not change significantly over the course of the day. To summarize, trainees thought the system was fun and interesting, in short, wonderful; and their opinions did not change over the course of the day.
Table 5.3
Trainee Affective Response Toward LEAP

<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean 0-3</th>
<th>Signif. p &lt;</th>
<th>Movement during day</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The system was wonderful (vs. terrible).</td>
<td>1.46</td>
<td>.0002</td>
<td>ns</td>
<td>--</td>
</tr>
<tr>
<td>5</td>
<td>I felt the system was fun (vs. no fun) to use.</td>
<td>1.78</td>
<td>.0001</td>
<td>ns</td>
<td>--</td>
</tr>
<tr>
<td>7</td>
<td>The system seemed interesting (vs. boring).</td>
<td>1.86</td>
<td>.0001</td>
<td>ns</td>
<td>--</td>
</tr>
</tbody>
</table>

5.3.2 Usability: General

The remaining questions in the Does the System Delight You questionnaire (Tables 5.4 and 5.5) dealt with general system usability. To summarize Table 5.4, in general, trainees reported: the system was easy to use, the commands were easy to remember, error recovery was easy, usage was straightforward and consistent, the system kept them informed and the feedback was especially useful. Furthermore, trainees reported: the screen sequence and the organization of information on the screens was very clear, the design and operation of the system meet the needs of both experienced and inexperienced users equally well, and they could continue to use the system without help.

Some responses were not significantly different from zero (Table 5.5): in fact, the questions were written in such a way that zero was the desirable value, and the mean responses were close to zero.
Table 5.4  
Trainee Perception of LEAP’s Usability, 1 of 2

<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean 0-3</th>
<th>Signif. p &lt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>The system was easy to use (vs. hard to use).</td>
<td>1.18</td>
<td>.0013</td>
</tr>
<tr>
<td>6</td>
<td>Feedback provided by the system was useful (vs. not useful).</td>
<td>2.09</td>
<td>.0001</td>
</tr>
<tr>
<td>17</td>
<td>It was easy (vs. difficult) to recover from errors in using the system.</td>
<td>1.04</td>
<td>.0054</td>
</tr>
<tr>
<td>18</td>
<td>There was a consistent (vs. no consistent) way of doing things.</td>
<td>1.3</td>
<td>.0009</td>
</tr>
<tr>
<td>22</td>
<td>I could continue to use the system without (vs. only with) someone helping me.</td>
<td>1.14</td>
<td>.0003</td>
</tr>
<tr>
<td>26</td>
<td>It is easy (vs. difficult) to remember the names, and use of, commands.</td>
<td>1.3</td>
<td>.0001</td>
</tr>
<tr>
<td>28</td>
<td>Tasks are always (vs. never) performed in a straightforward manner.</td>
<td>1.22</td>
<td>.0009</td>
</tr>
<tr>
<td>29</td>
<td>The system always (vs. never) kept me informed about what it was doing.</td>
<td>1.46</td>
<td>.0002</td>
</tr>
<tr>
<td>30</td>
<td>The organization of information on the screen was very clear (vs. confusing).</td>
<td>1.36</td>
<td>.0014</td>
</tr>
<tr>
<td>31</td>
<td>The sequence of screens was very clear (vs. confusing).</td>
<td>1.13</td>
<td>.0041</td>
</tr>
<tr>
<td>32</td>
<td>Both experienced and inexperienced users’ needs were always (vs. never) taken into consideration in the design and operation of this system.</td>
<td>.96</td>
<td>.0002</td>
</tr>
</tbody>
</table>
Table 5.5
Trainee Perception of LEAP’s Usability, 2 of 2

<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean 0-3</th>
<th>Signif. p &lt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.</td>
<td>In Practice mode, LEAP expects me to respond too often (vs. not often enough).</td>
<td>.30</td>
<td>.169</td>
</tr>
<tr>
<td>15.</td>
<td>I believe my proficiency is above (vs. below) LEAP’s proficiency ratings.</td>
<td>.03</td>
<td>.437</td>
</tr>
<tr>
<td>33.</td>
<td>If there was any bias in the design of this system, it was towards experienced (vs. inexperienced) computer users.</td>
<td>.22</td>
<td>.273</td>
</tr>
</tbody>
</table>

Of these responses, question 33 corroborates question 32. The responses to questions 14 and 15 are especially interesting, and the data are displayed in vertical bar graphs in Figures 5.12 and 5.13. Trainees appear to be quite content with the frequency LEAP expects them to respond to situation-action pairs (Figure 5.12); this is gratifying, since LEAP makes an effort to be clever about effectively dividing the task of responding between the student (practice and review) and LEAP (skim and scaffold). Trainees also believe their actual proficiencies are accurately reflected in the proficiency ratings LEAP assigns them (Figure 5.13). Again this is gratifying since the weighted three-factor student model used in LEAP is a new one for tutoring systems.
14) In Practice mode, LEAP expects me to respond _______.

Figure 5.12
Trainee perception of LEAP’s interactivity

15) I believe my proficiency is _____ LEAP’s proficiency ratings.

Figure 5.13
Trainee perception of LEAP’s Proficiency ratings
5.3.3 Usability: Specific

Another questionnaire, Is this System Usable?, dealt with specific system usability. As before, a t-test compares the mean to a ‘noncommittal’ value of zero. The responses are presented in Table 5.6.

For most items (eleven of eighteen) responses were significantly different from zero and were in the desired direction. Trainees found LEAP was generally easy to use. The button names were clear. LEAP’s trainee proficiency ratings were likable and understandable. The guide was easy to read and to navigate through. The conversations were straightforward, realistic, and the voices were easy to understand. The summary videos were helpful. On the negative side, LEAP’s response time was perceived as generally slow (excepting voice and video response), and the videos, while neither great nor distracting, were perceived as too small and as jumpy (which they were). As a whole, trainees did not mind hearing their own voices, though responses spanned the spectrum, with peaks at the center and at the extremes of the scale.
<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean</th>
<th>Signif. p &lt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>The Guide was easy (vs. difficult) to navigate through.</td>
<td>1.79</td>
<td>.0001</td>
</tr>
<tr>
<td>2.</td>
<td>The conversation simulations were straightforward (vs. confusing).</td>
<td>2.07</td>
<td>.0001</td>
</tr>
<tr>
<td>3.</td>
<td>The conversation simulations were realistic (vs. contrived).</td>
<td>1.79</td>
<td>.0001</td>
</tr>
<tr>
<td>4.</td>
<td>The button names were clear (vs. confusing).</td>
<td>2.07</td>
<td>.0001</td>
</tr>
<tr>
<td>5.</td>
<td>The dialogue voices were easy (vs. difficult) to understand.</td>
<td>1.93</td>
<td>.0001</td>
</tr>
<tr>
<td>6.</td>
<td>I felt comfortable (vs. uncomfortable) hearing my own voice.</td>
<td>.45</td>
<td>.1116 ns</td>
</tr>
<tr>
<td>7.</td>
<td>The summary videos at the end of each conversation were helpful (vs. unhelpful).</td>
<td>.89</td>
<td>.0114</td>
</tr>
<tr>
<td>8.</td>
<td>All the videos were too small (vs. too large).</td>
<td>0.57</td>
<td>.002</td>
</tr>
<tr>
<td>9.</td>
<td>The print on the guide was easy (vs. difficult) to read.</td>
<td>1.70</td>
<td>.0001</td>
</tr>
<tr>
<td>10.</td>
<td>The animations introducing each chapter were easy (vs. difficult) to see.</td>
<td>1.52</td>
<td>.0001</td>
</tr>
<tr>
<td>11.</td>
<td>I did (vs. did not) like the words used in the proficiency ratings.</td>
<td>1.36</td>
<td>.0002</td>
</tr>
<tr>
<td>12.</td>
<td>In general, the LEAP system responded slowly (vs. quickly) when I was using it.</td>
<td>0.93</td>
<td>.0078</td>
</tr>
<tr>
<td>13.</td>
<td>Whenever I clicked for a video or animation to be presented, the video would appear too slowly (vs. too quickly). (Zero is good.)</td>
<td>0.39</td>
<td>.0625 ns</td>
</tr>
<tr>
<td>14.</td>
<td>Whenever I clicked to hear a text being read, the voice would begin too slowly (vs. too quickly).</td>
<td>0.14</td>
<td>.2018 ns</td>
</tr>
</tbody>
</table>

Continued, next page
Table 5.6, continued

<table>
<thead>
<tr>
<th>#</th>
<th>Question</th>
<th>Mean 0-3</th>
<th>Signif. p &lt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>LEAP was generally easy (vs. difficult) to use.</td>
<td>1.46</td>
<td>.0001</td>
</tr>
<tr>
<td>16</td>
<td>The overall quality of animations and videos was jumpy (vs. smooth).</td>
<td>0.89</td>
<td>.0134</td>
</tr>
<tr>
<td>17</td>
<td>The overall quality of animations and videos was great (vs. distracting)</td>
<td>0.21</td>
<td>.2685 ns</td>
</tr>
<tr>
<td>18</td>
<td>I did (vs. did not) understand the words used in the proficiency ratings.</td>
<td>1.75</td>
<td>.0001</td>
</tr>
</tbody>
</table>

The histograms for selected questions are plotted below, presenting the pattern of trainee responses in more detail. The first three histograms (Figures 5.14 through 5.16), for questions 2, 3, and 5, show the strength of trainees’ beliefs that the conversations, as modeled, were straightforward, realistic, and easy to understand.

The distribution of responses to Question 6 (Figure 5.17) reveals the range of trainees comfort/discomfort upon hearing their own voices. While there can be little doubt that hearing one’s own voice for the purpose of comparing it with an expert’s is a crucial learning experience, the range of trainee comfort with hearing one’s own voice exposes yet another opportunity for individualized instruction. The general lesson for tutor developers who intend to record trainee performance of any sort is that the review of one’s own performance by the trainee may provoke a range of affective reactions that the tutor should be prepared to deal with.
2) The conversation simulations were _________.

Figure 5.14
Trainee perception of LEAP's conversation straightforwardness

3) The conversation simulations were _________.

Figure 5.15
Trainee perception of LEAP's conversation realism
5) The dialogue voices were __________ to understand.

Figure 5.16
Trainee perception of LEAP's dialogue voice quality

6) I felt ___________ hearing my own voice.

Figure 5.17
Trainee comfort at hearing own voice
In sum, (and explicitly in Question 15, Figure 5.18) trainees reported LEAP to be highly usable. System slowness is a hardware issue that could easily be improved (in this particular case at least) by inserting a second microprocessor in the motherboard of the computer (a Sun Sparc 10) that was used as a server. \(^3\)

15) LEAP was generally _____________ to use.

---

Figure 5.18  
Trainee perception of LEAP's ease of use

Finally, a comparison between the Arizona group, who heard semi-professional actors playing customers and reps, and the Colorado groups, who did not, on these 18 questions yielded no significant differences, not even on the questions relating directly to conversations: 2, 3, and 5. Further comparisons between groups will be made in a later section.

---

\(^3\) In fact, we actually did this, but later discovered the second microprocessor had been surreptitiously removed early in the field trials.
5.3.4 Summary of Affective and Usability Results

Trainees who used LEAP reported they learned a lot, understood how to use LEAP, and were happy, confident, in control, interested, and pleased. On the issue of LEAP versus human role play partners, trainee responses spanned the spectrum of possible responses, showing a slight preference for the extremes. Trainees are content with the frequency LEAP expects them to respond to situation-action pairs, and believe their actual proficiencies are accurately reflected in the proficiency ratings LEAP assigns them.

Trainees reported LEAP to be highly usable. The system was straightforward, consistent and easy to use; command usage and error recovery were easy; feedback was especially useful. The screens were very clear, the system meets the needs of both experienced and inexperienced users. LEAP’s trainee proficiency ratings were likable and understandable. The conversations were straightforward, realistic, and easy to understand. Trainees did not mind hearing their own voices. LEAP’s response time was perceived as slow, a hardware issue that could easily be improved.

5.4 Instructional Value

This section reports trainee perceptions of the value of each of LEAP’s features in terms of its contribution to their learning\(^4\) (Figure 5.19). Data were acquired with The Instructional Value questionnaire administered to all trainees at the end of their day-long session with LEAP.

---

\(^4\) Results are reported for the last three groups of trainees to use LEAP. An interview format was attempted for the first two groups. Analysis of the responses revealed little of value; interviewers tended to write cryptic notes, and questions were often misinterpreted either by the trainees or the interviewers.
The Instructional Value questionnaire is composed of semantic differential questions, similar to those described in the previous section, together with a number of open-ended questions. The responses to questions in semantic differential form were plotted on histograms. The mean of the responses was calculated. A $t$-test was performed comparing the mean to a 'noncommittal' value of zero. For some questions in this section, a variation of the semantic differential is used. For example, a question that asks about how much of something, has no meaningful opposite. In these cases, a scale of 0 to 6 was used, and no significance test was calculated.

The responses to semantic differential questions are presented below as a mean of the trainees’ responses, or, if interest merits, a histogram illustrating the distribution of responses. The responses to verbal questions are summarized in paragraph form.

The results are presented for each of LEAP’s features, as viewed from the trainees’ perspective. The first section presents trainee opinion regarding LEAP’s general instructional value and the value of Top Level features. The next section probes the instructional value of the Rehearse Conversation features in detail. Finally the value of Examine Contact Flow and the Guide to VMS are presented.
5.4.1 Top Level And Overall

This section presents the results of two broad questions, and the perceived value of LEAP’s top-level, major features, including the Recommend function, the Student Model, and the three main study methods: study the guide, practice conversations, and explore the domain.

To the question:

(1) Is there anything in particular that you would like us to know about LEAP?

Most respondents made a positive general comment, i.e.: “It's a good system. You've done a good job.” Three said more initial instruction was needed: “Need more instruction on how to use LEAP.”, and three mentioned some way to improve the system.

LEAP was a real confidence builder. At the end of the day of using LEAP, trainees were very confident they could do a genuine VMS Change Order (Figure 5.20).
Today you practiced VMS Change Orders:

How confident are you that you could do a real VMS Change Order now?

![Bar chart showing trainee confidence in doing a real change order.]

Mean: 4.25

Figure 5.20

Trainee confidence in doing a real change order

To the question:

Please comment on your response:

Trainees reported they were very confident they could do a genuine task, namely a real VMS Change Order, either because of the information they found in LEAP or the practice LEAP provided them.

**Recommend.** The Recommend button is one of LEAP’s key design features. It is one of the ways LEAP implements the design philosophy of providing intelligent tutoring without imposing its intelligence on trainees. The histogram (Figure 5.21) shows that no trainee made the mean response to Recommend. Instead responses were of two camps; trainees either found Recommend rather useful for learning (a slight majority) or they did not find...
it at all useful, depending on whether they wanted to be told which topic and exercise to study next or not. This dichotomous reaction to Recommend illustrates differences in learning styles among trainees, and points up the potential value of flexible instructional approaches in any learning situation.

(3) On the top level screen there is a Recommend button. When you click on it LEAP will recommend a topic and a study method, and select a conversation for you. How useful was the Recommend button for learning VMS?

![Bar chart showing responses to Recommend button](image)

Mean: 1.25
Positive deviation from "noncommittal" rating of "0": Significant
P < .0021

Figure 5.21

Instructional value of the feature: Recommend

Tables of contents: LEAP has two tables of contents, one for topics and one for conversations. Taken together, the tables of contents provide trainees with a picture of the scope of the training materials in LEAP as well as direct access to the training materials. Trainees rated the Topics table of contents
useful, \( (M = 1.50, p < .0002) \); and they rated the Conversations table of contents useful \( (M = 1.13, p < .0054) \). Each of the tables of contents also provides trainees access to their student model in the form of a Proficiency rating in Topics, and a Challenge rating in Conversations. Trainees found their Topic Proficiency rating useful \( (M = 1.25, p < .0021) \). They found their Conversation Challenge rating useful \( (M = 1.13, p < .0019) \). Histograms for these two ratings (Figures 5.22 and 5.23) show their concentration at the mean.

Your Proficiency rating on topics

![Histogram](image)

**Mean:** 1.25  
Positive deviation from "noncommittal" rating of "0": Significant  
P < .0021

Figure 5.22

*Instructional value of the feature: Topic Proficiency*
The next question (Figure 5.24 through 5.26) requested trainees rate the three main instruction methods (study, practice, and explore) for their value in learning VMS change orders. Trainees rated the methods: Rehearse Conversations, Guide to Voice Messaging, and Examine Contact Flow. The mean usefulness ratings were 1.69, 1.56, and 1.20 respectively. These ratings correspond only partially to the time trainees actually spent in these three modes. As will be discussed further in the section on logged data, trainees spent most of their time in Rehearse Conversations, and very little in either Guide to Voice Messaging, or Examine Contact Flow. Responses for all study methods were significant.

1 The modal response in all three cases was 2. In the case of explore, there was a secondary peak at 0. See the vertical bar graphs of the results.
As expected, trainees rated the Rehearse Conversation highest of the three, and trainees in fact spent most of their time in this mode.

The Guide was developed more to showcase the future possibilities of multi-media than to present invaluable content. Content was minimized because of the relatively high expense of developing multimedia, the need for frequent updating of information, and the desire to reduce duplication of material that was available in the classroom or in the reps’ Desk Reference. It appears trainees saw the Guide as valuable because of its gee-whiz value, not because of its content. Trainees actually spent very little time in the Guide; their usage data do not support the relatively high rating they give it.

Trainees rated Examine Contact Flow (Explore) the lowest and spent the least time using this study method. This result is disappointing in the sense that trainees did not seem to appreciate the unique value of Explore -- the ability to learn all the details of a single Topic in a concentrated form. There are two plausible explanations: First, trainees used LEAP for only one day and did not have a lot of experience with whole conversations, and the whole conversations provided something that Explore did not, a context. On the other hand it was necessary to go through a whole conversation to practice just a small portion of it -- something not necessary in Explore. In sum, trainees may have needed the context of whole conversations and may not yet have been ready for the more abstract, more topic-focused approach of Explore. Second, while the functionality of Explore was quite different from Practice, the Explore interface had the same appearance as the Practice interface, and students may have found this confusing.
(5) On the top level screen there are three Instruction Choices: Study the Guide, Rehearse Conversations, and Examine Contact Flow. How useful did you find each one in learning about VMS change orders?:

Guide to Voice Messaging

<table>
<thead>
<tr>
<th>Number of Responses</th>
<th>Completely obstructive</th>
<th>Neither useful nor obstructive</th>
<th>Extremely useful</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Mean: 1.56
Positive deviation from "noncommittal" rating of "0": Significant
P < .0001

Figure 5.24

Instructional value of the method: Guide to Voice Messaging
Rehearse Conversations

**Figure 5.25**

Instructional value of the method: Rehearse Conversations

Mean: 1.69
Positive deviation from "noncommittal" rating of "0": Significant
P < .0001
Examine Contact Flow

<table>
<thead>
<tr>
<th>Number of Responses</th>
<th>Completely obstructive</th>
<th>Neither useful nor obstructive</th>
<th>Extremely useful</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>-1</td>
<td>-2</td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td>-3</td>
<td>-3</td>
</tr>
</tbody>
</table>

Mean: 1.20
Positive deviation from "noncommittal" rating of "0": Significant
P < .0006

Figure 5.26

Instructional value of the method: Examine Contact Flow

5.4.1.1 Summary of Overall and Top Level Instructional Value

To summarize: This section presented the results of two broad questions, and the perceived value of LEAP’s top-level, major, features. Trainees had a positive overall response to LEAP and were very confident they could do a genuine task, after studying and practicing it in LEAP. Most trainees found Recommend rather useful for learning, although some did not. The tables of contents provided useful information, and the Student Model, in the form of Proficiency and Challenge ratings, was also useful. Trainees rated the three main instruction methods: Rehearse Conversations, Guide to Voice Messaging, and Examine Contact Flow, first, second, and third, respectively, unexpectedly reversing the instructional value of the latter two methods.
5.4.2 Rehearse Conversation

LEAP's central learning activity is practicing or rehearsing conversations. The questionnaire examined trainees' perceptions of instructional value in this activity in detail. This section reports first on the perceived instructional value of the three conversation study methods, then on the value of the two simulation windows, and finally on the end-of-conversation activities.

5.4.2.1 Rehearse Conversation: Overall

The first questions (Table 5.7) asked trainees to rate the values of three main activities in Rehearse Conversation mode: Observe, Focused Practice and Full Practice.

<table>
<thead>
<tr>
<th>Question</th>
<th>M</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>How useful was Observe for learning VMS?</td>
<td>2.19</td>
<td>.0001</td>
</tr>
<tr>
<td>How useful was Focused Practice for learning VMS?</td>
<td>1.88</td>
<td>.0001</td>
</tr>
<tr>
<td>How useful was Full Practice for learning VMS?</td>
<td>1.75</td>
<td>.0003</td>
</tr>
</tbody>
</table>

These ratings are high and in the expected sequence. The histograms of responses to these questions (Figures 5.27 through 5.29) peak at the 'Extremely useful' end of the scale. Trainees believed the most valuable mode was Observe, the next most valuable was Focused Practice, and third most valu-
able was Full Practice. Trainees appreciated the value of observing an expert. They seem to understand that it is also necessary to practice a skill to acquire it, and they seem to understand that Full Practice was not as productive as Focused Practice. Finally, the responses to the follow-up questions show trainees observed themselves acquiring knowledge and skills over the course of the day.

(7) In the Rehearse Conversation there are three ways to rehearse a conversation (to practice a role play): Observe, Focused Practice, and Full Practice: how useful was each of these for learning VMS?

Observe

![Bar chart showing the number of responses for the observe method]

- Mean: 2.19
- Positive deviation from "noncommittal" rating of "0": Significant
  \[P < .0001\]

**Figure 5.27**

Instructional value of the method: Observe
Focused Practice

Mean: 1.88
Positive deviation from "noncommittal" rating of "0": Significant
P < .0001

Figure 5.28

Instructional value of the method: Focused Practice
To the question:

(8) Did Observing a conversation a few times prepare you adequately to do it yourself? (Yes/No)?

If you answered No, what more would you need?

All but one trainee reported that observing a conversation a few times was adequate preparation for doing it oneself. One trainee mentioned that she preferred to go straight into Full Practice. Another mentioned that Observing (i.e., listening to) a conversation is a good method of learning for those with reading disabilities.
To the question:

(9) In Focused Practice, LEAP would do parts of the conversation for you, what was your reaction to that?

Most trainees reported a positive reaction. "Great" some responded to skim "[It] reinforced what I have already learned," others responded to scaffold "... [it] lead (sic) into the full practice option."

To the question:

(10) As you rehearsed conversations (practiced role plays) over the course of the day, what did you observe about your ability to do VMS change orders?

Most trainees reported observing their knowledge increase. A few reported their confidence increased, and a couple reported areas where they still needed improvement.

5.4.2.2 Conversation and SONAR Windows

Trainees spent most of their Rehearse Conversation time in the Conversation and SONAR simulation windows. In both these windows LEAP presents situations and trainees take actions. In both windows LEAP provides feedback and a graduated series of hints. This section examines trainee perception of the instructional value of these features.

In the Conversation window, LEAP occasionally presented the situation visually, in text, (i.e., when the situation was one that experienced reps would recall and there was no overt clue) though usually it did not, since the situation was normally verbal speech (i.e., the customer made a statement). Trainees reported some difficulty with the changing situation presentation \( (M = 3.47 \text{ on a scale of } 0 - 6) \) (Figure 5.30). During the field trial, however, no observers noted any trainee having difficulty with the situation presentation.

---

2 As mentioned, for some questions in this section, a variation of the semantic differential was used for questions that ask about how much of something, and have no meaningful opposite. In these cases, a scale of 0 to 6 was used, and no significance test was calculated.
Also in the Conversation window there were five ways of responding to a situation (Figure 5.31). Most trainees reported no difficulty in dealing with the different response types. ($M = 1.71$ on a scale of 0 -- 6). However, during the field trial, observers noted that some trainees did have difficulty with these multiple response types and this part of LEAP needs to be simplified.

The SONAR window contained a simulated SONAR database. While some trainees thought the SONAR simulation was fine, many trainees reported that the simulation had shortcomings which interfered with the learning process (Question 15). Given LEAP's development history -- the WITS software could not provide the needed functionality; its replacement, VAPS, a simulation package, was found not to be appropriate either; so at the last minute we realized we had to create the simulation in-house ourselves -- what LEAP did have was an outstanding achievement by the programmer.
In the Conversation Window, sometimes LEAP presented the situation and sometimes it did not.

Did you find the changing situation presentation confusing?

![Bar Chart]

**Figure 5.30**

Is the changing situation presentation confusing?

Please comment about how LEAP presents the situation:

Trainees reported that sometimes they did not know what was expected of them.
(14) In the Conversation Window, LEAP presented a number of response types:

- Click something
- Record something (with Start recording and Stop recording)
- Pick multiple choice (after recording something)
- Pick multiple choice (without recording something)
- Do something in SONAR instead

Was dealing with the different response types confusing?

![Bar chart](image)

Mean: 1.71

**Figure 5.31**

Is the number of response types confusing?

To the question:

(15) What do you want us to know about your experience using the SONAR simulation?

Most trainees mentioned ways in which the SONAR simulation needed improvement. (It did.) Some trainees thought it was fine as-is.
Feedback: LEAP provided feedback in both the Conversation window and the SONAR window. Because the Conversation window presented a list of responses, simply informing the student that her answer was incorrect was enough to enable the student to eventually select the correct answer. In the SONAR window the feedback was more specific. It reported the value it expected and the field it expected the value to be typed into, as well as the value the trainee typed and the field she had typed it into. Trainees rated the conversation feedback 3.44 (Figure 5.32), and the SONAR feedback 3.69 (Figure 5.33), both on a scale of 0 -- 6.

(16) After every Situation and Response, LEAP would continue ahead if you selected the correct response, or give you some feedback if you missed it. If you clicked a wrong multiple choice response, you got some feedback: How would you characterize that feedback?

![Bar Chart]

Mean: 3.44

Figure 5.32

Instructional value of conversation feedback
If you made an error in SONAR, you got some feedback: How would you characterize that feedback?

![Bar chart showing the mean feedback rating as 3.69.](image)

**Figure 5.33**

*Instructional value of SONAR feedback*

**Hints:** Two questions asked about the value of hints (Figures 5.34 and 5.35). Trainees found the conversation or what-to-say hints ($M = 2.40$) more valuable than the SONAR or what-to-do hints ($M = 1.80$), probably because it is more difficult to generate the appropriate verbal action than it is to respond to the contextual clues provided by the database screens.
If you didn’t know what to do, you could ask LEAP for a hint. There are two kinds of hint, a hint for what to say next, and a hint for what to do next in SONAR, how useful was each kind of hint for learning VMS?

Hints for what to say next

<table>
<thead>
<tr>
<th>Number of Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

**Mean:** 2.40

Positive deviation from "noncommittal" rating of "0": Significant

*P < .0001*

**Figure 5.34**

Instructional value of conversation hint
5.4.2.2.1 Summary of Conversation and SONAR Windows

Rehearse Conversation, LEAP’s predominant study method consists of two main simulation windows: Conversation and SONAR. In these windows, LEAP presents situation-action pairs, feedback and hints. Trainees rated Observe, Focused Practice and Full Practice highly and in the expected sequence. Trainees believed Observing a conversation was adequate preparation for doing it oneself. Most trainees reported observing their own knowledge increase. Although field observers noted that some trainees had difficulty with multiple response types, most trainees reported none. Many trainees reported that SONAR simulation shortcomings interfered with the learning process. Trainees rated the conversation feedback 3.44 and the SONAR feed-
Trainees found the conversation or what-to-say hints ($M = 2.40$) more valuable than the SONAR or what-to-do hints ($M = 1.80$).

5.4.2.3 End-of-conversation Windows

When a trainee completes a conversation, several events take place. The Conversation window clears, and buttons for the trainee to see a Video Conversation Summary, to Repeat Conversation, or Exit (return to Top-level) appear. Also, the Conversation History window opens, displaying a script of the entire conversation, including the trainee's (corrected) responses. This section presents the instructional values trainees reported for these windows.

**Conversation History Window:** At the end of a conversation, the Conversation History window appears. The window has two control buttons: the Talk-Only button causes the window to display the conversation script; the Talk & SONAR button makes the window display both the script and the SONAR actions. Trainees rated the Conversation History highly, ($M = 1.56$) (Figure 5.36).

The conversation history window could present two views of the conversation: Talk-only and Talk & SONAR. A few trainees used one view exclusively, but most reported using both equally (Figure 5.37).

In the Conversation History window trainees could play back any selected line of the conversation, and, for the action lines, compare the recording of themselves with that of the original speaker. Trainees found this feature quite useful: ($M = 4.53$ on a scale of 0 -- 6) (Figure 5.38), and nearly all trainees mentioned ways that making this comparison helped them to improve their spoken responses. Summarized responses to the Conversation History questions are below:
At the end of a conversation, the Conversation History window appeared.

Overall, how would you rate the value of the conversation history for learning VMS change orders.

![Bar chart showing the instructional value of the conversation history]

**Number of Responses**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completely obstrusive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither useful nor obstructive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extremely useful</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Mean: 1.56

Positive deviation from "noncommittal" rating of "0": Significant

P < .0002

**Figure 5.36**

Instructional value of the conversation history
(18) In the Conversation History you could select between two views: Talk-only, and Talk and SONAR:

Please indicate how much you used each view:

![Bar chart showing number of responses for different views](chart.png)

**Mean:** 0.38  
Positive deviation from "noncommittal" rating of "0": Non-Significant  
P < .2068

**Figure 5.37**

Instructional value of the Talk-only and Talk & SONAR views
In the Conversation History you could play back any selected line of the conversation, and compare yourself with the original speaker. How useful was this feature for learning VMS?

![Number of Responses](image)

<table>
<thead>
<tr>
<th>Number of Responses</th>
<th>Mean: 4.53</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Not at all useful</td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Extremely useful</td>
</tr>
</tbody>
</table>

Figure 5.38

Instructional value of commands: Hear Original and Hear Myself

To the question:

What did you learn by comparing your responses with the original speaker's responses?

Nearly all the trainees mentioned ways they could improve their responses, i.e., "I needed to be more clear and to respond the customer's question." "... I needed to learn my products better." "How to reply with less technical detail."

Other End-of-Conversation features: At the end of a conversation, besides the History, trainees could see a Video Conversation Summary, and they could Repeat the conversation.
Trainees rated the Video Conversation Summary just slightly better than 'neither useful nor obstructive.' \(M = .50, ns\) (Figure 5.39). Their written response to the follow up question confirmed their ratings.

Trainees rated the Repeat Conversation feature fairly highly: \(M = .87, p < .0159\) (Figure 5.40) which corresponded with the frequency they used the feature. Of course, trainees could have returned to the top-level and selected the same conversation again, but making the Repeat Conversation option explicit encourages its election.

(20) At the end of a conversation, you could play a Video Conversation Summary. How useful was this feature for learning VMS?

![Figure 5.39](image)

Mean: 0.50
Positive deviation from "noncommittal" rating of "0": Non-Significant
\(P < .1585\)

Figure 5.39

Instructional value of the video conversation summary
To the question:

What did you learn by reviewing the video conversation summary?

Most trainees did not respond to this question, or said they did not use it, or that they learned nothing from it. A couple of trainees specifically complained about the quality e.g., “Squeaky sputtering - not helpful at all, too difficult to understand,” a couple said they learned something specific e.g., “How to respond to the customer.”

(21) At the end of a conversation, you could Repeat the conversation. How useful was this feature for learning VMS?

![Bar chart](image)

- Completely obstructive: 1 response
- Neither useful nor obstructive: 4 responses
- Extremely useful: 2 responses

Mean: 0.87

Positive deviation from "noncommittal" rating of "0": Significant

P < .0159

Figure 5.40

Instructional value of the command: Repeat Conversation
To the question:

What did you learn by repeating the conversation?

The majority of trainees did not respond, those who did, named specific kinds of things they learned, e.g., “How to rephrase it.” “Things I missed at first.”

Two questions referred to characteristics of practicing in the context of a complete conversation. Trainees do not object to doing a whole conversation, but neither do they find it particularly useful ($M = .53, ns$) (Figure 5.41). At the same time, they do not find the alternative of seeing summaries of the out-of-topic portions of the conversation particularly appealing (nor particularly off-putting) either ($M = .4, ns$) (Figure 5.42). Since in some cases of Focused Practice, trainees spent more time skimming and scaffolding than practicing, redesigning the tutor so that it would summarize the pre- and post- portions of the conversation in paragraph form appears to be an option. However, practicing in the context of a whole conversation (versus practicing in the context of summarized versions of the out-of-topic portions of it) permits the review of selected SA pairs. Trainees did not seem to mind reviewing, but they also did not seem to be aware of its instructional value.
The conversations in Rehearse are always entire conversations, not just parts.

What is the effect of doing an entire conversation from start to finish even if you are concentrating on just one part of it - i.e., just one topic in the conversation?

![Graph showing the distribution of responses on a scale from -3 to 3. The scale labels are: Completely obstructive, Neither useful nor obstructive, and Extremely useful. The bars show the number of responses for each category.]

Mean: 0.53
Positive deviation from "noncommittal" rating of "0": Non-Significant
P < .1504

Figure 5.41

Instructional value of doing an entire conversation
What if, instead of a whole conversation, you just got the part you were concentrating on, with a paragraph summarizing the parts that happened before and after? Would the experience be:

![Bar Chart]

Mean: -0.40  
Positive deviation from "noncommittal" rating of "0": Non-Significant  
P < .2025

**Figure 5.42**

Instructional value of doing an abbreviated conversation

To the questions:

(23) During Focused Practice LEAP would occasionally ask you to do parts of a conversation that you had already practiced many times.

What was your reaction to that?

The responses were about equally split between positive, neutral, and negative, e.g., "To me it was extremely helpful" "It was OK," "Boring."

What was the effect of that on your learning?

Many trainees did not respond or made neutral comments "No major impact," some made positive comments "It polished my skills" a few made negative comments "Mind wandered."
To summarize: Trainees rated the Conversation History highly; they reported using Talk-only and Talk & SONAR equally. Trainees found the play-back-and-compare feature quite useful and believed it helped them to improve their spoken responses. Trainees were neutral regarding the Video Conversation Summary, but they liked the Repeat Conversation feature. Trainees were neutral on the topics of whole-conversation practice, summaries of out-of-topic portions, and review.

5.4.2.4 Summary of Rehearse Conversation

Trainees rated the three main activities in Rehearse Conversation mode: Observe, Focused Practice, and Full Practice as “Extremely useful” and in that sequence. Trainees were able to observe themselves acquiring knowledge and skills over the course of the day. Trainees rated Feedback, Hints, Conversation History, and Repeat Conversation highly; they were neutral on the Video Conversation Summaries.

5.4.3 Examine Contact Flow

This section reports trainee perceptions of the instructional value of the Examine Contact Flow study method. Questions address the general utility of the method, and the value of four commands, two of which were unique to Examine Contact Flow.

Overall, how useful was the ability to Examine Contact Flow for learning VMS change orders? \( M = 1.31, p < .0023 \) (Figure 5.43). A similar version of this question is asked near the beginning of the questionnaire (Figure 5.26), so the responses can be compared for consistency. The mean of the first response was 1.20, quite consistent.

Trainees could reverse direction and Back Up through the contact flow in order to repeat something, or to take another branch of the possible con-
versations. Trainees rated this feature $M = 3.67$ (on a scale of 0 -- 6) (Figure 5.44).

Trainees could turn the Audio On or Off, reading the statements of expert reps and either listening to them or not. The former provided authentic examples, the latter, speed. Trainees rated this on/off switch $M = 3.62$ (on a scale of 0 -- 6) (Figure 5.45).

Trainees could select Observe or (Full or Focused) Practice. Trainees indicated that they used Practice more than they used Observe ($M = .79$, $p < .0256$); however, the modal trainee used these features equally (Figure 5.46).

At the end of a topic's contact flow, trainees could Repeat the Flow. Trainees rated this feature ($M = 3.71$ on a scale of 0 -- 6) (Figure 5.47). This feature was not expected to be particularly valuable, since most topics are relatively shallow, about six SA pairs deep, and the Back Up command is also available in this mode (as is the option to re-select the topic at the top-level).

In the History of Examine Contact Flow, trainees could hear a particular SA pair response spoken by all the reps in all the conversations recorded in LEAP. Trainees rated this feature ($M = 4.21$ on a scale of 0 -- 6) (Figure 5.48).

To summarize Examine Contact Flow: overall trainees rated the method as important for learning. Trainees indicated that they valued all the features: Back Up, Audio On/Off, Repeat, and Hear All, about equally. Also, trainees Practiced the topics in Examine Contact Flow somewhat more than they Observed them. Histograms of the responses follow:
Overall, how useful was the ability to Examine Contact Flow for learning VMS change orders?

Figure 5.43

Instructional value of Examine Contact Flow
(25) In Examine Contact Flow you could Back Up through the contact flow: How useful was this feature for learning VMS?

![Bar chart showing the instructional value of the command: Back Up]

Mean: 3.67

Figure 5.44

Instructional value of the command: Back Up
(26) In Examine Contact Flow you could turn the Audio On or Off: How useful was this feature for learning VMS?

![Bar chart showing the instructional value of the Audio On/Off feature. The x-axis represents the level of usefulness ranging from Not at all useful to Extremely useful, and the y-axis represents the number of responses. The mean rating is 3.62.]

Mean: 3.62

Figure 5.45

Instructional value of the command: Audio On/Off
In Examine Contact Flow you could choose between Observe and Practice, Please indicate how much you used each method:

![Bar chart showing the preference for Observe versus Practice.](image)

- **Mean:** 0.79
- **Positive deviation from "noncommittal" rating of "0":** Significant
- **P < .0256**

**Figure 5.46**

Preference for Observe versus Practice
End of Contact Flow Windows

(28) At the end of a Contact Flow you could Repeat the Flow: How useful was this feature for learning VMS?

![Bar Chart]

Mean: 3.71

Figure 5.47

Instructional value of the command: Repeat the Contact Flow
(29) In the History of Examine Contact Flow you could hear the same response spoken by different expert reps in different circumstances. How helpful was this feature?

![Bar chart](image)

Mean: 4.21

Figure 5.48

Instructional value of hearing same response by different reps

5.4.4 Guide to VMS

This section reports trainee perceptions of the instructional value of the Guide to VMS. Several questions asked about details of Guide use:

- The modal trainee looked at all the information in the Guide.

- Trainees rated the printed information in the guide more useful than obstructive ($M = 1.40, p < .0001$).

- On the issue of whether they preferred to read the information or hear it read to them, the response was bimodal; although many did not care either way, more than half preferred to have it read to them.

- The video-based information was seen as neither useful nor obstructive. ($M = .40, ns$)
Finally, trainees were asked whether they used their Desk Reference during the day, and if so, what they used it for. Half the trainees used the desk reference, half did not. Trainees mentioned a variety of items, primarily VMS availability, that they looked up in the desk reference. Trainees were expected to use their desk references as necessary during conversations. Proper use of the desk reference was taught by the instructor (not by LEAP).

In sum, half the trainees viewed all or most of the information in the Guide; surprisingly, most preferred to have the text read to them, but they were not overly impressed with the video advertisements promoting the features of VMS.

5.4.5 Summary of Instructional Value

This section reported trainee perceptions of the value of LEAP’s features in terms of the contribution of each feature to their learning, as recorded in the Instructional Value Questionnaire. Trainees were very confident they could do an actual job task, namely a Voice Messaging change order, after studying and practicing it in LEAP. At the top level, most trainees found the Recommend function and their Student Model quite helpful to their learning. Trainees rated the Rehearse Conversation and Guide to Voice Messaging study methods highly, and rated Examine Contact Flow much lower, although significantly higher than a noncommittal rating of zero. In Rehearse Conversation, trainees rated the three activities: Observe, Focused Practice, and Full Practice as “Extremely useful.” Trainees could see themselves learning. Trainees rated Feedback, Hints, Conversation History, and Repeat Conversation highly; they were neutral on the Video Conversation Summaries. Trainees rated Examine Contact Flow as important for learning, and they valued its features: Back Up, Audio On/Off, Repeat, and Hear All. In the Guide, most trainees preferred to have the text read to them.
5.5 Command Usage

5.5.1 Introduction

This section presents data describing how trainees actually used LEAP. It summarizes the usage data captured by LEAP’s logging function. The logging function captured trainees’ commands, actions, and student model, and time-stamped them. An example of logged data and a summary of the logged data is displayed in Appendix B, Logged Data: Summary and Sample. For example, the log recorded each time each trainee entered and exited the Guide to Voice Messaging. From this logged data, various other data could be calculated, for example: the average (mean) number of times trainees entered the Guide, how long, on average, they stayed, and the total amount of time (mean) they stayed in the Guide.

The first section presents overall usage data regarding the three major parts of LEAP: the Guide, Rehearse Conversation, and Explore Contact Flow, and total study time.

The second section describes trainees’ accomplishments by several measures: number of SA pairs tried, scores on those SA pairs, and the total knowledge attained. This section also examines how much trainees learn versus how well they learn it, as well as the effect of time on task on each of these measures.

The third section summarizes trainee use of certain commands: Recommend, Help (top-level), Help (conversation-level), Hints, Change Instruction Style, Repeat Conversation, Video Conversation Summary, and Hear Original/Myself.

The fourth section examines trainee use of various methods for selecting conversations, the use of individual commands is considered as a predic-
tor of achievement, and the outcomes of self-directed versus tutor-directed learning are examined.

5.5.2 Overall Usage Data

This section presents data describing how much time trainees actually spent in LEAP and how many exercises they tried. It also presents overall usage data regarding the three major parts of LEAP (the Guide, Rehearse Conversation, and Explore Contact Flow) how many times they entered each part, how long they stayed each time, and the total time in the part.

LEAP was field trialed in five sites for ‘one day’ each. The actual amount of time trainees spent in active use of LEAP varied from three to five hours. The mean time spent actively using LEAP was 239 minutes, or four hours. On average, trainees tried eleven different conversation exercises.

On average, trainees entered the Guide four times, staying a mean of 4.2 minutes each time. The mean of the total time trainees spent in the guide was 12.9 minutes.

On average, trainees selected Rehearse Conversation 20.3 times, staying a mean of 10.7 minutes each time. The mean of the total time trainees spent Rehearsing conversations was 207 minutes.

On average, trainees selected Explore Contact Flow 5.4 times, staying a mean of 4.3 minutes each time. The mean of the total time trainees spent Exploring Contact Flow was 18.6 minutes. The next few pages (Figures 5.49 - 5.59 and Table 5.8) display more details on these overall usage data.
Figure 5.49

Study time, total

Figure 5.50

Conversations tried one or more times
Figure 5.51
Guide, times entered

Figure 5.52
Guide, average time in

236
Figure 5.53
Guide, total time in

Figure 5.54
Rehearse, times entered
Figure 5.55

Rehearse, average time in

Figure 5.56

Rehearse, total time in
Figure 5.57

Explore, times entered

Figure 5.58

Explore, average time in
Figure 5.59

Explore, total time
Table 5.8
Overall Usage Data

<table>
<thead>
<tr>
<th>Measure of Usage:</th>
<th>M</th>
<th>SD</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conversations tried</td>
<td>11</td>
<td>2.8</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>(one or more times)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Study time, total minutes in:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Guide</td>
<td>12.9</td>
<td>8.6</td>
<td>1</td>
<td>38</td>
</tr>
<tr>
<td>Rehearse</td>
<td>207.5</td>
<td>33.2</td>
<td>142</td>
<td>263</td>
</tr>
<tr>
<td>Explore</td>
<td>18.6</td>
<td>18.7</td>
<td>0</td>
<td>70</td>
</tr>
<tr>
<td>Guide + Rehearse + Explore</td>
<td>238.9</td>
<td>31.9</td>
<td>182</td>
<td>303</td>
</tr>
<tr>
<td>Times trainees entered:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Guide</td>
<td>4</td>
<td>2.6</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>Rehearse</td>
<td>20.3</td>
<td>5.0</td>
<td>10</td>
<td>31</td>
</tr>
<tr>
<td>Explore</td>
<td>5.4</td>
<td>6.6</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>Study time, minutes per entry:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Guide</td>
<td>4.2</td>
<td>3.6</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>Rehearse</td>
<td>10.7</td>
<td>2.8</td>
<td>7</td>
<td>18</td>
</tr>
<tr>
<td>Explore</td>
<td>4.3</td>
<td>3.1</td>
<td>0</td>
<td>11</td>
</tr>
</tbody>
</table>

5.5.3 Accomplishments

This section describes trainees’ accomplishments by several measures: number of SA pairs tried, scores on those SA pairs, and the total knowledge attained. This section also examines how much trainees learn versus how
well they learn it, as well as the effect of time on task on each of these measures.

In the course of their 1-day use of LEAP, trainees tried a mean of 50.4% of the SA pairs (Figure 5.60, Table 5.10). For the SA pairs they tried, they attained a mean score of 55.5 (of a maximum score of 100) (Figure 5.61, Table 5.10). As described in the section on student modeling, this score is not a percentage score, but a performance indicator based on three weighted factors: the number of times trainees have seen the SA pair, their average score on the SA pair, and the number of times they have consecutively performed the Action correctly (or incorrectly).

The two values, Percent of SA pairs Tried, and Score, were multiplied to obtain a measure of overall knowledge acquisition for each trainee. The mean Knowledge acquired was 27.6 (Figure 5.62, Table 5.10). (Note: since the Score is a composite, non-linear number, it can reasonably be claimed that a score of .4 on 20% of the material is not equivalent to a score of .2 on 40% of the material, though the Knowledge measure is identical in both cases. This nonequivalence is more likely to be important when scores are low -- below .4 -- and less significant when scores are above .4)

---

1 As mentioned in the section on Topic Proficiency, a score of 30 is Almost!, 40 is Good, and 75 is Excellent.

2 It is important to keep in mind that the Knowledge value reflects LEAP’s understanding of the trainee’s knowledge, which is not necessarily the trainee’s actual knowledge. For example, if the trainee happens to be an expert in something before s/he begins using LEAP, it will take LEAP a little while to figure that out; LEAP’s user model will lag the user’s actual knowledge for that time. On the other hand, if the user starts with little or no knowledge of the domain (presumably the nominal case) LEAP will acquire knowledge about the user as the user acquires knowledge about the domain.
Figure 5.60
Model, percent tried

Figure 5.61
Model, score

243
The next two figures are among the most informative displays of trainee learning in LEAP. The first (Figure 5.63) plots, for each trainee, how much each trainee learned versus how well he or she had learned it by the end of the day. The next (Figure 5.64) shows the distribution of the rate of learning in LEAP, where rate of learning is calculated by dividing the total amount of knowledge acquired by the total amount of time spent in Rehearse mode, or Knowledge/Rehearse-minute.

The scatter graph in Figure 5.63 plots the final Score versus SA pairs Tried (%) for all trainees. The trainees attained widely varying results though, as mentioned above, the results are more or less normally distributed. A score of 40 for a topic was labeled as ‘Good.’ Most trainees attained Scores of Good or Excellent for one topic before going on to the next. Trainees’ overall Score will fall whenever they try new material for the first time, and the more new ma-
terial they try the further their score will fall, thus a trainee’s final location on the scatter graph below says nothing about his or her path of progress to that position. For example, the trainee at 60, 25 probably knows some of the material very well,

![Scatter graph](image)

**Figure 5.63**

**Vms knowledge as score versus percent of SA pairs tried**

How fast did trainees learn and how did this learning rate differ among trainees? Trainees spent most of their time in Rehearse Conversation mode, and acquired most of their Knowledge there. To capture the rate of Knowledge acquisition, each trainee’s Knowledge score was divided by the amount of time they spent rehearsing conversations. Trainees’ Knowledge Acquisition Rates were then plotted on a histogram (Figure 5.64). This histogram of trainee Knowledge Acquisition Rates (Knowledge per minute of Rehearse time) is a bell-shaped curve, centered on the mean, indicating that
no matter how trainees varied in their use of LEAP, they tended to learn at
about the same rate. No one subgroup learned at a rate notably different from
any other. The histogram is bell-shaped, not uniform, bimodal, multimodal,
or heavily skewed toward either end. This result supports the claim of practi-
cality by illustrating LEAP's ability to provide instruction to a variety of
trainees.

![Image of a histogram](image)

**Figure 5.64**

**Knowledge per minute of Rehearse time**

Trainee confidence (Figure 5.20) is not an indicator of learning. There is
no significant correlation between students' confidence in their ability to per-
form the task on the job and any of the measures of their learning (Table 5.9).
Table 5.9
Trainee Confidence as a Predictor of Learning

<table>
<thead>
<tr>
<th>Trainee confidence versus results:</th>
<th>Correlation</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confidence for doing a real change order vs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of SA pairs tried</td>
<td>.086</td>
<td>ns</td>
</tr>
<tr>
<td>Score on SA pairs</td>
<td>-.232</td>
<td>ns</td>
</tr>
<tr>
<td>Performance</td>
<td>-.049</td>
<td>ns</td>
</tr>
<tr>
<td>Rate of learning</td>
<td>-.25</td>
<td>ns</td>
</tr>
</tbody>
</table>

For example, some students who were only moderately confident had the highest Performance scores, while some students who were most highly confident had the lowest Performance scores (Figure 5.65).

Figure 5.65
Trainee confidence as a predictor of performance
Knowledge acquisition rate for time spent in Guide + Rehearse + Explore is only slightly lower, on average, than for time spent in Rehearse. However, knowledge acquired during Rehearse is fairly normally distributed. In contrast, knowledge acquisition rate for time spent in Guide + Rehearse + Explore is not normally distributed (Figure 5.66), and for 6 trainees, drops to nearly half that for time spent in Rehearse. This drop is partially an artifact of the design of the student model, which only credits the trainee when she sees or tries an SA pair. Excepting this artifact, the main observation is that time spent wandering around in LEAP is time wasted.

![Figure 5.66](image)

**Figure 5.66**

Knowledge per total minutes in Guide + Rehearse + Explore

Let us turn now to the effect of Rehearsal time on the two components of trainee Knowledge: Average Score and Percent of SA Pairs Tried.
In Figure 5.67 we see that the amount of time trainees spend in Rehearse does not significantly influence their final score (p < .46, ns). This is because trainees tend to remain on a topic until they have attained a score for the topic of ‘Good’ before moving on to the next topic. Although the overall score will temporarily drop when a new topic is taken up, repeating exercises that address the topic will raise the score, thus Scores remains more or less constant throughout the trainees’ time in Rehearse.

If trainees had had a longer time to practice, or LEAP had contained less material, the picture might be different. LEAP’s spiral curriculum algorithm recommends that trainees learn each topic to a level of ‘Good’, then revisit all topics, studying to attain a level of ‘Excellent.’ With the spiral curriculum, we might expect to see a jump in scores after trainees had competed the first pass through the curriculum.

![Figure 5.67](image_url)

**Figure 5.67**

*Rehearse, total time versus score*
In Figure 5.68 we see that the time spent in Rehearse is a significant predictor of the amount of material learned ($p < .008$). In general, trainees who studied longer tried more material, and as the previous chart showed, trainees learn all material to about the same degree.

![Figure 5.68](image)

**Figure 5.68**

Rehearse, total time versus percent of SA pairs tried

**Summary:** LEAP contained an abundance of learning material. During the one-day trial, trainees tried about half of it, practicing until they attained a score of about half the maximum. Trainees' knowledge acquisition rate (Knowledge/Rehearse-minute) appears to be normally distributed, implying that LEAP accommodates diverse learning styles. Rehearse Conversation is LEAP's most productive study method; time spent wandering around in other parts of LEAP is time wasted. Trainees tend to Rehearse a topic until
they have attained a ‘Good’ score. Time spent in Rehearse Conversation is a significant predictor of the amount of material learned.

Table 5.10
Accomplishments

<table>
<thead>
<tr>
<th>Measure of Usage</th>
<th>M</th>
<th>SD</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model, percent tried</td>
<td>50.4</td>
<td>14.0</td>
<td>22</td>
<td>75</td>
</tr>
<tr>
<td>Model, score</td>
<td>55.5</td>
<td>9.8</td>
<td>27</td>
<td>71</td>
</tr>
<tr>
<td>VMS Knowledge, Percent</td>
<td>27.6</td>
<td>8.0</td>
<td>12.5</td>
<td>43.3</td>
</tr>
<tr>
<td>Knowledge per minute of Rehearse time</td>
<td>.133</td>
<td>.035</td>
<td>.072</td>
<td>.222</td>
</tr>
<tr>
<td>Knowledge per Total time in Guide + Rehearse + Explore</td>
<td>.116</td>
<td>.032</td>
<td>.067</td>
<td>.174</td>
</tr>
</tbody>
</table>

5.5.4 Command Usage, as Logged

This section summarizes trainee use of certain commands: Recommend, Help (top-level), Help (conversation-level), Hints, Change Instruction Style, Repeat Conversation, Video Conversation Summary, and Hear Original/Myself.

Trainees used the Recommend function a mean of 7.5 times; the modal usage was 2 to 4 times (Figure 5.69, Table 5.12). The details of the Recommend algorithm are described elsewhere. When a trainee clicked Recommend, LEAP would recommend a topic, an exercise, and a study method for the exercise, which the trainee could then accept or reject. The log
function did not record the extent to which trainees accepted or rejected the Recommend function.

LEAP’s top-level Help describes trainees’ options at the top level of LEAP. Trainees clicked on top-level Help a mean of 1.5 times (Figure 5.70, Table 5.12). Practice-level Help describes trainees’ options at the practice (Rehearse Conversation and Examine Contact Flow) level of LEAP. Trainees clicked on practice-level Help a mean of 2.5 times (Figure 5.71, Table 5.12). Trainees did not use Help much. This corroborates their responses in the Usability questionnaire that LEAP was generally easy to use, and that initial instruction on how to use LEAP was sufficient.
Figure 5.70

Top level Help

Figure 5.71

Practice level Help
While Help tells trainees how to use LEAP, Hints tells trainees what to do next in terms of VMS change orders. Trainees used Hints a mean of 32 times (Figure 5.72, Table 5.12).

LEAP always pre-selects an instruction style (Observe, Focused Practice, or Full Practice) for both the speaking and the SONAR sides of a conversation. Trainees could, however, change the instruction style for either side or both, as often as they liked, to make the conversation harder or easier as they felt necessary. Trainees changed the instruction style a mean of 21.1 times, just over once per conversation (Figure 5.73, Table 5.12).
At the end of a conversation, trainees could elect to repeat the conversation (Figure 5.74, Table 5.12). Trainees chose to repeat conversations a mean of 4.9 times; about 25% of the time they practiced a conversation, they were Repeating a conversation.
At the end of a conversation, trainees could choose to see a video-based Summary of the conversation’s salient points. The Summary served to reinforce the main activities of the exercise by describing them from an expert rep’s perspective. Trainees clicked on the Summary button an average of 4.8 times; or about 44% of the conversations they studied (Figure 5.75, Table 5.12).
At the end of a conversation (or earlier if they chose), trainees could review their performance in the Conversation History window, selecting any line in the conversation and listening either to the original expert rep, or to themselves, or both. Trainees chose to hear the Original a mean of 17.1 times (Figure 5.76, Table 5.12) and to hear themselves a mean of 25.5 times (Figure 5.77, Table 5.12). Many trainees elected to hear both, thereby comparing and contrasting themselves with the expert rep (p < .012), as LEAP was designed for them to do (Figure 5.78). Also of interest is the range of trainee preferences; for example, those who chose to Hear Original about 25 times, chose to Hear Myself from 0 to 75 times.
Figure 5.76

Hear Original

Figure 5.77

Hear Myself
Trainee perception of the value of LEAP’s features for learning was not a predictor of their use of the features. There is no significant correlation between students’ reported perceptions of commands as valuable for learning and students’ actual use of the commands in their learning. (Table 5.11)
Table 5.11
Commands: Perceived Value and Actual Usage

<table>
<thead>
<tr>
<th>Command Value versus Usage:</th>
<th>Correlation</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommend value vs. Recommend use</td>
<td>.344</td>
<td>ns</td>
</tr>
<tr>
<td>Hint value vs. Hint use</td>
<td>.169</td>
<td>ns</td>
</tr>
<tr>
<td>Video Summary value vs. Video Summary use</td>
<td>.415</td>
<td>ns</td>
</tr>
<tr>
<td>Repeat Conversation value vs. Repeat Conversation use</td>
<td>.101</td>
<td>ns</td>
</tr>
<tr>
<td>Hear Original/Self value vs. Hear Original/Self use</td>
<td>.084</td>
<td>ns</td>
</tr>
</tbody>
</table>

For example, trainee value and usage of the Recommend command is presented in Figure 5.79. Some trainees who valued Recommend highly used it a lot. Others who rated Recommend highly used it as little as those who saw it as having no value.
Summary: This section described trainee use of certain commands. Trainees varied widely in their use of the Recommend function; many trainees used it relatively infrequently. Trainees needed very little Help in using LEAP, but used Hints more than once per exercise. They also changed instruction style more than once per exercise, and chose to Repeat conversations about a quarter of the time. Trainees chose to see a video conversation Summary for about half the conversations they studied. Trainees elected to compare and contrast the recordings of themselves with expert reps, as LEAP was designed for them to do. While the amount that each of LEAP’s commands was used, on average, varied widely from one command to another, the overall usage pattern was consistent among commands: there was a single peak (Figures 5.69 - 5.78), and a wide range of usage (Table 5.12).
Table 5.12
Command Usage, as Logged

<table>
<thead>
<tr>
<th>Command:</th>
<th>M</th>
<th>SD</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommend</td>
<td>7.5</td>
<td>5.6</td>
<td>1</td>
<td>18</td>
</tr>
<tr>
<td>Top level Help</td>
<td>1.6</td>
<td>1.2</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Practice level Help</td>
<td>2.5</td>
<td>2.4</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>Hint</td>
<td>32.1</td>
<td>28.6</td>
<td>4</td>
<td>149</td>
</tr>
<tr>
<td>Change Instruction Style</td>
<td>21.1</td>
<td>15.2</td>
<td>5</td>
<td>71</td>
</tr>
<tr>
<td>Repeat Conversation</td>
<td>4.9</td>
<td>4.6</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>Video Summary</td>
<td>4.8</td>
<td>3.6</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>Hear Original</td>
<td>17.1</td>
<td>17.7</td>
<td>0</td>
<td>63</td>
</tr>
<tr>
<td>Hear Myself</td>
<td>25.5</td>
<td>23.0</td>
<td>0</td>
<td>72</td>
</tr>
</tbody>
</table>

5.6 Achievement Versus Self- and LEAP-directed Study

This section examines trainee use of various methods for selecting conversations, considers the use of individual commands as predictors of achievement, and examines the learning outcomes of self-directed versus tutor-directed study.

5.6.1 Conversation Selection Methods

There are four ways to select an exercise for practice:

- Click on Recommend (and Accept),
- Click on Instruct after selecting a Topic,
- Click on Instruct after selecting a Conversation.
• Click on Repeat at the end of a conversation,

Details on the usage of the first and last commands were presented above. Details on the usage of the middle two are presented immediately below, together with a pie chart (Figure 5.82) showing the relative frequency trainees used the four exercise selection methods.

Trainees clicked the Instruct button from the Topics Table of Contents an average of 9.0 times (Figure 5.80, Table 5.13), and from the Conversations Table of Contents an average of 9.4 times (Figure 5.81, Table 5.13), indicating that trainees find these two views of the domain about equally useful. Clicking Instruct from Topics gives LEAP a bit more control of instruction, because it then can select the most suitable conversation.

![Instruct (from Topics)](image)

**Figure 5.80**

Instruct (from Topics)
Table 5.13

Conversation Selection Methods

<table>
<thead>
<tr>
<th>Command</th>
<th>$M$</th>
<th>$SD$</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instruct (from Topics)</td>
<td>9.0</td>
<td>8.7</td>
<td>0</td>
<td>31</td>
</tr>
<tr>
<td>Instruct (from Conversations)</td>
<td>9.4</td>
<td>6.9</td>
<td>1</td>
<td>23</td>
</tr>
</tbody>
</table>
5.6.2 Individual Commands as Predictors of Achievement

The number of times trainees took certain actions: Recommend, Repeat Conversation, Change Instruction Style, Hint, Hear Original, Hear Myself, Instruct (from Topics), and Instruct (from Conversations) was considered as a predictor of the number of SA pairs tried (%). Correlations were low, none of these, by itself, significantly predicted the number of SA pairs tried.

The number of times trainees took the same actions was considered as a predictor of the trainee’s Score. Again, correlations were low, none of these, by itself, significantly predicted the trainee’s Score except Repeat Conversation (p < .023) (Figure 5.83). In retrospect, the relationship is easy to understand; repeating a conversation gives trainees an opportunity to raise the score on SA pairs they have already tried, rather than adding new untried SA pairs to their knowledge base.

Finally, Change Instructional Style appeared to correlate significantly with the number of SA pairs tried (and consequently Knowledge and rate of
learning (Knowledge per rehearse-minute)), but dropping the two highest outliers completely eliminated the correlation. There is an important clue here: The two outlier trainees who changed instructional style the most, learned the most. It is possible that LEAP asks too little of trainees, since it currently focuses on only one topic at a time, and either reviews or scaffolds the remainder of the conversation.

The two points above reveal possibilities for making LEAP’s instruction more effective for this application. First, have the trainee do more SA pairs in each conversation. The evidence for this change comes from trainees who clicked on Change Instruction Style a lot; they learned a lot more than others. Second, have trainees stay on the same conversation until it is learned, no matter what the Topics it covers; LEAP’s currently varies conversations as much as possible within Topics. The evidence is that those who Repeated Conversations a lot learned more than those who did not.

**Figure 5.83**

Repeat Conversation versus score
5.6.3 Self-directed versus Tutor-directed Learning

In LEAP, with regard to learning activities, trainees have complete latitude to choose self-direction, tutor-direction, or any intermediate state. This section discusses the differences in achievement between the more self-directed learners and the more tutor-directed learners.

To begin, we must establish a measure of self- versus tutor-directedness. We must also determine a suitable measure of achievement. Regarding the former, a trainee is more self-directed the more she specifies what and how she will study, for example the more she selects the topic, conversation and study method during practice. Certain commands recorded in the log provide an indication of self- versus tutor-directedness (Table 5.14). When the trainee clicks Recommend and Accept, LEAP selects both the topic and the conversation. When the trainee clicks on Instruct after selecting a Topic, LEAP selects the conversation. In the two remaining cases the trainee selects the conversation. In all cases LEAP selects the study method (Observe, Focused Practice, Full Practice) the conversation opens with, but the trainee can change it at any time during the conversation, to whatever she desires.

For the measure of achievement, candidates are Percent of SA pairs Tried, Score on the pairs tried, total Knowledge acquired (the product of Percent Tried times Score), and Rate of learning, or Knowledge per Rehearse-minute. Of these, the last seems best, since it factors out any differences in total amount of time individual trainees studied, and includes both quantity and depth of study (Percent and Score respectively).
Table 5.14
Command Usage and Tutor-directedness

<table>
<thead>
<tr>
<th>Command</th>
<th>Tutor Sets These</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommend, times used</td>
<td>Topic</td>
</tr>
<tr>
<td></td>
<td>Conversation</td>
</tr>
<tr>
<td></td>
<td>Study Method</td>
</tr>
<tr>
<td>Instruct (from Topics)</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Conversation</td>
</tr>
<tr>
<td></td>
<td>Study Method</td>
</tr>
<tr>
<td>Instruct (from Conversations)</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Study Method</td>
</tr>
<tr>
<td>Repeat Conversation</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Study Method</td>
</tr>
<tr>
<td>Instruct Style Change</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
</tr>
</tbody>
</table>

The measures of Recommend and Instruct Style Change are somewhat crude\(^3\), so there is no point in attempting too fine-grained an analysis. The measure of self-directedness focused on SA pair selection (in the context of topics and conversations) and was calculated as in Equation 1.

\[
\text{Self-directedness} = \frac{[(1/2) * b] + c + d}{a + b + c + d} \tag{1}
\]

In Equation 1 the letters a - d represent the number of times these commands were pressed:

\(^3\) Crude means, for example, that LEAP’s Log did not record the difference between Recommend Accept and Recommend Reject; nor did it distinguish at what point within a conversation a trainee hit Instruct Style Change, nor if she hit it once or five times in succession.
a = Recommend
b = Instruct (from Topics)
c = Instruct (from Conversations)
d = Repeat Conversation

The rationale for this formula is that a trainee has complete control over conversation selection when she clicks on Instruct (from Conversations) and Repeat Conversation so these choices are awarded one point each. She has some control when she clicks on Instruct (from Topics), so these choices are awarded 1/2 point each. She has no control when she clicks on Recommend, so these choices are awarded no points. The resulting total is made into a relative measure by dividing it by the total number of times all were clicked.

The resulting measure of self-directedness was compared to various measures of achievement: Percent of SA Pairs Tried, Score, Knowledge, and Knowledge per Rehearse-minute. None of the correlations were significant, but one: Self-directedness versus Knowledge/Rehearse-minute, appeared worthy of further investigation (Figure 5.84).
Trainees at the left end of the self-directed scale, i.e., the tutor-directed trainees, have a learning rate of about .13 percent per minute. As trainees become more and more self-directed, their learning rates begin to scatter; some learn slower than the tutor-directed trainees, some learn faster. No matter, the 'self-directed' value is a highly significant predictor (p < .0009) of the distance of a learning-rate point from the regression line. The trainees who learn faster appear to have good reason for ignoring LEAP's direction. As mentioned, LEAP's designers may be able to improve its instructional methods by having LEAP emulate them, i.e., by repeating conversations instead of varying them, and by giving students more SA pairs to practice within each con-
versation. On the other hand, trainees who learn slower than the tutor-di-
rected trainees, could probably benefit by taking more direction from LEAP.\(^4\)

A second measure of self-directedness was also considered. This mea-
sure took Instruct Style Change into consideration. The goal of this second
measure is to give two factors approximately equal weight: trainee selection of
what to study, and trainee selection of how (intensively) to study. The first
factor is represented by the earlier self-directedness measure, the second by
Instruct Style Change.\(^5\) To make the two factors relative (i.e., dimensionless)
each was divided by its median value. The two were then multiplied together
to obtain a self-directedness measure. The formula for this measure is dis-
played in Equation 2.

\[
\text{Self-directedness} = \frac{a}{b} \times \frac{c}{d}
\]

In Equation 2 the letters a - d represent the following values:

- \(a = \) Self-directedness as measured by Equation 1
- \(b = \) Median value of self-directedness as measured by Equation 1
- \(c = \) Number of times a command to change instructional style was
  pressed
- \(d = \) Median value of number of times a command to change instruc-
  tional style was pressed

\(^4\) It's possible, of course, that by following LEAP's instructional directives, they would learn
even more slowly.

\(^5\) It is a reasonable assumption that when clicking on Instruct Style Change, trainees were
switching from Focused Practice to Full Practice, or from Observe to Focused or Full Practice.
Conversations came up in Observe occasionally, and Focused Practice otherwise. Switching to
Observe from either of the other two modes would have lowered their score, not raised it.
This measure appears reasonable. The attempt to measure its effectiveness was thwarted by the condition mentioned in the previous section, namely, two trainees used Instruct Style Change extensively, and as a result, significantly increased the number of SA pairs they tried during practice compared to the number of SA pairs LEAP would have had them try. The effects of practicing more carried over into measures of Knowledge and Learning Rate. Also, as mentioned, dropping these two trainees from consideration reduced those same correlations to non-significance. Thus this second measure of self-directedness while having face validity does not provide further insight in this case.

**Summary:** This section considered the use of individual commands as predictors of achievement and examined the outcomes of self-directed versus tutor-directed learning. The number of times trainees used various commands was neither a predictor of the number of SA pairs tried nor of the trainee’s Score (except Repeat Conversation, which raises the score on SA pairs already tried, and Change Instructional Style, which increases the number of SA pairs tried). LEAP’s instruction may thus become more effective by changing two current instructional strategies: first from varying conversations to remaining on them, and second from focusing on one topic at a time to practicing the whole conversation.

Trainees may choose self-direction, tutor-direction, or any intermediate state. Self-directedness was not correlated with any measure of achievement, however, as trainees become more self-directed, their learning rates begin to scatter; some learn slower than the tutor-directed trainees, some learn faster. Those trainees who learn faster are justified in ignoring LEAP’s direction. Trainees who learn slower than tutor-directed trainees could benefit by taking instructional direction from LEAP. Thus a third change in strategy would be
to have LEAP intervene when trainees are floundering, and tell them to take more direction from the tutor.

5.6.4 LEAP Usage Summary

This section presented data describing how trainees actually used LEAP as captured by LEAP's logging function. LEAP was field trialed in five sites for 'one day' each. Trainees actively used LEAP four hours, trying eleven different exercises. They spent 13 minutes in the Guide, 207 minutes in Rehearse Conversation, and 19 minutes Exploring Contact Flow. During the trial, trainees tried about half of LEAP's learning material, attaining a score of 'Good' on it. Trainees' knowledge acquisition rate is normally distributed indicating LEAP accommodates diverse learning styles. Rehearse Conversation is LEAP's most productive study method and its use correlates with the amount of material learned. Trainee command usage can be generally characterized as having a single peak, either broad or narrow, at some value, and a broad range. Many trainees used Recommend infrequently. Trainees needed little Help but many Hints. They changed instruction style frequently, Repeated conversations a quarter of the time, and saw a Summary half the time. Trainees frequently compared and contrasted their voice recordings with expert reps'. Frequencies of individual command use did not correlate with learning outcomes except Repeat Conversation, and Change Instructional Style. LEAP's instruction may be made more effective by remaining on conversations longer, practicing entire conversations, and by becoming more directive when trainees are floundering. Tutor-directed trainees learned as fast, on average, as self-directed trainees, with far more consistent results.
5.7 Individualization of Instruction

"... the power of an ALE [adaptive learning environment] to adapt its instruction to a student based on data it collects during tutorials ... is a major topic of basic research into ALEs [that has received]... too little attention ...” (Winne, 1993 p. 304)

In this section, two measures of LEAP’s ability to individualize instruction are presented. The first examines the sequencing of exercises presented to students. The second compares and contrasts three students’ means of arriving at the same ending performance level.

5.7.1 Variations in Exercise Sequencing

One way to measure individualization of instruction is to examine the extent to which LEAP varies in its sequencing of exercises for students to study. If LEAP were not individualizing instruction it would visit all conversation exercises in the same sequence, since student response history (learning) would have no effect on the selection of exercises. On the other hand if LEAP were individualizing instruction by taking into account student response history, the position in the instructional sequence of any given exercise should appear more and more ‘random’ the later it is, on average, visited. That is exactly what the data in Figure 5.85 reveal.

To obtain these data, a grid was laid out consisting of a row for each exercise and a column for each student. The column for each student was filled with numbers indicating the sequence in which he or she first visited each exercise. These numbers were then averaged for each exercise and their standard deviation calculated.
The data go in the expected direction, the result is significant \((p < .001)\) and the exercise’s position in the sequence alone accounts for 53\% of the variance.

**Figure 5.85**

Conversation SD versus average position in visitation sequence

---

1 While the data go in the expected direction, there are two confounding factors. First, four of the five training groups received initial instruction on how to use LEAP on the conversation addvms-custom-ring-laess, the other class received initial instruction on addvms-ccw-cf. Starting instruction from two different points will introduce a certain amount of randomness into the sequencing. Second, students were free to pick any exercise at any time, to exit exercises early, repeat exercises as often as desired, etc. This means that LEAP had to deal with rapidly diverging student models.
5.7.2 Same Performance, Different Paths

This section looks at the extent to which three trainees who are nearly identical in terms of overall Performance scores (Score x Percent Tried) differ from each other in terms of instructional choices, and notes that LEAP accommodates their different approaches. No claim is made for LEAP’s instructional skills here, (we have already looked at how well LEAP-directed students learn in comparison with the best students) it is the student’s instructional choices we are observing, not LEAP’s instructional abilities. The point is simply to show that even students who end up at the same point make diverse instructional choices, and that LEAP is versatile enough accommodate their approaches.

The three trainees selected for close examination were selected because they are alike in the sense that their overall Performance scores (Score x Percent Tried) are nearly identical to each other and near the mean for all trainees, and furthermore the two components of Performance: Percent Tried and Score are the most similar for these trainees. However, it will be shown that their individual learning styles, in terms of command usage, are very different from each other. Furthermore, the pathways by which they achieved their identical Performance are very different.

Although the data in this section are numerical, a qualitative interpretation seems in order. The fact that the three trainees are alike or different by some amount on one measure of command use is not by itself very meaningful. However, if the trainees are quite different from each other on a number of measures, then we can establish qualitatively that the trainees do take different approaches to learning while in LEAP. Conversely we can say that LEAP is successful to the extent that it permits these different approaches to learning to be successful.

276
To compare trainee similarities and differences on a number of dimensions requires a standard measure of dispersion, or difference. Standard deviation is the measure of dispersion chosen for this work, other measures would give similar results. The range of the three selected trainees' responses was divided by the standard deviation for the group to get a value indicating the similarity/difference among the three trainees (the Selected Trainees' Range per all trainees' SD, or STR/SD). When this value is less than 1, the three trainees' actions were quite similar, when the value is more than one, their actions were rather different.

In the tables below (Tables 5.15 through 5.17), the first part shows, for all trainees tested, the minimum, mean, maximum, and standard deviation values, and the second part lists the three individuals' scores. The bottom row of the table is a measure of individual differences among the three trainees. The value in the bottom row is computed by taking the range of the three trainees' scores, i.e., how the three trainees differed among themselves, and dividing it by the standard deviation for all trainees.

Table 5.15 establishes the similarity of the selected trainee's Performance. All the STR/SD are much less than 1. Among the trainees scoring near the mean, these three are also most nearly identical to each other in the Percent of VMS material tried, and their Score on that material.

---

2 Several measures of dispersion exist: the most common are range, variance, and standard deviation; others are mean deviation, the interquartile range, the semi-interquartile range, the midquartile, and the 10-90 percentile range.
Table 5.15
Similarity of Selected Trainee’s Performance (1 of 3)

<table>
<thead>
<tr>
<th>Performance</th>
<th>Percent Tried</th>
<th>Score (of 100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Trainees:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Min.</td>
<td>12.5</td>
<td>22</td>
</tr>
<tr>
<td>M</td>
<td>27.6</td>
<td>50</td>
</tr>
<tr>
<td>Max.</td>
<td>43.3</td>
<td>75</td>
</tr>
<tr>
<td>SD</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>Selected Trainees:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G1-1</td>
<td>26.6</td>
<td>41</td>
</tr>
<tr>
<td>G2-5</td>
<td>27</td>
<td>38</td>
</tr>
<tr>
<td>G2-6</td>
<td>27.7</td>
<td>39</td>
</tr>
<tr>
<td>Selected Trainees’ Range per all trainees’ SD:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STR/SD</td>
<td>0.14</td>
<td>0.21</td>
</tr>
</tbody>
</table>

Figure 5.86 shows the distribution of Performance for all trainees, with the position of the three selected trainees marked with Xs. Since the scores are nearly identical to each other and near the mean for the group, they fall in the same bar. (Figure 5.87 shows a contrasting example.)
As mentioned, while the three trainees are very similar in their overall Performance scores, they arrived at the same end point by very different methods. In their use of the following instructional options (Tables 5.16 and 5.17), the three trainees differed among themselves by 1, 2 or even 3 standard deviations:

**Instruct from Topics:** The three trainees varied among themselves in their use of this function by more than two standard deviations (Table 5.16). This was expected to be the standard study mode; trainees would select a topic to study, press the Instruct button, and LEAP would present a sequence of conversations relating to the topic until the topic had been mastered.

**Instruct from Conversations:** The three trainees are similar to each other on this particular measure (Table 5.16): They did not select specific conversations to study from the Conversations Table of Contents, but instead used a variety of other methods of selecting conversations.

Figure 5.86

VMS performance, showing values for selected trainees
**Recommend:** The three trainees varied among themselves in their use of this function by more than two standard deviations (Table 5.16). This was expected to be the standard study mode for trainees who were not particularly self-directed. LEAP’s logging function did not record whether the recommendation was accepted or rejected, so the extent to which trainees took LEAP’s advice on which topic to study is not known.

**Repeat Conversation:** The three trainees varied among themselves in their use of this function by more than three standard deviations (Table 5.16). As would be expected, the use of this function is inversely proportional to Instruct from Topics. Repeating a conversation is a way to practice the same SA pairs until they are mastered.

**Changing the Instructional Style:** The three trainees varied among themselves in their use of this function by more than a standard deviation (Table 5.16). LEAP selects the instructional style it believes optimal for learning a particular topic at a particular experience level (first it selects Observe, then it selects Focused Practice). Trainees could switch styles if they found the particular style too hard or too easy; they could switch styles of either the conversation portion or of the SONAR portion of the conversation, or both; and switch styles back again, as often as they liked.
Table 5.16
Similarity of Selected Trainee’s Performance (2 of 3)

<table>
<thead>
<tr>
<th></th>
<th>Instruct from Topics</th>
<th>Instruct from Conversation</th>
<th>Recommend</th>
<th>Repeat Conversation</th>
<th>Instruction Style Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Trainees:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Min.</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Mean</td>
<td>9</td>
<td>9</td>
<td>7.5</td>
<td>4.9</td>
<td>21</td>
</tr>
<tr>
<td>Max.</td>
<td>31</td>
<td>23</td>
<td>18</td>
<td>19</td>
<td>71</td>
</tr>
<tr>
<td>SD</td>
<td>8.6</td>
<td>7</td>
<td>5.5</td>
<td>4.6</td>
<td>15</td>
</tr>
<tr>
<td>Selected Trainees:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G1-1</td>
<td>21</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>11</td>
</tr>
<tr>
<td>G2-5</td>
<td>9</td>
<td>3</td>
<td>14</td>
<td>9</td>
<td>20</td>
</tr>
<tr>
<td>G2-6</td>
<td>1</td>
<td>4</td>
<td>14</td>
<td>16</td>
<td>28</td>
</tr>
<tr>
<td>Selected Trainees’ Range per all trainees’ SD:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STR/SD</td>
<td>2.33</td>
<td>0.29</td>
<td>2.18</td>
<td>3.04</td>
<td>1.13</td>
</tr>
</tbody>
</table>

Figure 5.87 shows the distribution of Repeat Conversation for all trainees, with the position of the three selected trainees marked with Xs. The range of the three selected trainees’ scores is more than three standard deviations of the group’s scores. (Figure 5.86 shows a contrasting case.)
Repeat Conversation, showing values for selected trainees

Table 5.17 presents data for two last instructional options, for time spent practicing, and for LEAP versus human role-play preference.

Hear Original and Hear Myself: The three trainees varied among themselves in their use of these functions by more than two standard deviations (Table 5.17). Upon completing a conversation, trainees could compare their spoken responses to those of the experts by using these buttons. Making the comparison permits the trainees to bring their performance closer to the experts’.

Rehearse, minutes: The three trainees varied among themselves in the number of minutes it took them to reach the mean performance level by more than one standard deviation (Table 5.17), indicating that different trainees learn at different rates.
Trainees were asked three questions (in semantic differential form) pertaining to their preference for practicing role-plays with humans: an instructor, fellow-trainee, or customer vs. LEAP. The preferences were summed for the three questions and the result is presented in the table below. On average, trainees preferred to practice role-plays with LEAP, but in fact the distribution of preferences is bimodal; trainees either strongly preferred practicing role plays with humans or they strongly preferred practicing role plays with LEAP.

Preference for role playing with a human or with LEAP: The three trainees exhibited preferences for role playing with both humans and with LEAP (Table 5.17).

Strength of Preference: The three trainees varied among themselves in the strength of their preferences by more than one standard deviation (Table 5.17).

To summarize, three trainees who were nearly identical in terms of overall Performance scores (Score x Percent Tried) differed from each other substantially in terms of instructional choices, learning time, and preferences. Trainees do take different approaches to learning while in LEAP, and LEAP permitted these different approaches to learning to be successful. Next, we take a look at a graphic representation of the trainees’ paths to performance.
Table 5.17
Similarity of Selected Trainee’s Performance (3 of 3)

<table>
<thead>
<tr>
<th></th>
<th>Hear Original</th>
<th>Hear Myself</th>
<th>Rehearse minutes</th>
<th>Prefers role plays with:</th>
<th>Strength of Preference (-3 to +3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Trainees:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Min.</td>
<td>0</td>
<td>0</td>
<td>142</td>
<td>LEAP</td>
<td>-3</td>
</tr>
<tr>
<td>Mean</td>
<td>17</td>
<td>25</td>
<td>207</td>
<td>LEAP</td>
<td>-0.8</td>
</tr>
<tr>
<td>Max.</td>
<td>63</td>
<td>72</td>
<td>263</td>
<td>Human</td>
<td>3</td>
</tr>
<tr>
<td>SD</td>
<td>18</td>
<td>23</td>
<td>33</td>
<td></td>
<td>2.3</td>
</tr>
<tr>
<td>Selected Trainees:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G1-1</td>
<td>24</td>
<td>60</td>
<td>209</td>
<td>Human</td>
<td>1</td>
</tr>
<tr>
<td>G2-5</td>
<td>13</td>
<td>10</td>
<td>231</td>
<td>LEAP</td>
<td>-3</td>
</tr>
<tr>
<td>G2-6</td>
<td>63</td>
<td>50</td>
<td>259</td>
<td>LEAP</td>
<td>-2</td>
</tr>
<tr>
<td>Selected Trainees’ Range per all trainees’ SD:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STR/SD</td>
<td>2.78</td>
<td>2.17</td>
<td>1.52</td>
<td>--</td>
<td>1.74</td>
</tr>
</tbody>
</table>

Figure 5.88 is a plot of the three trainees’ performance (Score vs. Percent of material tried) after each exercise. The plots trend from the initial value of 0,0 toward the upper right corner of the chart. When a trainee tries a new exercise or a new topic, the Percent of Material tried will increase, and the Score will usually decrease since the score represents the average score for all SA pairs tried. When the trainee repeats an exercise, the percent of mate-
rial tried will remain constant\(^3\) and the score will change (presumably in-
creasing!).

For example, as noted in Table 5.18, trainee G2-6 begins by doing one
conversation in Observe mode, viewing 10\% of the material and attaining a
score of 6. She then repeats the exercise in Full Practice and, being guided by
the instructor, attains a score of 72. At this point she tries a new exercise,
adding 16\% of LEAP’s material, and decreasing her overall score to 31. By re-
peating the same exercise three more times, she raises her score to 76. In the
next exercise (as a detailed examination of the log -- not shown -- reveals) the
trainee messes around with Help and Hints, changes instruction styles sev-
eral times, exits early (i.e., before completing the exercise), breaks for lunch,
and is generally unproductive for about an hour and a half, adding only one
percent new material and raising her score by only one point. In general, this
student’s strategy is to stay in the same topic, Order Voice Messaging, and re-
peat conversations many times. In Figure 5.88, her learning curve rises high
and stays high.

\(^3\) The percentage of material tried will remain constant as long as the trainee does not switch to
Full Practice, thereby adding new SA pairs to his/her user model.
Table 5.18
Performance of Trainee G2-6

<table>
<thead>
<tr>
<th>Percent of Material</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>72</td>
</tr>
<tr>
<td>26</td>
<td>31</td>
</tr>
<tr>
<td>26</td>
<td>58</td>
</tr>
<tr>
<td>26</td>
<td>58</td>
</tr>
<tr>
<td>26</td>
<td>67</td>
</tr>
<tr>
<td>26</td>
<td>76</td>
</tr>
<tr>
<td>27</td>
<td>75</td>
</tr>
<tr>
<td>27</td>
<td>76</td>
</tr>
<tr>
<td>27</td>
<td>76</td>
</tr>
<tr>
<td>27</td>
<td>76</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

In contrast, as noted in Table 5.19, trainee G1-1 tries many topics, repeats few conversations and tends not to change instructional style. The trend of her performance is first horizontal, covering a lot of LEAP's content, and then vertical, as LEAP finally has her repeat exercises only after trying all the related exercises once.
Table 5.19

Performance of Trainee G1-1

<table>
<thead>
<tr>
<th>Percent of Material</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>71</td>
</tr>
<tr>
<td>12</td>
<td>60</td>
</tr>
<tr>
<td>15</td>
<td>49</td>
</tr>
<tr>
<td>19</td>
<td>49</td>
</tr>
<tr>
<td>21</td>
<td>46</td>
</tr>
<tr>
<td>22</td>
<td>44</td>
</tr>
<tr>
<td>22</td>
<td>48</td>
</tr>
<tr>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>36</td>
<td>35</td>
</tr>
<tr>
<td>36</td>
<td>35</td>
</tr>
<tr>
<td>38</td>
<td>49</td>
</tr>
<tr>
<td>39</td>
<td>48</td>
</tr>
<tr>
<td>40</td>
<td>49</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Finally, trainee G2-5 (tabulated data not shown) takes the middle ground in terms of numbers of topics selected, conversations repeated, and changes in instruction style. The trend or path of her performance is also in
the middle, above G1-1 and below G2-6. The performance paths for the three trainees are shown in Figure 5.88.

![Figure 5.88: Three trainees' paths to performance](image)

To summarize, two measures of LEAP’s ability to individualize instruction were presented. The first examined the sequencing of exercises presented to students. The second compared and contrasted three students’ means of arriving at the same ending performance level. One way to measure individualization of instruction is to examine the extent to which LEAP varies in its sequencing of exercises for students to study. If student learning is
idiosyncratic and LEAP individualizes instruction, the position of any given exercise in the instructional sequence should appear more and more ‘random’ (or trainee-specific) the later it is, on average, visited. That is exactly what the data reveal. Another way to measure individualization of instruction is to examine the extent to which trainees who are alike in their overall Performance are quite different in their means of attaining it, i.e., in their individual learning styles. Three trainees with nearly identical Performance were shown to vary among themselves by as much as three standard deviations in their use of many commands, and to use fundamentally different pathways (depth-first vs. breadth-first learning) to Performance.

5.8 Instructors’ Reflections

The next few paragraphs summarize the responses from the Instructor’s Reflections Questionnaire, administered after each instructor’s trainees had worked with LEAP for a day.

All instructors reported that their students learned effectively with LEAP, that students liked working with LEAP, and that LEAP was generally easy for students to use.

The instructors reported that revising the SONAR simulation and Help screens would improve the training effectiveness of LEAP; that making LEAP faster or more responsive would improve how much students liked LEAP; and that relaxing those database responses that were overly-specified, making the cursor easier to see (in the simulation), and providing more detailed introductory training on LEAP would improve LEAP’s ease of use.

While instructors would prefer LEAP over student-student role plays, they would not prefer LEAP over student-instructor role plays. Furthermore:

It’s easier to learn to develop rapport with a human-human role play. I can’t imagine how to bring that into LEAP. The sales process requires
listening for inflections in voice etc., as buying signals, confusion about a product, etc.

Instructors perceived LEAP’s content as complete, appropriate, and not substantially different from what was taught in class. However:

"LEAP is a tool - not a complete curriculum or training experience - There's plenty of time and opportunity to employ other teaching methods as well."

To the last question ‘Do you have any other comments for us?’ instructors wrote:

"Hope this one gets to you. Not a total substitute for instructor or classroom time - good for refreshers and introducing new products."

"This is a very nice learning tool but too time consuming as yet. Good luck! I think it can be very helpful."

"This was a wonderful demonstration. Each student was able to use different levels based on their knowledge and comfort area. Feedback was one-on-one and they could repeat conversations until they felt competent. Others could go on when they wished, to new and more challenging levels. I was very impressed. Everyone seemed to feel they had gained something - even though some had completed more conversations than others."

In sum, instructors of the trainees who used LEAP during the field trials reported their perceptions of LEAP as a flexible, adaptive instructional medium, an improvement over student-student role plays, and a medium from which students learned effectively; in short, a good tool for classroom use.
5.9 Summary of Evaluation Procedures

Evaluation of LEAP focused on four main themes: first, measuring trainees' affective responses toward LEAP and their perceptions of its usability and instructional value, second, measuring trainees' usage of LEAP, and the resulting learning, third, measuring how well LEAP was capable of individualizing instruction, and fourth, measuring instructors' response to having LEAP in their classrooms.

Trainees' most salient demographic feature was the range of response on every dimension. However, trainee ages clustered in the late 20s and late 30s, their education clustered on high school and two-year degrees, many trainees were new-hires, two-thirds had experience with voice messaging, and half had software experience.

The instructors' initial attitudes toward LEAP as a teaching technology were positive, and they believed their students' attitudes were also positive. They believed LEAP would help the training process in several ways, but feared LEAP would replace needed human interaction with frightening machines.

Trainees reported they learned a lot and were happy, confident, in control, etc. On LEAP versus human role plays, responses spanned the spectrum, slightly favoring the extremes. Trainees believed the proficiency ratings LEAP assigned them.

Trainees reported LEAP to be highly usable. Feedback was especially useful. The system meets the needs of both experienced and inexperienced trainees. The conversations were straightforward, realistic, and easy to understand. LEAP's response time could be improved.
Regarding LEAP’s instructional value: Trainees were very confident they could do a change order after practicing it in LEAP. Trainees found the Recommend function and the Student Model quite helpful. Trainees rated the Rehearse Conversations and Guide to Voice Messaging study methods highly; Examine Contact Flow much lower. Trainees rated Observe, Focused Practice, and Full Practice as extremely useful. Trainees could observe themselves learning.

LEAP’s logging function captured how trainees actually used LEAP. Trainees used LEAP four hours, trying eleven different exercises one or more times. They spent most of the time in Rehearse Conversation, tried half the SA pairs, attaining a score of Good or Excellent on them. Usage graphs for each of LEAP’s commands generally had a broad range and a single peak. Trainees needed little Help with LEAP but needed many Hints on Voice Messaging. They changed instruction style frequently, and frequently compared their voice recordings with expert reps’.

LEAP’s instruction may be made more effective by repeating conversations, practicing entire conversations, and by becoming more directive when trainees are floundering. Tutor-directed trainees learned as fast, on average, as self-directed trainees, with more consistent results.

LEAP’s individualizes instruction. The position of any given exercise in the instructional sequence is trainee-specific, i.e., it appears more and more random the later it is visited. Trainees who are alike in their overall Performance may be quite different in their individual learning styles as indicated by their patterns of command use and pathways to Performance.

The trainees’ instructors reported LEAP to be a flexible, adaptive instructional medium, an improvement over student-student role plays, and a
medium from which students learned effectively; in short, a good tool for classroom use.

To summarize LEAP's four main evaluation themes: first, trainees enjoyed learning with LEAP, found it highly usable, could see themselves learning, and believed themselves capable of performing the real task; second, trainees learned about half of the SA pairs to a level of Good or Excellent in four hours of practicing conversations, requesting many hints and frequently listening to expert reps; third, LEAP provided performance information to self-directed trainees and provided individualized recommendations to tutor-directed trainees; and fourth, instructors found LEAP a welcome addition to the classroom.
CHAPTER 6
CONCLUSIONS and RECOMMENDATIONS

In this chapter I present a general discussion of what I have learned about building intelligent tutoring systems; this section is not about the specifics of LEAP, though I refer to LEAP to exemplify points, but about building intelligent tutoring systems in general. In the first section I distinguish among three related areas of ITS research; AI research in the domain of tutoring is different from instructional research in the domain of tutoring, and building practical tutors is different from them both. In the next and largest section I make a number of recommendations relating to the design of ITSs. In the third section I point out some of the difficulties in evaluating ITSs. In the next section I present some conclusions and recommendations about authoring tools and shells. In the last section I describe the appropriate time for and methods of cost/benefit analysis.

6.1 Tutor-related Research Areas

In this section I distinguish three tutor-related research areas comparing the research goals for each: AI research in ITS, instructional research in ITS (or tutoring research), and practical or application-related research in ITS.

6.1.1 AI Research in ITS

The goal of AI researchers, in tutoring and in other domains, is to find a way to have the computer do something intelligently. AI researchers want to discover, for example, what one must know in order to teach, and how to embed that intelligence in a program (Grosz & Davis, 1994). AI researchers in tutoring are successful, then, if their programs run and exhibit a capacity for tutoring. AI researchers want to make computers intelligent without concern for questions such as: whether an exercise is a genuine learning task, or the extent to which an
actual student would need or benefit from an intervention. In contrast, tutoring researchers want to apply computer intelligence to tutoring humans.

6.1.2 Tutoring Research

When tutoring researchers apply computer intelligence to tutoring, the result is measured in terms of trainee learning. Instructional researchers apply AI techniques, pioneered by AI researchers; their standard of success is that trainees' learning processes improve.

Existing instructional design principles can and should be applied to tutoring, and new theories are also required. Existing instructional design principles do not provide the appropriate level of detail or prescriptiveness that a computer-based tutor needs, nor do existing theories, presupposing human instruction and unintelligent media, utilize a computer-based tutor's unique combination of strengths and alleviate its weaknesses. The goal of tutoring research should be to develop these new principles of computational instructional design.

Tutoring research should tackle portions of actual training problems with the goal of solving them better than they are solved by conventional instruction. Tutoring research should take place on the testbed of an ongoing training activity. The context of the ongoing training activity will provide a stream of trainees and instructors, who are needed for evaluating the quality of the tutoring. The training context will also provide a means for instructing trainees in related information needed for learning which the tutor does not provide. The use of the tutor in a real training context will reveal the genuine achievements of and difficulties for tutoring research.

Tutoring research may focus on tutoring strategies one at a time. While the context is real, there is no need for tutoring researchers to provide a whole training solution. Real trainees will inform the conclusion of what can be done well by tutors and what is best left to conventional instruction. Frequent devel-
opmental testing (formative evaluation) of the tutor on users will make the effect of intelligent tutoring strategies clear. Time lost to building conventional features and to building user interface items, etc., will be more than made up for by time saved pursuing interesting but unproductive inferencing power. No claims can be made about gains in tutoring research without testing the tutoring strategies on real trainees.

6.1.3 Practical Tutors

In contrast to AI researchers, who want to make computers intelligent, and tutoring researchers, who want to find intelligent tutoring strategies that optimize learning, practical tutor builders want to put tutors into the workplace or learning environment. Practical tutor builders are concerned with the real world; unlike the tutors AI researchers and tutoring researchers build, practical tutors must not only really teach but do so within real-world constraints.

One goal of a practical tutor is that trainees learn in a real-life setting. Building a practical tutor may turn out to be straightforward and no more difficult than any other training and software development effort, but this will only become known when building a practical tutor is the explicit goal.

Practical tutor builders apply the tutoring strategies acquired from tutoring research and use conventional instruction to fill in as needed; the result is a good training system with some intelligent tutoring strategies in it.

It is a given that a tutor encodes some instructional knowledge; however a practical tutor must encode enough instructional knowledge to make good instructional decisions over the range of teaching situations in which the tutor finds itself with respect to trainees’ learning styles, current knowledge states, and learning goals. It is also a given that a tutor has some instructional resources; however a practical tutor must encode an adequate variety of instructional resources on which to apply its knowledge. For example, however knowledgeable
the tutor is about selecting exercises, if there are too few exercises to choose from, the tutor will not be able to select the optimal one for a given situation. LEAP's three-level approach, with the ability to select a study method and to select and modify study materials at each level of detail, provides a vast space of instructional possibilities. A second goal, then, of practical tutors is to encode a broad range of instructional knowledge and instructional resources, so as to have the potential to optimize learning.

A practical tutor has another goal beyond the optimization of learning: the certification of learning. This third goal (the most important from a summative perspective) is to certify that the trainee is done, that the trainee indeed knows the knowledge that the tutor is attempting to teach, that the tutor and the trainees have reached their learning goal. The student model will reveal when this goal has been attained.

To summarize, in this section I have pointed out the distinction between AI research, tutoring research and building practical tutors. I define tutoring research as a different kind of research from AI research, one that is a necessary step in the quest for practical tutors. Tutoring research does not claim to make the computer intelligent nor claim to show a positive cost/benefit ratio. It finds new principles of computational instructional design and validates them. Tutoring research takes the results of AI research in tutoring and builds on them; it also informs the direction of AI research in tutoring. Tutoring research contributes toward the goal of practical tutors by requiring many of the features of practical tutors since trainees must use these tutors (just as they will use practical tutors) in order that their tutoring abilities be evaluated.

6.2 Design

In this section I draw the following conclusions and recommendations: First, keeping stakeholders involved in the tutor development process is vital to
the success of a tutoring project. Second, keeping trainees oriented on multiple levels is necessary for effective learning. Third, ITSs are one of a spectrum of training design options and their fitness to the training need must be carefully analyzed. Next, representing domain expertise for instructional needs is different from representing it for expert systems. Fifth, an iterative instructional design process is a necessity for tutoring researchers. Sixth, whether the tutor or the trainee should be in control of the learning process depends on the purpose of the tutor. Seventh, I recommend several forms of feedback. Finally, I recommend minimizing multi-media and using simulations in place of actual environments.

6.2.1 Design for Stakeholders

Getting the many stakeholders involved and keeping them involved during the development process is vital for the acceptance of a practical tutor. Working with stakeholders from the start ensures that stakeholders understand and agree with the goals of the project; keeping stakeholders involved means they can provide assistance when needed, and can more easily accept the inevitable delays and changes in the nature and scope of the project. Working closely with stakeholders also helps ensure the success of the project by keeping all project members aware of the goals and interests of the stakeholders. With the stakeholders’ interests in mind, the answers to many questions are straightforward. For example: Which machine? The trainees’ machine. How much network capacity? The network capacity that reaches the trainee’s desks. How flexible does the knowledge representation have to be, and how far should the authoring system be developed? They have to enable developers to keep up with changes in the domain knowledge that occur during the development period, and they have to provide reviewers meaningful access to the knowledge.

One of the major contributing factors to LEAP’s acceptance was the continuous contact the development team maintained with its stakeholders: new
service reps, experienced service reps, trainees, trainers, instructional designers, union representatives, and managers from all related organizations.

6.2.2 Design for Trainees

In this section I present recommendations to keep in mind when designing for trainees. In brief, one should keep trainees informed of where they are in the instructional process, make trainees' options clear, both in terms of interface actions and instructional activities, and finally, design to enable transfer.

Trainees need to know where they stand in terms of reaching their instructional goals. LEAP's trainees found their student model information, as reported in topic scores and exercise challenges, very useful. Trainees may have found it helpful if LEAP had been even more explicit in terms of how much material each topic contained, how much of it they had tried, and their average score on that material. In Explore mode for example, it might have been informative to display even individual SA pair scores; these scores, if displayed as color-coded nodes in a tree diagram of the topic, would have indicated which branches could be gainfully explored.

As with all software, user interface considerations are extremely important; for example, screens should be designed so that the trainee's attention focuses naturally at the desired location, and buttons should be labeled so that a naive user can guess their function.

The instructional landscape should not be a mystery to the trainee. The organization and extent of the instructional materials must be made clear; e.g., topics and exercises should have descriptive names, the nominal topic and exercise sequence should be obvious, etc.

Tutors must be designed for the full range of trainees, who may vary widely in: age, education, experience, product familiarity (i.e., domain knowledge), and gender. One of the more interesting results of the evaluation of the
utility of LEAP’s features for learning was the range of responses to each feature: on a seven-point Likert scale (from -3 to +3, or from “completely obstructive” to “extremely useful”), the usual range of responses was 5 or 6. While most features did have a distribution around some peak value, the range of responses to each feature indicates the value of a tutor’s providing multiple ways for trainees to attain a learning goal.

The design of the training environment should replicate significant details of the work environment. Care should be taken not to overlook features that could cause problems with transfer from the training to the working environment. For example, U S WEST’s service reps worked on UNIX terminals and LEAP was developed on a UNIX terminal, but not until an expert service rep tried the development terminal was it discovered that the training keyboard was different in several important ways from the service center keyboard.

6.2.3 ITS and Other Training Design Options

In this section I place the ITS on a spectrum of training design options, for the purpose of determining where it might most suitably be used. An ITS is one of several kinds of instruction, ranging from on-the-job training through conventional instruction to ITSs. The designs are summarized in Table 6.1.

The first option, mentioned in the early days of tutoring (Winkels, Breuker, & Sandberg, 1988), was to design the tutor as a substitute for a person looking over one’s shoulder as one worked, and coaching by making occasional helpful comments. It is now understood that this sort of tutor is a near-impossibility, owing to the difficulty of discerning unambiguously the task the trainee is attempting.

The second option, on the job training (OJT), is the common informal learning situation: the student is placed in the real environment, tasks are selected by the supervisor, peers provide coaching. This sort of training design (or
training non-design) has a negligible initial cost, but may have high labor and productivity costs.

The next training design option consists of a real environment, but the genuine tasks and coworkers are replaced with a set of exercises and an instructor; for example, a woodworking shop or a culinary arts class. The environment is a real shop or kitchen, but the activities are mainly instructional. This design has a moderate initial cost, moderate labor cost, and a moderate productivity cost.

Next, the actual environment is replaced with a simulated one, for example, an airplane or an industrial process may be replaced by a simulation. The exercises and instructor are as before. The simulator reduces the danger, may reduce the cost of equipment, and increases the ability to monitor the trainee’s actions in detail. This design has a high initial cost, a moderate labor cost, and a low productivity cost.

Finally, to the simulated environment and exercises, a computer tutor is added, partially replacing the human instructor, or providing instruction when a human is too expensive. The computer tutor requires not only tutoring skills, but also a student model and an expert model. This design has a high initial cost, a low labor cost, and a low productivity cost.

To summarize, this discussion of training designs focuses on three features of the training space: the environment, the tasks, and the instructor. The characteristics of each feature range from the natural to the artificial: from the actual workspace to a simulated one; from naturally occurring tasks to predefined exercises; from peer tutoring through professional instruction to a computer tutor. At the all-natural end of the spectra is OJT, at the other end, ITS. For some tasks, ITS appears to have a potentially lower overall cost than OJT.
Table 6.1
Training Design Options

<table>
<thead>
<tr>
<th>Environment is:</th>
<th>Tasks are:</th>
<th>Tutoring by:</th>
<th>Kind of Training:</th>
<th>Cost: Initial</th>
<th>Cost: Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real</td>
<td>Natural</td>
<td>Computer</td>
<td>Intractable</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Real</td>
<td>Selected</td>
<td>Co-workers</td>
<td>On the job</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Real (e.g., machine shop)</td>
<td>Exercises</td>
<td>Instructor</td>
<td>Conventional</td>
<td>Med.</td>
<td>Med.</td>
</tr>
<tr>
<td>Simulated (e.g., flight simulator)</td>
<td>Exercises</td>
<td>Instructor</td>
<td>Conventional</td>
<td>High</td>
<td>Med.</td>
</tr>
<tr>
<td>Simulated</td>
<td>Exercises</td>
<td>Computer</td>
<td>ITS</td>
<td>High</td>
<td>Med.</td>
</tr>
</tbody>
</table>

6.2.4 Representing Domain Expertise

Clancey observed that the expert model for an ITS not only has to solve domain problems, but in order to teach, it has to solve them using a process closely akin to the approach human experts take. Anderson, too, claims the expert module must be able not only to solve exercises in the domain but also to solve them in a cognitively plausible manner. There is no doubt that a tutor’s domain expert should model the human problem solving process. However, in neither case can these expert models solve domain problems in natural settings (Clancey’s NEOMYCIN does not work with human patients, and Anderson’s LISP tutor does not observe the world and write LISP programs to address its problems, nor does it interpret the text of the exercise). In fact, if an expert system can be written to perform a task, there is no further need for training humans to do so. We disagree with Anderson’s (1988) statement “By definition, intelligent
tutoring systems can be built only for domains for which expert systems ... exist” (p. 25).

Instead, we believe that what is important is that the tutor have an expert model of the human problem solving process for the limited domain of the tutor’s instructional environment. As LEAP’s expert model illustrates, a weaker (less inference-capable) expert model may serve for the purposes of instruction, and such a model has the added benefit of being constructable by those who are neither computer scientists nor knowledge engineers. Domain knowledge is represented as rule-like SA pairs; domain exercises are instantiated as linear transition networks. Based on Murray’s (1991) classification of low inferencing versus high inferencing tutors, LEAP is a low inferencing tutor, yet trainees learn from it.

One key function of a tutor’s expert model ‘rules’ is to act as indicators of domain knowledge, that is, to encode something like: Given knowledge Q, when conditions Y and Z pertain, it is appropriate to take action X. Then, if a trainee takes action X when conditions Y and Z pertain, the tutor has a basis for incrementing the certainty that the trainee understands knowledge Q, which supports taking action X under conditions of Y and Z; i.e., the application of the rule indicates an understanding of the supporting knowledge. The action is the visible tip of the iceberg of knowledge.

In LEAP, for example, an SA pair represents one meaningful action taken on the interface, given a context-specific situation. LEAP has a two-layered representation of SA pairs, one abstract layer and one instance layer. The abstract layer is excellent for representing domain knowledge and for characterizing knowledge across exercises and within topics. It is also good for authoring, student modeling, linking the current action to related information in presentations, and hinting. The instance layer illustrates a specific application of the rule in a particular conversation. The abstract layer describes the action, the instance layer illus-
trates the action. Again, LEAP’s expert model could not attend a human customer; it does model an expert rep’s problem solving process and is well-suited to LEAP’s instructional tasks.

The importance of an expert model is matched only by the difficulty of acquiring it. Capturing and representing the domain expert knowledge is perhaps the major task of building a tutor. Yet, it is difficult to capture a cognitively plausible expert representation even for the limited domain of the tutor’s instructional environment. Having access to a good domain expert makes the task considerably easier: LEAP’s SA pairs and topic nets evolved from a representation suggested initially by a trained instructional designer who had once been a service representative herself\(^1\). Nevertheless, LEAP’s knowledge acquisition process was not straightforward. The sequence of database screens served to constrain the conversation, but this did not become obvious for some time. We could not completely and correctly acquire screen names, field names and data entry tasks solely from interviews. Videotape of the screens together with voice recording of simulated conversations did not reveal the underlying structure of the conversations, nor even resolve the screen in adequate detail to obtain database interactions. Videotape does not have adequate resolution for capturing screen details. Some means of capturing database screens and logging keystrokes synchronously with conversation was a necessity.

To summarize, I conclude that the representation of domain expertise should be guided by instructional needs, not by the goal of reproducing domain expert performance in a natural environment. Acquiring even this more-limited domain expert will consume a lot of resources.

\(^1\) Marilyn Burlingame, Telcom Training Corporation, Vancouver BC.
6.2.5 Design: Instruction

In this section I make these instructional design recommendations: keep instructors in the design and development process, use instructional design theory plus an iterative design approach, and provide a variety of learning activities leading up to full task performance.

During the design phase, it is important to remember the tutor is part of a larger instructional system. In particular, the tutor should be designed to augment the abilities of, not to replace, the human instructor. Keeping instructors in the instructional process enables the tutor to focus on doing what tutors do best and simultaneously reduces the perceived threat of ITSs (i.e., automated instruction) to the instructor; both points contributed to LEAP’s success.

Conventional instructional design theories, with their admitted shortcomings when applied to ITS (see the Literature Review section), still have a lot to offer the builder of ITSs. First, they provide a language for discussing instruction. Second, they provide a set of constraints to which ITS should conform (e.g., telling is not teaching). Third, they provide a set of instructional actions tutors can apply. While instructional design theory is necessary, it is not sufficient; an iterative design approach to obtain trainee, teacher, designer, and SME input and feedback is also required.

The tutor’s instructional design should permit a variety of approaches to learning because trainees’ preferences and usages vary widely (see Evaluation chapter). The tutor’s instructional design should have a range of actions that permit the trainee to increasingly shoulder the task, for example: observe, do with hints, do by self. Scoring should be set up so as to require the trainee to eventually do the task by him or herself.
6.2.6 Design: Tutor Control Versus Trainee Control

AI researchers, tutor researchers, and practical tutor developers each have a perspective on the issue of control. AI researchers assume the tutor is in control and try to create a system capable of making control (tutoring) decisions intelligently. Tutor researchers wear both hats: In laboratory experiments, they might also want full tutor control (with the control group having trainee control) but in experiments in naturalistic settings they would require the same conditions as developers of practical tutors. Because trainees are ultimately responsible for their own learning, developers of practical tutors know that tutor control of the learning situation is unacceptable. The tutor can only recommend actions and advise on their probable effects. A practical tutor must allow trainees any degree of control they desire and should encourage students to take control of their learning.

If trainees are to have control over their learning activities, then the tutor must have an interface that gives trainees a view of and access to the instructional resources in the tutor. Something akin to the table of contents of a textbook should do. The tutor must further provide the information required for making informed instructional decisions, that is, trainees should have direct viewing access to meaningful forms of the domain expert knowledge and of their own student model.

Access to the expert model should be like access to reference material relating to the domain: concise statements of core knowledge, all in one place, not necessarily meaningful without studying the related supporting information that is available in some form.

The student model should give feedback to trainees about the effects of their choices (both their accomplishments and their learning rates). The student model should also give the trainee a clear picture of what the tutor knows about
the trainee that forms the basis for its instructional recommendations. Access to
the student model demystifies the tutor. “Ah, it thinks I don’t know that because
I got it wrong twice.” “I have a score of Excellent on that section because I have
used all the rules correctly.”

6.2.7 Design: Feedback

Feedback on trainee performance can take place on several levels: action,
exercise, and topic, and I recommend that it be provided at every level. On the
action level, feedback, either explicit or implicit, after every action tells trainees if
they have taken an appropriate action or not, and if not, gives them the chance to
try again and to practice taking the appropriate action for the situation and con-
text. Action-level feedback ensures that correct performance is practiced until
learned. Besides feedback, the system may also need the capability of providing
direction, ‘feed-forward,’ in the form of recommendations and action-level hints
(graded) to speed trainees’ attainment of correct performance.

On the exercise level, LEAP informed trainees of the exercise’s Challenge,
which is a form of feedback about one’s current state that is useful when select-
ing an exercise to study next. A tutor should also inform trainees, after they have
completed an exercise, of both performance and mastery information. Figure 6.1
shows how these data could have been presented in LEAP. The figure contains
seven items of information: practice and review scores (numbers and percent
correct); starting and ending scores for the conversation; gain (ending score less
starting score); maximum score for the conversation given the topic; and chal-
lenge (maximum score less starting score). Five of these data are presented
graphically for ease of comprehension.
Trainees reported that the topic-level feedback in LEAP (their topic score) was very useful to their learning. Moreover, since topic names and scores were grouped in a two-column list, a trainee’s overall progress on the course material was also clear at a glance. Trainees tended to stick with a topic, once chosen (as did the tutor’s own instructional guidance), until they reached a score of Good or Excellent on it, rather than pursue a number of topics simultaneously. Thus simply providing trainees feedback on where they stood in terms of knowing a topic influenced their behavior.

Since trainees made instructional choices on only the topic and conversation levels, the topic Score and conversation Challenge were sufficient to inform these decisions. It was not necessary to provide scores for individual rules. As noted in the section Improvements in LEAP, feedback, in terms of rule scores, might help trainees select paths (specific sequences of SA pairs within a topic) when they are doing Full Practice in Explore mode.
6.2.8 Design: Media

In this section I make three recommendations regarding instructional media in an ITS: do not use multimedia if it can be avoided, use real objects from the work environment when there is an instructionally sound reason to do so, but simulate software from the work environment.

Intelligent multimedia is at a primitive stage of development (Maybury, 1993). Any multimedia in an ITS must therefore consist of simple presentations. When multimedia is used, users insist it be of high quality, yet high quality multimedia is costly to produce and all multimedia is difficult to update. After a few minutes of use, the gee-whiz response to sound and motion on a computer screen becomes ho-hum and trainees are not impressed. Moreover, more is not always better; for example, not only are line drawings easier to comprehend than color photographs, they also require less bandwidth to transmit and less disk space to store. In sum, the instructional cost/benefit of multimedia must be carefully considered.

Supporting information can appear in any instructionally appropriate form (such as hardcopy). Sometimes it is desirable to use real objects, such as reference books, telephone headsets, etc., that are used on the job; items such as telephone headsets provide realism to the training situation, while items like reference books provide an opportunity for practice using the actual object, reducing the problems of transfer (in contrast, say, to providing the same information embedded in the instructional materials).

From an instructional standpoint, it is highly desirable to include software tools from the trainees’ work environment in the tutor, so that, for example, trainees practice with and the tutor observes trainees working with the same

---

2 Regrettably, it is true that some decision-makers never get beyond their first impressions.
database they will later use on the job. However desirable this may be, it does not appear practical to use actual software tools from the job environment within the ITS itself (Lefkowitz & Farrell, 1991). Building a wrapper that reports the trainee’s actions in the context of the tool is an extremely difficult task (Lefkowitz, personal communication, July, 1993); one we had to give up in the LEAP project. The simple information that the trainee has entered or exited the tool is inadequate for tutoring. Unless the tool is homegrown (made by the same in-house software development group that built the tutor), it is unlikely that it will be possible to put hooks inside the tool that permit it to send information to the tutor about what the trainee is doing. Therefore it appears that the best approach for ITS developers is to simulate the software tools that trainees will interact with. Replicating just the functionality of the actual software that is needed for training will keep cost to a minimum. To restate this last point: It is very difficult to establish meaningful communication between existing software used on the job and an ITS, however desirable it might be from an instructional perspective; the only alternative is to simulate it.

6.2.9 Design: Simulation

In contrast to the previous section, in this section I conclude that for tutoring purposes, a simulated environment is better in some ways, especially for beginners, than the actual environment, and that more realism in the simulated environment is not necessarily better.

There is a trade-off between training in the actual work environment and training in a simulated one. The actual work environment provides a vast space and resources for study, but it is not under the tutor’s control; both events in the

3 The appearance, the functionality, and the versatility of the actual software (rather than a simulated, limited version of it) may be a plus for advanced trainees, and for more experienced trainees reduces the frustration of not finding functionality they are accustomed to.
environment and trainee’s actions in it can be impossibly difficult for the tutor to interpret. A simulated work environment is a small (albeit expensive) space completely under the tutor’s control. Consequently, trainee actions in a simulated environment are interpretable by the tutor (a prerequisite for tutoring to take place!). Simulations should be appropriate for learning; they need not, indeed should not, be precise duplicates of actual work situations; instead they should bring out those features trainees need to become aware of.

One might assume that the greater the fidelity of training simulations to the work environment, the better; however, two events in LEAP showed otherwise. Conversations in LEAP are generally briefer than actual conversations\(^4\). LEAP conversations sometimes substitute one-liners for what would be sub-conversations (e.g., Would Tuesday be OK? Yes.), they skip over tasks unrelated to the current subject matter (e.g., Consider subtask X done.) They do not ramble, loop, or contain any chit-chat. These conversations call for ‘the suspension of disbelief’ yet trainees, trainers, and content evaluators found them fully appropriate for training.

LEAP’s SONAR database simulation reproduced only those screens and functionalities of the database necessary for training. The simulated SONAR was slow. To most trainees it didn’t matter, but for those skilled at keyboarding, the simulation’s slowness tended to cause them to err and they complained. Surprisingly, the instructors showed no sympathy for these trainees’ complaints; they said it was just like the real SONAR!

\(^4\) It was not permissible to record actual conversations with customers. It was permissible to listen to these conversations, and to record simulated conversations, where expert reps played the role of customers.
6.3 Evaluation

In this section I briefly make two recommendations regarding the scope and method of ITS evaluation, then present a longer discussion of the difficulties of evaluating the student model.

6.3.1 Evaluation: Scale

Tutors that are designed to do full, complex tutoring cannot be fully evaluated in one-hour nor even one-day sessions. Although LEAP underwent a large-scale evaluation by ITS standards, a longer-term evaluation would have revealed more, since use of the tutor’s features can be expected to change over time. For example, trainees, after they acquire experience with whole conversations in LEAP, and have internalized the structures of conversations and can provide context themselves, might use Practice mode less and Explore mode more. Also, evaluating a larger sample of trainees might reveal more useful information (such as significant characteristics of trainee sub-groups).

6.3.2 Evaluation: Log

Frequency of feature use (as logged) and perceived feature value (as reported) are two independent measures. Trainees who rated a feature as more valuable did not necessarily use the feature more than trainees who rated the feature as less valuable. Therefore it is important to log actual feature use and to gather data independently on perceived feature value, since they do not correlate.

6.3.3 Evaluation: Student Model

Evaluating the student model is a reasonable and desirable goal. The problem is: To what should one compare it? The student model is a detailed compilation of the trainee’s knowledge based on long-term, repeated observation of the trainee’s every action. No other measure of trainee knowledge can match it in terms of length and detail of observation.
The common method of evaluating the student model, i.e., by comparing it to the trainee's post-test results, has some value, since test construction has a long history and demonstrably valid post-tests can be constructed. However, tests are different from student models in two ways: a test is based on a sample of the trainee's results, while the student model is based on a census of a trainee's actions. Similar results from the student model and the post-test would support the validity of the student model, but differing results would not be helpful for diagnosing the nature of problems with the student model. For example, when Corbett & Anderson (1992) report the difference between their student model's claim of mastery learning and the post-test result of non-mastery, they can only speculate on the causes (and they can think of three!). In this respect, a comparison of the student model with post-test results is a summative evaluation of the student model, suitable for confirming success, but not for isolating difficulties.

One distinction to bear in mind when evaluating a student model is the difference between trainee data used in the model and the student model itself. Trainee data is, for example, the information that the trainee has used rule X correctly five consecutive times. The student model is a calculation or conclusion, based on the data, for example, the conclusion that the trainee 'knows' rule X. While the data is undoubtedly correct, the student model may or may not be, since it may be based on numerous, perhaps unarticulated, assumptions. For example, the conclusion of 'knows' rule X implies that the trainee will use the rule correctly at all subsequent opportunities. Yet applying the rule correctly in new circumstance requires recognition of the opportunity, and recollection of the rule.

There are other complications that make determining the student model from trainee data less than straightforward. For example, different kinds of action may require different amounts of effort to learn (pressing 'Enter' versus recognizing and responding to a sales opportunity), so different methods of deter-
mining student model values from student data need to be determined and applied. Anderson proposes that, correctly written, all rules require about the same amount of effort to learn. I am more inclined to break actions into chunks that perhaps combine more than one rule, but that seem natural to the domain: e.g., “May I have your name and telephone number starting with area code, please?” or “Let me give you that number. Do you have a pen and paper handy?” Also, since rules are the action tip of a knowledge iceberg, trainees may not have acquired the knowledge for correct application of the rule in every circumstance. Finally, student model calculations may be based on generally valid parameters that do not apply to a particular sub-group of trainees.

In sum, there are many factors that could affect the student model as it is calculated or inferred from trainee history data, and only careful analysis together with empirical formative evaluation can reveal these factors.

6.4 Authoring

Most ITSs are built by writing a large amount of code, a resource-intensive process; authoring tools and tutoring shells promise to reduce the resources required to build tutors. Authoring tools are software tools for entering and editing an ITS’s domain and tutoring knowledge. Domain and tutoring knowledge are entered into a shell, a set of software modules that accepts and tutors the output of the authoring tools. Shells for some domains may include a set of default tutoring skills. Both domain knowledge and tutoring knowledge should be represented in a form that is authorable by instructional designers, who, as a rule are non-programmers. The explicit declarative representation of domain and tutoring knowledge in a format that is meaningful to both humans and computers is what makes intelligent tutors intelligent. Constructing authoring tools for entering and editing this knowledge is an important aspect of intelligent tutor research.
The capability of an ITS shell to apply instructional decision-making skills at runtime means that final instructional design decisions need not be made at development time. Instead, what needs to be done is to encode domain expertise (a non-trivial task since it requires formalizing domain knowledge to some extent), to input instructional resources such as presentations and exercises, and to input or edit tutoring skills if necessary. The ITS will then use the domain knowledge and instructional resources together with its tutoring skills to optimize learning.

6.4.1 Need for Domain Authoring Tools in Practical Tutors

Authoring tools are important for a practical tutor if, over the lifetime of the tutor, any of the following could change:

- the domain knowledge itself
- the knowledge, skills, aptitudes, attitudes, or values of incoming trainees
- the purpose or goal of the instruction
- the resources of instruction, or the allotment of instructional tasks among the resources of instruction
- the human perspective of how the tutor does or should work

In other words, authoring tools are required for practical tutors. Authoring tools are also required for experimental tutors if they are going to be more than single-use experimental systems. Authoring tools are required even for these systems if the domain knowledge is more than some minimal amount.

6.4.2 Tutoring Shells

Authoring tools imply the existence of a tutoring shell. A tutoring shell is composed of the software modules that tutors for two (or more) different domains, or parts of a domain, have in common, i.e., the modules that are reusable from domain to domain; the more similar the domains, the more modules their
between authoring and running the system, in order to make the effects of the authored materials visible. These capabilities include the ability to pause the system after seeing a problem, make a change, back up, restart, etc.

Apart from authoring the domain and tutoring knowledge bases, the author needs to be able to edit the student model, since testing the tutor requires observing the tutor’s response to the student model in various states.

6.4.4 Authoring Multimedia in Practical Tutors

Because job and task knowledge changes rapidly in the business world, easy revision of domain knowledge and tutoring knowledge is a requirement. For example, recorded speech requires re-recording when the content of the speech changes, whereas synthesized speech requires only re-typing. In LEAP’s case, since recorded speech was an instructional necessity, the effort of ‘making a few edits’ in LEAP’s conversations is illustrative. The edits took one person only a short time to research and input as text, but since these edits affected the conversational part of the domain expertise, and the experts who made the original recordings were no longer available, every conversation that had even a single change had to be entirely re-recorded! The logistical effort of getting a half-dozen people into a recording room with all the associated equipment and its operators, together with the editing and programming effort to put the results into the tutor can be imagined. Expensive, difficult-to-modify, multi-media is unsuitable for practical tutors.

6.4.5 Creating and Modifying Tutoring Strategies

A simple yet flexible approach to the design problem of permitting easy modification of tutoring strategies is to provide authors with sliders that change parameter weights. Changing parameter weights, as done in LEAP, is an idea used in some of the earliest intelligent systems, such as checkers and backgammon programs. In these systems, the move evaluator was an equation with a
served and quantified, and actual cost/benefit figures will gradually become available.

Calculating costs of training programs is straightforward, calculating benefits is not, mainly because it is very difficult to establish cause and effect relationships in the real, complex business world. The easiest benefits to observe are reduced training time and reduced training development time. Benefits such as improved service and improved quality are also observable, but it is harder to attribute them solely to training since many other factors, such as the systems of supervision and reward, also influence them. It is highly desirable to attribute increased sales to training, but sales are influenced by many factors, including factors outside the control of the organization.

In LEAP'S case, (a) the need to train large numbers of trainees, (b) the distribution of trainees over a vast geographic area, (c) the requirement for small group instruction, and (d) the need for long training times, all combined to make current training immensely expensive. Thus LEAP'S potential ability to (a) deliver instruction electronically, (b) over the network, (c) to individualize instruction, (d) and to cut training time, while (e) ensuring uniform results of training, is immensely appealing to the client.

Furthermore, besides the need for training new service reps, there is a frequent need at U S WEST for further training for even larger numbers of existing service reps whenever new products and services are introduced, or existing ones are improved, dropped, combined, etc. Again, LEAP’s potential to deliver individualized instruction to reps’ desktops over the network at a time of their choosing, and to ensure uniform results, has immense practical appeal.

The development and deployment of a functionally practical tutor appears to be doable. To the extent that benefits of training can be measured, the benefits of such a tutor can be measured. However, the issue of cost cannot be explored
by building one-of-a-kind tutors; they are in all probability simply too costly for any application. The cost to be considered is the cost of putting domain expertise into a tutoring shell, so that the cost of building the shell is amortized over many tutors, and the per-tutor cost is based primarily on the cost of inputting new domain expertise. Determining the cost/benefit of tutors is a part of exploring the practical side of tutor-building.

6.6 Summary

In this chapter I have presented a general discussion of what I have learned about building intelligent tutoring systems, referring to LEAP to exemplify points. First, AI researchers using the domain of tutoring have different goals from instructional researchers in the domain of tutoring, and builders of practical tutors have concerns unique to themselves. Second, I made a number of recommendations regarding stakeholders, trainees, training design options, domain expertise representation, the tutoring instructional design process, control of the learning process, feedback, multi-media and simulations. In the final sections of the chapter I concluded that evaluation of the student model is an important research task, that shells and authoring tools are required for both research-oriented and practical tutors, and that it is still too early for cost/benefit analysis of ITS.
This chapter presents results and contributions from the LEAP ITS research project. The project attained its design, implementation, and evaluation objectives, surpassed conventional computer-based instructional alternatives, and was embraced by its stakeholders. This research makes four contributions to the intelligent tutoring field: Focused Practice is a new instructional method, LEAP is a benchmark example of a successfully fielded practical tutor, the LEAP project is an example of a design approach taking into account theories of situated cognition, socio-technical systems, and participatory design, and the LEAP project has shown the advantages of a new paradigm for ITS research: shifting from a computational to an instructional focus. The final section of the chapter suggests a number of improvements in LEAP and proposes five methods to improve the tutoring process, including one which guides learners in developing good learning practices.

7.1 Results

This section briefly presents three main results of this work, the attainment of the project's objectives, the advance of computational instruction beyond conventional computer-based training, and the embracing of the tutor by the members of the organization for whom it was developed.

7.1.1 Attainment of Objectives

Tables 7.1, 7.2, and 7.3 restate the design, implementation, and evaluation objectives of the LEAP project, together with the chapter and section where the achievement of the objective is discussed. Although they receive brief treatment here, their achievement is not insubstantial.
<table>
<thead>
<tr>
<th>Design Objective</th>
<th>Where Discussed: Chapter &amp; Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Design a complete practical tutor</td>
<td>4. Design of LEAP</td>
</tr>
<tr>
<td>- Simulate customers and database software</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Model an expert service representative’s behavior</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Track trainee progress using a student model</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Make appropriate tutoring suggestions</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Embody one or more theories of instruction</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>• Design a set of authoring tools</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Tools for inputting new domain knowledge</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>- Tools for modifying tutoring skills</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td>• Design a general-purpose tutoring shell</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td></td>
<td>4.5. Authoring in LEAP</td>
</tr>
</tbody>
</table>
Table 7.2
Implementation Objectives

<table>
<thead>
<tr>
<th>Implementation Objective</th>
<th>Where Discussed: Chapter &amp; Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Build the items listed above in the design objectives</td>
<td>4. Design of LEAP, Appendix A: LEAP Walkthrough</td>
</tr>
<tr>
<td>• Put an ITS into use in a work environment</td>
<td>5. Evaluation</td>
</tr>
<tr>
<td>• Demonstrate a practical ITS, or</td>
<td>4. Design of LEAP, Evaluation</td>
</tr>
<tr>
<td>• Identify why a practical ITS is not yet feasible</td>
<td>NA</td>
</tr>
<tr>
<td>• Gain support of potential organizational and individual trainees</td>
<td>5. Evaluation, 6. Conclusions and Recommendations</td>
</tr>
</tbody>
</table>
Table 7.3
Evaluation Objectives

<table>
<thead>
<tr>
<th>Evaluation Objective</th>
<th>Where Discussed: Chapter &amp; Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Show that trainees learn from the tutor</td>
<td>5. Evaluation:</td>
</tr>
<tr>
<td></td>
<td>5.5. LEAP Usage</td>
</tr>
<tr>
<td>• Show that trainers elect to use LEAP in their classes</td>
<td>5. Evaluation:</td>
</tr>
<tr>
<td></td>
<td>5.2.2. Instructor Anticipations,</td>
</tr>
<tr>
<td></td>
<td>5. Evaluation:</td>
</tr>
<tr>
<td></td>
<td>5.8. Instructors’ Reflections</td>
</tr>
<tr>
<td>• Demonstrate that instructional designers can use the tools</td>
<td>2 Overview of the Research:</td>
</tr>
<tr>
<td></td>
<td>2.4.2. Evaluation Procedures</td>
</tr>
<tr>
<td>• Show the ability to apply instructional design theory in ITS construction</td>
<td>4. Design of LEAP:</td>
</tr>
<tr>
<td></td>
<td>4.4.6. Tutoring Decisions Summary</td>
</tr>
<tr>
<td>• Document the adequacy/inadequacy of tutoring methods and strategies</td>
<td>5. Evaluation,</td>
</tr>
<tr>
<td></td>
<td>6. Conclusions and Recommendations,</td>
</tr>
<tr>
<td></td>
<td>and this chapter</td>
</tr>
</tbody>
</table>

7.1.2 LEAP Goes beyond CBT

This section describes how LEAP goes beyond conventional computer based training (CBT) and how it matters to the trainees in terms of their perception of value and use of features that are only available from an ITS (e.g., Recommend, Focused Practice).

LEAP goes beyond CBT fundamentally in having an explicit declarative representation of domain knowledge, primarily the SA pair, plus conversations and topics. The SA pair in turn makes possible LEAP’s expert model and student model. These in turn permit the characterization and scoring of topics and conversations in terms of SA pairs, further permitting topic and conversation selec-
tion and sequencing by the tutoring module. A detailed understanding of the current context of SA pair presentation in terms of topic, conversation, and student model information makes possible the tutoring module’s decision to practice, review, skim, or scaffold each SA pair each time it appears. The representation of knowledge in SA pairs similarly makes possible the automatic generation of hints and feedback. A CBT version of the tutor could present conversations in Observe and Full Practice modes, but not in Focused Practice. Nor could a CBT version of the tutor calculate topic Score or conversation Challenge, since it would have no means of determining which portions of a conversation belonged to which topic, nor which portions of one conversation were shared by others.

LEAP’s ability to do more than CBT mattered to trainees. For example, in Focused Practice, trainees believed LEAP expected them to respond just often enough. That is, the increment of new learning per exercise was neither too large nor too small, SA pairs were neither over- nor under-practiced, nor were they over- nor under-reviewed. The complex functionality of Focused Practice would be nearly impossible in CBT. Trainees also rated the value of the Recommend feature highly, a feature which would be impossible for CBT. Trainees rated the value of the topic Score and conversation Challenge as highly useful to their learning, and, as mentioned, a CBT version could not calculate these.

A CBT version would be unable to generate a list of random responses for each conversation action. If the same set of responses were to appear with each repetition of the conversation, the correct response would quickly become guessable by trainees. The graduated hints generated automatically by LEAP that trainees perceived as extremely useful and used often, would have meant a vast, clearly impractical, amount of work for a CBT author.

Trainees rated the Conversation History window and its features very highly. A CBT version of LEAP could not have linked trainee responses with
their SA pairs and saved them for review, nor presented features like Hear Original and Hear Myself.

A CBT version of the tutor could not have Explore (Examine Contact Flow), except as a completely separate implementation from the conversations, an arduous undertaking.

While there is no single aspect of LEAP that could not be replicated in some manner by CBT, it would be a Herculean feat to replicate all of LEAP’s features and versatility by conventional programming methods. LEAP’s ability to individualize instruction would simply not be possible in a CBT version of the instruction.

7.1.3 LEAP Was Accepted by Its Users

LEAP was accepted by its users. Not only was LEAP perceived as desirable by trainees for their learning, by instructors for their initial training and continuation training classes, and by instructional designers, as documented above; the trainees’ union (Communications Workers of America), and management (U S WEST Home & Personal Services), the trainers’ management (U S WEST Learning), and the research department (U S WEST Advanced Technologies), all agreed that the LEAP project was worth continuing. Advanced Technologies (AT) asks its customers, that is, the other U S WEST organizations for whom it performs research, to rate its performance on each project using a standard score card. The LEAP tutor received the highest score ever given an Advanced Technologies project.

A second, soon-to-be-fielded, version of LEAP, LEAP II was funded. LEAP II has a development team several times larger than the original team. U S WEST also decided that LEAP was a unique item, worth an attempt to patent, and a patent has been applied for (J. M. Szuma, personal communication, May 5, 1994).
7.2 Contributions

In this section I describe four contributions this research makes to the intelligent tutoring field. First, Focused Practice is a new instructional method, made possible by computer technology; next, LEAP serves as a benchmark example of a practical tutor; third, the LEAP project serves as an example of the socio-technical systems approach; and finally, LEAP is an example of a different paradigm of ITS research, research that does not claim to advance artificial intelligence, but to advance computational instructional design.

7.2.1 Focused Practice: A New Instructional Method

Focused Practice is a new instructional method. The details of Focused Practice are discussed in the Design section and summarized briefly in this paragraph. Unique to LEAP, Focused Practice keeps trainees working at the edge of their competence by modifying exercises to an appropriate level of difficulty based on the trainee’s current capabilities and learning goals. Focused Practice permits complex realistic tasks to be used as exercises in a tutor: the tutor modifies them to an instructionally appropriate simpler form, while retaining the fully realistic context. Focused Practice is made up of trainee-performed SA pairs (Practice and Review), and tutor-performed SA pairs (Skim and Scaffold). An instructional method as complex and dynamic as Focused Practice is only possible with ITS technology. When combined with LEAP’s exercise selection process based on an overall assessment of a trainee’s score in each SA pair, and combined with LEAP’s feedback and graduated hints, Focused Practice provides trainees with a new, powerful, individualized learning experience.

7.2.2 Practical Tutor: A Benchmark Example

Another contribution is one “YES!” response to the research question: Is it possible to construct an ITS for use in real settings, given the current state of computer technology and ITS design knowledge? LEAP provides an existence
proof of a practical tutor. In this section I list several general and specific characteristics of a practical tutor. As discussed in previous chapters, LEAP exhibits these characteristics.

General characteristics of a practical tutor:

- Is used in a real setting. A practical ITS must teach in a way that serves the larger instructional process (i.e., an existing training program), of which it must be an integral part.

LEAP was designed to be part of an existing training program, and functioned in a real setting. See Chapter 5.Evaluation, sections 5.1. Methodology and 5.2. Initial Conditions.

- Fits into the current instructional social context, i.e., trainees and experienced workers want to use it, trainers and instructional designers want to work with it, management and union want it.

LEAP’s evaluation has shown that its users want to continue using it. See this chapter, section 7.1.3. LEAP Was Accepted by Its Users.

- Exhibits potential financial feasibility:
  - Is a shell, refillable with more instructional materials of a similar sort.
  - Addresses an appropriate instructional task (a domain with a large number of trainees and a high per-trainee cost).

LEAP appears to be potentially financially feasible at this point.

- Has a well-designed interface.

LEAP’s interface was refined over many iterations; during field-trials very few problems were attributable to the interface. See Chapter 5.Evaluation, sections 5.3. General Affective and Usability Results.

- Has software that works correctly and dependably and does not crash when trainees work with it.

LEAP did not crash (after the first day of field trials).
Tutor-specific characteristics of a practical tutor:

- Has a knowledge representation suitable to serve as a shell; simple enough for authoring by an instructional designer who is not a computer scientist.

LEAP’s knowledge representation of Situation-Action Pairs, topic networks and conversation sequences, serves the domain and authoring needs well. See Chapter 4. Design, section 4.2. The Domain Representation.

LEAP can be considered a shell for customer service representative tasks, tasks involving simultaneous conversation with a customer and interaction with a database. See Chapter 4. Design, section 4.5. Authoring in LEAP.

- Has a student model capable of being modified by an instructional designer who is not a computer scientist.

LEAP has a functional, comprehensible, adjustable student model. See Chapter 4. Design, section 4.3. The Student Model.

- Has a substantial amount of domain expertise for trainees to acquire.

LEAP had, even on its first field test, more than two full days of domain knowledge in it. (The average trainee reached a score of 50% on half the material in one full day of use.) See Chapter 5. Evaluation, section 5.5. LEAP Usage.

- Has a substantial amount of instructional skill, enough to tutor in a wide variety of circumstances, so as to individualize instruction. A practical ITS must teach in a way that challenges, supports, respects, and engages its trainees.

LEAP had enough variety of instructional skills that not all were used extensively in the day-long field trial. See Chapter 5. Evaluation, section 5.5. LEAP Usage.

- Has a substantial amount of instructional resources (presentations, exercises, hints, feedback, etc.) for the instructional capabilities to draw upon as needed, and to enable prolonged practice.

LEAP had a number of presentations, exercises, exploratory areas, etc., so that trainees could see all SA pairs in several contexts. See Appendix A. LEAP Walkthrough.
Attempting to build a practical tutor sheds light on the distinction, if any, between practical tutors and the current state of the art. While LEAP would benefit from both practical improvements and further research (see Further Work), it is clear that LEAP is already, in functional terms, a practical tutor.

7.2.3 ITS Research: A Socio-Technical Systems Approach

Clancey (1993), one of the more influential ITS researchers, stopped doing ITS research for several years when he realized none of his programs were being used for instruction; he focused instead on how theories of situated cognition, the socio-technical systems approach, and the practice of participatory design should influence ITS design. What Clancey has proposed as a result may be radical to AI researchers; it is standard practice for instructional designers. Table 7.4 presents Clancey’s proposals for the practice of research, and our process for developing LEAP.
Table 7.4
Design Objectives

<table>
<thead>
<tr>
<th>How Clancey (1993, p. 17) would develop Guidon-Manage today:</th>
<th>How we developed the LEAP ITS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participating with users in multidisciplinary design teams <strong>versus</strong> viewing teachers and students as my subjects.</td>
<td>LEAP’s core design team consisted of an instructional designer (me), a computer scientist, and a cognitive psychologist. We worked closely, in regularly scheduled, frequent, and prolonged contact with beginning and expert service reps, their instructors, instructional designers, union, and management assigned to the LEAP project. I began my study of the job tasks of a service rep and the learning task of becoming a service rep before knowing the particular task LEAP would teach. The instructional task of teaching voice messaging was determined not by the research team but by the customer - who would eventually use the resulting tutor.</td>
</tr>
<tr>
<td>Adopting a global view of the context in which a computer system will be used <strong>versus</strong> delivering a program in a computer box.</td>
<td>As LEAP’s instructional designer I spent a lot of time, not only with the people, but also in the places where the tutor would be used. A primary concern throughout the development process was that the tutor be integrated technically and socially into both initial training, taking place in training suites, and continuation training, taking place on service rep’s desks.</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>How Clancey (1993, p. 17) would develop Guidon-Manage today:</th>
<th>How we developed the LEAP ITS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Being committed to provide cost-effective solutions for real problems versus imposing my research agenda on another community.</td>
<td>Building a practical tutor was one of the two main objectives of the project. The customer agreed to the tutoring approach because it appeared to have the potential to address the long-term training need. The customer selected the domain because it was a genuine training problem. Tutoring research took place in the context of addressing the customer's need.</td>
</tr>
<tr>
<td>Facilitating conversations between people versus only automating human roles.</td>
<td>Attempting to make expert knowledge explicit in the detail required for it to be executable, promotes a good deal of discussion among experts representing various interests. Distributing the expertise electronically company-wide promotes the same sort of discussion.</td>
</tr>
<tr>
<td></td>
<td>A portion of the instructor's role was automated; the automation served to augment and extend the instructor, not to replace him or her. The tutor can be used for about one-quarter of class time: time where the instructor previously had to simultaneously observe role plays and provide feedback to six individuals, an impossibly difficult task.</td>
</tr>
</tbody>
</table>

Continued, next page
Table 7.4, continued

<table>
<thead>
<tr>
<th>How Clancey (1993, p. 17) would develop Guidon-Manage today:</th>
<th>How we developed the LEAP ITS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Realizing that transparency and ease of use is a relation between an artifact and a community of practice <strong>versus</strong> an objective property of data structures or graphic designs.</td>
<td>In LEAP I tried hard to make the simulated conversations similar to (though not identical to) actual customer-expert rep conversations, even at the cost of ease of learning and ease of student modeling. The conversations had to feel authentic to trainees and instructors for the tutor to be accepted in the classroom.</td>
</tr>
<tr>
<td>Relating schema models and ITS computer systems to the everyday practice by which they are given meaning and modified <strong>versus</strong> viewing models and programs as constituting the essence of expert knowledge that is to be transferred to a student.</td>
<td>During development, we had a number of people use the tutor, from ITS experts to man-in-the-street novices; the majority of users were novice and expert service reps who used the tutor for many iterations. Usability was improved until the field trials were underway.</td>
</tr>
<tr>
<td>(Clancey elaborates this point by saying &quot;... today we view ... [the disease taxonomy of Neomycin] not as a product to be delivered to a student, but as a partial model of a practice&quot;).</td>
<td>LEAP very nicely related abstract SA pairs to everyday practice, where they appear in conversations. The SA pairs themselves could only be seen in Explore mode, where instances could be also be accessed with a mouse-click.</td>
</tr>
<tr>
<td>In designing LEAP I did view the domain knowledge base as a product to be delivered, while simultaneously being fully aware of it as a partial model of practice, in the sense of knowing that after mastering the training materials, trainees still have a lot to learn. Clancey (1993) makes some excellent points regarding further knowledge that should be learned along with any model. That knowledge is not best taught by a tutor and is not included in LEAP.</td>
<td>Continued, next page</td>
</tr>
</tbody>
</table>
How Clancey (1993, p. 17) would develop Guidon-Manage today:

Viewing the group as a psychological unit versus modeling only individual behavior.

(Clancey elaborates this point by saying "... learning for the individual is becoming a member of a community of practice").

How we developed the LEAP ITS:

LEAP clearly emphasizes individual behavior; but then LEAP does not pretend to produce experts in the field, it aims instead for producing competent beginners. From an instructional designer's perspective, it is obvious that an ITS cannot produce experts. Traditional simulator-based training, for example, begins with experts. For novices, LEAP opens the door to the community of practice.

To a large extent, the LEAP project serves as an example of the socio-technical systems approach Clancey advocates. I believe the positive outcome of the LEAP project, i.e., its acceptance by users and its continuation, is due in large part to this approach. In the next section I propose a new paradigm for ITS research, shifting, in Clancey's (1993) terms from "... exploring what a computer can do to determining through practice which designs are useful" (p. 7). The only way to do research on the tutoring capabilities of an ITS is to have trainees use it. A research tutor need not be a practical tutor though it must have many of the characteristics of a practical tutor if actual trainees are to use it. I elaborate on these distinctions in the next section.

7.2.4 ITS Research: A New Paradigm

In an earlier chapter I distinguished among three sorts of ITS research leading to practical tutors:

- AI research in the domain of intelligent tutoring systems
- Instructional (or tutoring) research in the domain of tutoring
• Research on practical tutors

In this section I begin by proclaiming the need for a paradigm shift in ITS research from AI research in the domain of ITS, to instructional, or tutoring, research in the domain of computational instructional design. I then justify the need for the paradigm shift, and present LEAP as an example of this shift. Finally I discuss the effects of the proposed paradigm shift, and describe some of the questions this shift opens to investigation.

7.2.4.1 The Need for a Paradigm Shift in ITS Research

Shifts in paradigms mean not only new ideas but also shifts in what constitutes good problems, what counts as a result, the experimental practice to validate results, and the technological tools needed to do research. ("Current Titles," 1994, page 8)

The time is ripe for a paradigm shift in ITS research. The practitioners of the new paradigm will view the domain of intelligent tutoring not as a place to do interesting AI research, but as a place to do research in tutoring strategies and other instructional issues.

The answer to the question of what constitutes good problems in ITS research should shift from interesting AI problems to interesting instructional problems. From "Can I build something that is intelligent and looks like tutoring?" to "Can I build something intelligent that is tutoring? (i.e., to something that takes actions that are genuinely useful for instruction, as shown by trainee-testing.)" From "How do I make my system intelligent?" to "What must I put into the system to ensure that my trainees learn optimally?"

The question of what counts as a result should shift from "Does the software run?" to "Do the trainees learn?" From "Is the software intelligent?" to

1 I am not advocating that AI research in ITS cease; it should continue; I am advocating that it now be put to use.
"Are the trainees intelligent?" From "What are software’s capabilities now?" to "What are trainees’ capabilities now?"

The experimental practice to validate results should shift from running the software and demonstrating how it handles an event, to having a group of trainees use the software and measuring its effect on their learning. AI researchers have demonstrated a large number of potentially interesting instructional possibilities, but they have little interest in, nor any idea of how to go about, applying them. Conversely, instructional researchers are concerned with how software teaches, they are not concerned with AI problems. Tutoring methods from many AI research tutors may work well; the only way to find out is by evaluating these methods to see whether trainees learn from them. The only way to create tutoring methods that do work well is to iterate through the build, test, revise cycle.

The technological tools needed to do research should shift from software that demonstrates a concept to software that tutors trainees; from software in a computer science laboratory to software in a classroom, on trainees’ desks. Outcomes should be measured not by observing software demonstrations, but by observing trainees’ rates of learning.

This call for a paradigm shift is not a request for building practical tutors, though software used by trainees must necessarily share many of the characteristics of practical tutors; nor is it a call for applied research, but for research about those aspects of machine-mediated instruction involving instructional intervention, i.e., about tutoring strategies implementable in (perhaps only in) and utilizable by ITSs, that successfully tutor humans.

7.2.4.2 Why a Paradigm Shift in ITS Research Is Needed

We need a paradigm shift from AI research in tutoring to instructional research in tutoring because there are many ITS research questions that are not AI
questions, but are instructional design issues. AI researchers cannot be expected
to address them. These questions fall mainly into two categories, representation
of domain expertise for instruction and representation of tutoring expertise for
instruction.

Concerning the issue of representing domain expertise for instruction
there are three points. First, Anderson’s (1990) cognitively plausible rule-based
representations of domain expertise are a success. The difficulties with this ap-
proach lie first in the cost of building these representations, second in the limited
domains to which they seem applicable, and third in the requirement for empiri-
cal validity-testing of the rules with each type of learner who will use them.

It may be possible that less expensive domain representations will work as
well for instruction. Murray (1991) observes that less inference-capable represen-
tations of domain knowledge are much less expensive to construct, and LEAP’s
low-inferencing representation worked well. Instructional research in ITS will
address this question.

There are many domains for which the rule-based approach may not be
applicable, such as the domain of argumentation (Cheikes & Linton, 1995), where
knowledge may characterized as incomplete, inconclusive, and inconsistent.
Representations of expertise suitable for instructing in these domains are yet to
be discovered.

There are also problems that we cannot wait for AI researchers to solve,
problems that are simply too difficult for the state of the art, such as speech un-
derstanding, natural language processing, and inferring user’s plans from ac-
tions. Instructional researchers can find work-arounds2 for these problems. A tu-
ton need not be intelligent in every way.

---
2 Example work-arounds from LEAP are: evaluating student’s selections from a randomly
generated list of responses in place of natural language understanding, coaching students on pre-
Finally there is the problem of utilizing the results AI researchers have already produced. These results are waiting to be put to use. AI researchers have demonstrated the possibilities; instructional researchers must now demonstrate the applicability of this work to instruction. AI research in tutoring does not address the specific goal of implementing programs that demonstrably help trainees learn.

Once a representation of domain expertise has been constructed in a form that is suitable for instruction, the task of creating a representation of tutoring expertise for it remains. As mentioned, this is not an AI question, but an instructional design issue. Tutors must be built to solve trainees’ problems, not interesting AI problems. Instructional researchers begin with their knowledge of instructional design theory, and try to find ways to use AI to do tutoring.

Good tutoring strategies that take advantage of AI techniques are not obvious by inspection; they have to be developed and tested with trainees over time. Even good tutoring strategies are not good under all conditions; nor is there one best way to do things for all trainees, as the range of usage and learning in LEAP illustrates. A range of tutoring strategies will have to be worked out as well as control strategies for choosing among them.

Existing instructional design theories are not adequate for building ITSs. Tutoring theories suitable for ITS do not exist at the appropriate level of detail or prescriptiveness; nor do any theories exist that take advantage of a computer-based tutor’s capabilities. Existing theories should be applied, new theories are required. The results of this paradigm shift will be advances in instructional design theory.

defined exercises instead of attempting to coach novices on actual customer calls, and using prerecorded human speech instead of synthesized speech to model expert performance.

338
7.2.4.3 The LEAP Project Is an Example of the Paradigm Shift.

The LEAP project stands in strong contrast to the typical ITS project, which ends with a demonstration of code that illustrates the implementation of an AI idea (some way to make a computer intelligent). This project eschewed AI research in the sense of demonstrating computer intelligence, but is very much ITS research in the sense of creating and testing tutors with instructional skills whose efficacy were evaluated on humans (find ways to make the trainees intelligent). The results of this project (a tutor that genuinely tutors, a new tutoring method, and a tutor that provides the impetus for a practical tutoring project) illustrate the potential of the proposed paradigm shift.

The LEAP project is an example of how a focus on tutoring skills will enable researchers to formalize, evaluate, and extend existing theories and methods of instruction. One of LEAP's contributions is the construction and evaluation of a tutor with the focus on tutoring, including the application of theories of instruction, and a set of tutoring skills. With a focus on tutoring, the test of the tutor was that trainees learned. LEAP encodes instructional actions compatible with a number of instructional theories, and has a suite of tutoring skills; it also contains adequate domain expertise, but not of the sort Anderson, for example, believes is necessary. Field trial results indicate that trainees do indeed learn from LEAP.

LEAP is an example of the proposed paradigm shift in the sense that success of the tutor was measured in trainee learning. To be sure, trainee learning was not measured in a controlled laboratory test, but in a field test, where learning was demonstrated to the satisfaction of the sponsors of the research (Winne, 1993): trainees, instructors, instructional designers, union, and managers, all of whom share the goal of well-trained, satisfied, and productive workers.
7.2.4.4 Effects of a Paradigm Shift

Three effects of a paradigm shift in ITS research will be: a change of venue, a focus on the issue of shared control, and the input of adequate instructional resources to evaluate tutoring strategies.

Evaluation based on trainee learning requires real trainees; real trainees in turn imply genuine meaningful learning tasks. Thus the proposed paradigm shift means a change in venue of tutoring research from computer science laboratories to trainees' desktops.

One of the effects of the proposed paradigm shift is the surfacing of the issue of shared control. In AI Research in ITS, the issue of shared control does not arise; the trainee is a passive object upon which the intelligent tutor operates. In ITS Research the trainee is an independent agent; the tutor and the trainee share the goal of the trainee’s becoming skilled and share control over the trainee’s process for doing so. Thus the design issues of shared control, including the communication of goals, plans, rationales, etc. become salient research issues.

AI researchers can encode a minimal amount of instructional skill and instructional resources, and still evaluate their ideas; for example, the PROPA ITS initially had only one tutoring strategy (solicited hints) and one exercise (Linton, 1995). In contrast, an ITS for instructional, or tutoring, research would have to have at least one tutoring strategy and adequate resources for testing the strategy (i.e., a mechanism for selecting exercises has to have a number of exercises to select among) as well as a control mechanism that decides when to invoke the strategy. To provide a meaningful context for learning, the tutoring software should also provide a number of presentations, exercises, and reference materials.
7.2.4.5 What the New Focus Could Investigate

Most ITS research to date can be characterized as AI research in ITS. Tutoring research in ITS can investigate the application of these results to the task of tutoring actual trainees. In this section I give examples of a number of questions tutoring research in ITS should investigate.

There are two major questions regarding any representation of domain expertise for tutoring: the representation’s suitability for instruction, and its ease of authoring. Most tutors use expert system-like rules to represent domain expertise, and represent only a minimum of domain expertise. This representation of domain expertise, while powerful, is resource-intensive to construct. One question to investigate is whether it is possible and cost effective to scale up domain expertise using rule-based representations. Another question to investigate is the suitability of this representation for non-formal domains, such as those that compose most job tasks.

"If you can write a domain expert system that does a task, why do you still need to train people to do it?" is an often-asked question. In fact, if you could write an expert system to do a task, you would not need to train people to do it. On the other hand, it is much easier to represent domain expertise in a form suitable for tutoring a task than for actually performing it. First, only a portion of domain expertise need be represented, that necessary for instructing novices; second, the representation does not interact with the complex real world, but with the simplified version of it modeled in the tutor. Writing rule-based domain experts for instruction should be more readily achievable than writing functional expert systems.

Another question is whether the representation of domain expertise need be rule-based at all. For example, LEAP’s expert representation was not. And many domains, such as argumentation (Schum, 1987), are not easily definable by
expert system rules. Murray (1991) makes the further observation that the do-
main expert’s inferencing power is paid for by cost of authoring. Thus a domain
expert should as low-power as possible as long as it is effective for instruction.

A paradigm shift to tutoring research can study the difficulties of author-
ing large knowledge bases. Conventional instructional design can be considered
a form of knowledge engineering. It is reasonable to believe an instructional de-
signer together with a domain expert can represent domain expertise in a form
suitable for novices to learn from an ITS. Authoring tools will not reduce the in-
tellectual effort, but can simplify the task. Tutoring research needs to determine
representations and authoring tools suitable for relatively rapid, high-quality au-
thoring.

A tutor that researches trainee learning will require the representation of
relatively large quantities of domain knowledge (Anderson’s lisp tutor has 500
correct rules and 1500 buggy ones for a semester-long course. LEAP has about
200 SA pairs for a two to four day training course.). Many ITSs for AI research
only have one or a few exercises. It is a tutoring research question to determine
what sort of a task the formalization of large quantities of domain knowledge is.

A paradigm shift to tutoring research can study the interaction between
quality and quantity in a representation, where the quantity of knowledge is
based on ease of authoring and the quality is measured by the suitability of the
representation of domain expertise for instruction.

The paradigm shift can investigate the incremental or synergistic benefit
of adding intelligent features, one by one, to a tutoring system.

As mentioned in the previous section, tutoring research will have to con-
front the issue of integrating trainee and tutor control. One method of giving the
trainee control, without ceasing to make tutoring decisions is passive tutoring.
Instructional researchers could investigate the effect of active versus passive in-
telligent tutoring; for example, instead of selecting exercises for the trainee, the tutor could present the list of exercises together with their rankings on various suitability measures, with explanations for the rankings available upon request, and let the trainee select the exercises. Or, for example, instead of attempting to determine the precise information to present in a hint, the tutor could let the trainee select the amount of information desired. This passive tutoring approach reduces tutor control while making full use of the tutor's intelligence; for example, when hinting, the tutor still has to determine and construct an appropriate hint, and credit the student model.

As Winne (1993) Anderson (1993), and others have observed, evaluations of ITSs are generally of low quality when they are done at all; and there is no consensus on how to improve the process. Tutoring research will, of necessity, address the evaluation issue.

The LEAP project serves as a benchmark illustration for some of the issues itemized above. LEAP was developed for a domain of the client's choosing, a non-formal domain, using a representation of expertise that was not rule-based, one that domain experts and instructional designers could understand and use. LEAP's domain inferencing power was low, yet trainees learned. LEAP's domain expert cannot actually perform the task, but is suitable for instruction. LEAP has an adequate authoring system. LEAP contained a relatively large amount of domain expertise. LEAP sported a complete integration of trainee and tutor control.

LEAP's evaluation is a benchmark for future work. The evaluation measured trainees' affective responses toward LEAP and their perceptions of its usability and instructional value, trainees' usage of LEAP and the resulting learning, LEAP's capacity for individualizing instruction, and instructors' response to having LEAP in their classrooms.
In order to leave the research laboratory, AI systems have to be integrated with conventional, software, such as databases, spreadsheets, etc. It is only reasonable to expect intelligent tutoring skills to be embedded in software that has much in common with conventional computer based training (CBT). Furthermore, no single medium is adequate for a complete instructional package. Instructional research in ITS will have to address software integration issues and instructional package integration issues.

Tutoring research will require a means for editing tutoring strategies with no more difficulty than domain knowledge is edited. This in turn requires an explicit declarative representation of tutoring expertise analogous to the representation of domain expertise. Without such a representation, authoring tutoring expertise requires writing code. Yet few ITS research projects have attempted to represent tutoring expertise in a manner amenable to tutoring research.

In brief, a paradigm shift could investigate representations suitable for instruction and ease of authoring, scale up of representations, representations for non-formal domains, authoring tools, authoring large knowledge bases, effects of intelligent tutoring on learning, control issues, active versus passive tutoring, evaluation issues, software and instructional integration issues, and representing and authoring issues for tutoring strategies.

7.2.4.6 Summary of Paradigm Shift

To summarize, it is time for a paradigm shift in ITS research from interesting AI problems to interesting instructional problems. There are many ITS research questions that are not AI questions, but are instructional design issues. Once AI research in tutoring has illustrated the possibilities, instructional researchers must determine how to apply these techniques to support trainee learning. Tutoring theories suitable for ITSs do not exist at the appropriate level of detail or prescriptiveness; nor do theories exist that take advantage of a
computer-based tutor's capabilities. The LEAP project is an example of how a focus on tutoring skills will enable researchers to formalize, evaluate, and extend existing theories and methods of instruction. The proposed paradigm shift means a change in venue of tutoring research from computer science laboratories to trainee desktops. The paradigm shift will investigate representations suitable for instruction and tutoring, authoring tools, issues of scaling up, representing non-formal domains, tutoring methods, evaluation methods, and numerous other issues that arise when tutoring trainees.

7.2.5 Summary of Contributions

In this section I have described four contributions from the LEAP project to ITS research: First, Focused Practice is a new instructional method made possible by AI research and developed for actually instructing trainees. Second, LEAP is a benchmark example of a practical tutor, it has every appearance of being successfully fielded. Third, LEAP is a benchmark example of a design approach taking into account theories of situated cognition, socio-technical systems, and participatory design. Finally, the LEAP project has shown the advantages of a new paradigm for ITS research: shifting from a computational to an instructional focus.

7.3 Further Work

The further work described in this section is of two types: further work in LEAP, and further research.

7.3.1 Further Work in LEAP

Below is a list of further work that could be done with the LEAP tutor, which if done, would yield potentially interesting research results. Each of the listed points is discussed in more detail in this section.

- Determine authoring cost
• Deliver training over the wide area network
• Improve alternative actions in conversation responses
• Provide tutor-selected branching during Focused Practice in Explore mode
• Improve presentation of a trainee’s knowledge state to the trainee
• Improve presentation of Recommendations to the trainee
• Improve Explore
• Improve the Evaluation Process
• Provide human-computer evaluation of spoken responses to SA pairs
• Perform a follow-up evaluation of the long-term effects of LEAP-based training

7.3.1.1 Determine Authoring Cost

One area of evaluation that needs more data is determining the cost of authoring domain expertise. In the current system, authoring was closely intertwined with development and it was not possible to segregate authoring costs from development costs. Putting a second set of exercises into the existing framework would result in a more precisely determined measure of the cost of developing exercises and related instructional materials.

7.3.1.2 Deliver Training Over the Wide Area Network

One technological change could have a large effect on LEAP’s cost/benefit equation. LEAP is currently delivered over a local network (LAN) to a half-dozen trainees in the same room. This delivery method works well for current classroom-based initial training. A much larger audience exists for, and there is more training material for, workplace-based continuing training, as U S WEST’s workplaces are distributed over 14 western US states. Successfully delivering continuation training to service reps’ desktops over the wide area network (wan)
would provide substantial benefit to the company, as current continuation training suffers from haphazard delivery and irregular quality.

7.3.1.3 Improve Alternative Actions in Conversation Responses

Most of the alternative answers in conversation actions were not based on empirical research of trainee errors, but generated automatically by the tutor which selected actions randomly from the nearby portion of the domain knowledge base. Long term solution: make it as easy as possible for course developers to do the hard empirical work of reviewing and evaluating trainee responses and selecting the common errors that need trapping and remediation by giving them a suitable tool for this task. Short term solution: provide the instructor with means of reviewing responses for the purposes of evaluation and coaching. (Instructor can already ‘sit in’ on conversations as they happen and make live commentary, and while conversations are saved to disk, there is currently no tool for the instructor to retrieve and review them.)

7.3.1.4 Tutor-selected Branching During Focused Practice in Explore Mode

Currently while doing Focused Practice in Explore mode, the trainee always selects which conversation branch to pursue. The tutor makes only the skim/scaffold/practice/review decision. One alternative to investigate is to have the tutor also select the conversation branch, so as to find and present the SA pairs that need practice. One advantage of this approach over that of presenting a diagram of the topic overlaid with the trainee’s scores for each SA pair and letting the trainee pick the path (an alternative discussed later in this section) is that the tutor, knowing the whole path in advance, could pick SA pair instances from a single conversation, and present them along with the more abstract SA pair texts that now appear in Explore mode. (The reader may recall that currently in Explore mode, instances are played to the trainee when she or he clicks on lines in the History window, but these are of necessity randomly selected in-
stances since the tutor cannot select instances corresponding to a path the trainee has not yet picked.)

7.3.1.5 Improve Presentation of a Trainee’s Knowledge State to the Trainee

LEAP’s student model keeps detailed information on every individual SA pair, as well as summative scores on SA pairs, exercises, and topics. It may be possible to display a holistic and detailed view of the student model by using techniques for visualizing large quantities of data from multiple perspectives, such as Shneiderman’s (1994) Treemap and Starfield strategies. In general these strategies provide: an overview of the data, the ability to zoom and filter, and details on demand.

7.3.1.6 Improve Presentation of Recommendations to the Trainee

One way LEAP presents the outcomes of its intelligent instructional decision-making is by making Recommendations. Currently LEAP makes its recommendations one at a time and presents them on a take-it-or-leave-it basis. It is not necessarily clear to the trainee how these recommendations are related to his or her learning goals and to the state of the student model (displayed in topic Scores and conversation Challenges). Using Shneiderman’s (1994) Treemap and Starfield strategies, it may be possible to combine the view of the student model together with the expected outcomes of alternative actions. With displays like Shneiderman’s, recommendations, in the sense of the values of alternative actions given the trainee’s goals and current state of the student model, would be communicated in the display of predicted progress.

7.3.1.7 Improve Explore

Trainees did not spend a lot of time in the advanced mode of Explore, so there is not a lot of data on problems they might encounter there. One potential problem is the lack of a map or overview of each topic, showing its branching
structure. Another is the lack of a student model overlay on such a map so a trainee could see which branches he or she has learned. One suggestion is a graphical representation of topic internal structures with SA pairs color-coded to indicate the strength of knowledge (e.g., black = unknown, green = known). The TreeBrowser (Shneiderman, 1994) might provide some inspiration here. This sort of display for intelligent trainee guidance contrasts with the alternative discussed earlier in this section of Focused Practice in Explore where the tutor guides the trainee through the portion of the topic that needs practice.

7.3.1.8 Improvement in the Evaluation Process

Kirkpatrick (1975) describes four key questions for the evaluation of training materials: Did trainees like the training? Did they learn the material? Do they use it on the job? Does it matter? As is common, the LEAP evaluation addressed only the first two questions. Of particular interest would be a follow-up study comparing the effects of LEAP-based practice to the conventional practice methods used to learn other products. One method for investigating the effect of LEAP-based practice would be to compare the ratio of voice messaging sales to sales of other products for LEAP-trained versus conventionally trained service reps. A second method for investigating the effect of LEAP-based practice would be to use an interview or questionnaire to acquire trainee self-reports of the difference between handling calls relating to voice messaging and calls relating to other products. The attention to completeness and detail, together with repeated practice in LEAP, should have prepared trainees in a manner superior to conventional training. The long-term existence of a training effect, however, and its effect on the bottom-line (i.e., Does it matter?) remain to be established.

7.3.1.9 Provide Human-Computer Evaluation of Spoken Responses to SA Pairs

In SA pairs where the action is of type Talk, trainees must first record their spoken response, then select an ‘equivalent’ response from a tutor-supplied list.
This two-step process is problematic from an instructional perspective, but is necessary because the fields of speech and natural language understanding are not yet sufficiently advanced to understand speech (of course, if computers could understand trainees' speech, they could understand customers' speech, and there would be no need for this training!). The reason this process is problematic from an instructional perspective is that trainees are learning to generate a verbal response, but they are being evaluated on (i.e., their student model is based on) their ability to recognize a written response. Since it is less cognitively demanding to recognize a correct response than it is to recall or generate one, the student model is currently based on a too-simple version of the task. The required improvement is to somehow evaluate trainees' spoken responses on their own merits.

One likely solution is the Response Evaluation Checklist. When using the Response Evaluation Checklist, trainees record their spoken responses then play them back (at the end of the conversation) and evaluate the verbal actions themselves. For example, trainees could use the checklist to evaluate: the propositional content of their responses, the phrasing, the brevity, the cohesiveness, the rambling, the empathy, etc., all in comparison to one or more examples of expert response. Some prior instructor-led training in this aspect of self-evaluation might be best. This solution has two interesting aspects: First, trainees, when evaluating their responses, serve, in effect, as intelligent input devices to the tutor. At the same time, trainees are learning to evaluate their own performance, a skill which will be of value long after formal training is complete.

Besides being a LEAP improvement issue, the Response Evaluation Checklist approach requires further investigation as a research issue. There are many possible ways of using it and the best one is not obvious. For example, trainees need not learn this skill simultaneously with conversation practice; they
could learn it on other trainees' pre-recorded lines, selected for the instructional value of their characteristics. Also, trainees could evaluate each other, and the instructor could evaluate the trainees. Not all spoken lines need evaluation every time. A statistically valid random sample can be used, or just the new lines can be checked, or just the lines with characteristics the trainee needs practice on, etc. There are many possibilities; the tradeoffs of specific approaches have to be evaluated not only for their instructional value, but for their practicality in a genuine training context.

7.3.2 Further ITS Research

Here I discuss some general issues of ITS research raised by the LEAP tutor project. First, I briefly characterize the need to make tutoring expertise as explicit as domain expertise. Then, the major portion of this section discusses five general methods for improving the tutoring process (using LEAP as a context for discussion).

7.3.2.1 Make Tutoring Expertise Explicit

Tutoring expertise is embedded in the tutor's code, not made explicit as the domain knowledge is (except for the weighted factors accessible via the authoring screens). Pulling out the tutoring expertise and making it explicit would be a major contribution to the generalization of the tutoring shell. Tutors tend to have either explicit domain knowledge and embedded tutoring knowledge or explicit tutoring knowledge and text-based domain knowledge. What is needed is to make both types of knowledge explicit: Clancey (1987) made a beginning that has not been followed up. The extent to which an explicit representation of tutoring knowledge is possible is unclear (since the tutoring knowledge acts on the domain knowledge and the tutoring interface) but the creation of explicit tutoring knowledge would essentially be a higher-level language for encoding tutoring theory and practice. It would make the tutoring side of the tutor as acces-
sible to instructional designers as the domain knowledge side is, and would reduce the need for working in concert with an AI programmer to make major changes in this portion of the tutor. A similar idea is implemented by Murray (1991) in his Parameterized Action Networks (PANs).

7.3.2.2 Improve the Tutoring Process

The LEAP tutor has the capacity to respond to a large number of instructional circumstances. We can refer to the tutor’s capacities to respond as its space of instructional responses. Modifying the tutoring process corresponds to revising the boundaries of this response space. (By way of contrast, a different way to modify the tutor is to modify its instructional resources -- to add or remove or change presentations, exercises, etc.) In this section I discuss possibilities for modifying the tutor’s response space.

ITS researchers have taken up the task of creating software that can teach better than before. Still, there is no reason to believe that their initial creations can respond in the optimal way to all instructional circumstances. There is always the possibility that the tutoring process could be improved not only by ITS researchers but also by the ITS itself or by other users. This section describes five methods for improving the tutoring process. The five methods are:

- Instructor modifies tutoring strategies “Teach them like this:”
- Trainee modifies tutoring strategies “Teach me like this:”
- Tutor modifies its domain knowledge (based on trainee solutions) “I’ll teach that:”
- Tutor modifies its tutoring strategies (based on trainee actions) “I’ll teach them like that:”
- Tutor modifies trainees’ learning strategies “Try learning like this:”
7.3.2.2.1 Instructor Modifies Tutoring Strategies

As currently implemented in LEAP, the instructor can access the authoring screens and modify the tutoring strategies either slightly or drastically, simply by moving sliders that change parameter weights. This method is discussed in detail in the section Authoring in LEAP. And while it appears impressive, its suitability, flexibility, comprehensiveness, and utility have yet to be evaluated.

7.3.2.2.2 Trainee Modifies Tutoring Strategies

A feature to be researched is letting trainees adjust some of the weights of the factors used in tutoring decision-making. For example, trainees might reasonably be expected to set the increment of new material presented per exercise as well as the frequency and amount of repetition and review. Coupled with some display of the tutor’s version of the optimal settings (the settings for the optimal learning rate for the average trainee), this feature would let trainees set their own balance of challenge, comfort, and learning style.

7.3.2.2.3 Tutor modifies its domain knowledge (based on trainees’ results)

Many ITS researchers have considered the possibility of a self-improving tutor. Tutors can self-improve in two ways: improve their domain knowledge and improve their teaching knowledge; these paragraphs deal with the former.

For example, Kimball (1982) described a tutor of integral calculus that remembered those solutions to exercises that were better than those already in its domain knowledge base. His method is not applicable in LEAP’s case, since LEAP’s domain knowledge representation is much more complex, and more importantly, LEAP has no way of evaluating a different solution as better. However, it seems reasonable to assume that LEAP’s authors are not all-knowing, and especially for continuation training, it seems important that LEAP at least have the capacity for trainees to communicate with authors, or other experts, in a Question & Answer or Bulletin Board mode. Trainees should not see
the machine as a source of knowledge but as part of a system for communicating knowledge, and when the machine does not have appropriate knowledge stored in its domain knowledge base, it should provide access to a human discussion of the situation. While authoring for the current version of LEAP, experts often disagreed among themselves about correct actions, and during the field trials, instructors sometimes, though rarely, disagreed with the tutor’s ‘right’ answers. Therefore, while LEAP cannot, by itself, modify its domain knowledge base since it has neither access to expert performance nor the capability of recognizing it, LEAP should at least provide trainees with a means of access to knowledge beyond that in its domain knowledge base. The DIME system proposed by NYNEX (R. Radlinski, personal communication, May 1993) is an example of one such approach. Further research, then, is needed to provide ITSs with some facility for human-human communication of expertise, and to provide a forum for discussion and resolution of questions.

A completely different approach is possible when a tutor can observe trainees perform real life tasks in context and build models of their performance. By comparing student models against each other for the same tasks, the tutor can distinguish the more capable performers from the less capable ones and coach the less capable ones to be more like the more capable ones. Machine tutors are not capable of doing a task this complex by themselves; we must assume some human work in conjunction with the tutor’s. Interventions would take the flavor of:

Tutor: Most people use the delete-forward key about twice as often as the backspace-delete key. You use only the backspace-delete key. Would you like a three-minute lesson on the delete-forward key?

Tutor: Most advanced users begin with a Stationery Form when they write letters. Using Stationery lets them write letters in about 80% of the usual
time and maintains a consistent format. Would you like a brief lesson on creating and using stationery?

To summarize, it seems unlikely that a tutor like LEAP could self-improve its domain knowledge base, so other ways of communicating more-expert expertise should be provided in the tutor. On the other hand, improving the tutoring strategies seems a real possibility, and the method is discussed next.

7.3.2.2.4 Tutor Modifies Its Tutoring Strategies (Based on Trainees’ Actions)

There is no reason to assume that a tutor’s instructional methods are optimal for all learners and learning situations. Since a tutor’s methods can be improved upon, the tutor should have some method for improving its tutoring skills.

Also, even if a tutor’s methods were optimal for a given set of conditions, there is no reason to believe they would always be so, given changes in trainees, learning goals, and domain material over time. Thus, even if a tutor’s methods were once optimal, a tutor should be able to evolve or adapt to a different set of learners or to modified domain knowledge, both of which might require different approaches to tutoring.

In this section we present three methods for improving tutoring strategies based on trainee actions: the first, crude but functional, is carried out by a human, based on logged data, the second, not recommended, is carried out by the tutor based on data gathered from automated controlled experiments, the third is an automated version of the Learning Rate versus Learning Strategies reported in the evaluation of LEAP.

Method 1. The simplest procedure is to have the tutor log data (the details of learning rates and learning methods), and have an instructional designer analyze this data off-line, determine improved study methods, if any, and reconfigure the tutor to teach using the improved methods. A more ambitious research goal is to have the tutor make the observations itself and reconfigure itself.
Method 2. One way to improve a tutor’s tutoring skills (not however, the one we are interested in) is to have the tutor be in control, experiment with different tutoring methods, and incorporate the results into its tutoring strategies. A connectionist approach to this task is suggested by Lesgold (1992), and O’Shea (1982) reports some success at having the tutor experiment with and modify its tutoring rules.

Method 3. In any case, full tutor control of learning activities (as needed for experimentation) is not acceptable to most adult trainees, thus we need to consider methods that permit a substantial amount of trainee control yet still allow the tutor to learn improved tutoring methods. In fact, the observations and improved tutoring strategies discussed in the Evaluation chapter, for example, depend on trainees taking the initiative. Here we propose expanding on this method.

The tutor should observe and log trainees’ learning rates and degree of self- versus tutor-direction as categorized in the Evaluation chapter. The tutor should also observe and log trainees’ learning methods, or patterns of command use. There are two sources of improvement or modification of the tutor’s tutoring strategies. The first is from the self-directed trainees who outperformed the tutor-directed trainees. The tutor should incorporate these more-productive study patterns into its own tutoring methods (with human intervention as necessary). That is, its tutoring methods should be revised to produce these learning patterns in trainees. The second is from the self-directed trainees who performed as well as the tutor-directed trainees, but used their own (different) learning methods. The tutor should note these alternate learning methods in order to be able to recommend them to those trainees for whom its nominal tutoring approach yields disappointing results.
7.3.2.2.5 Tutor Modifies Trainees' Learning Strategies

Scardamalia and Bereiter (1994) make a valid criticism of the instructional approach usually taken in intelligent tutoring systems for its locus of intentionality:

With intelligent-tutoring systems, the intentionality resides in the system's own diagnostic and decision process. The contrasting view, which we have embodied in CSILE, is that an important part of education is for trainees themselves to learn to carry out those diagnostic and decision processes. (p. 278)

They are correct. It is important for trainees to learn to observe their own learning processes and make their own learning decisions. The approach outlined in this section shows how a tutoring system might support this sort of learning.

As mentioned in the previous section, full tutor control of learning activities is not acceptable to most adult trainees. Full tutor control is also undesirable from an instructional design perspective, since the benefits of self-directed learning include increased motivation, more active involvement in the learning process, more investment in learning outcome, etc. Thus a tutor needs to permit a substantial amount of trainee control yet apply the power of intelligent tutoring to optimizing the learning process. One means of accomplishing this is to have the tutor use its tutoring skills, not to make tutoring decisions for the trainee, but rather to observe and modify the trainee's learning strategies. This approach has the benefit that trainees finish training not only having mastered the domain knowledge but also with an improved capability for learning to learn.

A working assumption in this section is that the trainee can choose to be tutor-directed, self-directed, or any combination of the two. In the tutor-directed condition, the tutor makes the instructional decisions, in the trainee-directed
condition, the trainee does so. Instructional decisions include selecting: topics, activities, increment of new material in exercises, amount and frequency of review, expository or exploratory sequencing of materials, etc.

In all cases, the tutor observes the trainee’s rate of learning and degree of self-directedness, classifies the trainee into one of the six categories described in Figure 7.1 and Table 7.5, and responds accordingly. For example, if the trainee is self-directed and learning at a slow rate, i.e., floundering, the tutor will intervene and encourage more tutor-directed study. In contrast, if the trainee is tutor-directed and learning at a slow rate, the tutor will change its tutoring method to try and find an alternative methodology more suitable to the trainee’s learning style. Next, if the trainee is tutor-directed and learning at an acceptable rate, the tutor will encourage the tutor to make more of his or her own study decisions, i.e., to become more self-directed. Finally, if the trainee is self-directed and learning at an acceptable rate, the tutor will not intervene, but will, as noted above, note the trainee’s methods for possible incorporation into its own instructional techniques.
Figure 7.1

Learning rate versus self-directedness
Table 7.5
Tutor Action, Based on Trainee Direction and Learning Rate

<table>
<thead>
<tr>
<th>Learning Rate is:</th>
<th>Tutor-directed</th>
<th>Self-directed</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>Encourage more self-directed study</td>
<td>Note this superior study method a b</td>
</tr>
<tr>
<td>Average</td>
<td>Encourage more self-directed study b</td>
<td>Note this alternative study method a b c</td>
</tr>
<tr>
<td>Low</td>
<td>Try an alternative study method</td>
<td>Encourage more tutor-directed study b</td>
</tr>
</tbody>
</table>

Note, a For later inclusion into tutor’s methods (with human assistance). Some study methods may be more suitable than others for certain learning styles.
b Trainees with these characteristics were observed during the LEAP field trial.
c Some trainees might apply only the study methods learned from tutor.

A new role for the ITS tutoring module, then, is to cease making tutoring decisions for the learner, and instead to guide learners in developing good learning practices (to tutor tutees in tutoring).

To summarize, this section on further ITS research has proposed five methods to improve the tutoring process; four methods give the trainee, the instructor, the instructional designer, and the tutor, respectively, the information and the control to modify the tutoring process; the fifth method guides trainees in developing good learning practices.
7.3.2.3 Beyond Skills-focused Tutoring

LEAP is an example of a skill-focused ITS, but ITSs have potential for broad educational application. Gardner (1991) believes that "educational institutions ... ought to seek to inculcate in their students the highest degree of understanding." By education for understanding he means "... simply a sufficient grasp of concepts, principles, or skills so that one can bring them to bear on new problems and situations ..." He lays out "overwhelming evidence documenting the limited nature of student understanding across the disciplines" and recommends specific interventions to "... dissolve the powerful misconceptions and stereotypes entertained by the unschooled mind" (p. 19). The interventions Gardner suggests are institutional, instructional, and technological; this discussion is limited to the last of these. Gardner provides several examples of ITS or learning environments in the domain of physics and math. However, he provides no examples of technological intervention in the arts and humanities domains. Also, Gardner provides examples of how two topics or concepts, namely evolution and democracy, might be taught from several perspectives or 'entry points' that correspond roughly to his seven intelligences. In particular he lists the narration, logical-quantitative, foundational, esthetic, and experiential entry points. To date, ITSs and related technologies have dealt only with the logical-quantitative perspective, which is the easiest to represent in machine-interpretable form, a prerequisite for constructing ITSs. The narration and foundational approaches require natural language understanding, which is currently highly constrained. The esthetic approach requires representing esthetic considerations, a relatively untouched area of AI research, and tutoring the experiential approach requires either creating 'experiences' on-screen or somehow sensing or monitoring a student's actual experiences. The former approach replaces actual objects with icons, the latter requires apparatus that is non-standard computer equipment. Either
approach may be acceptable, depending on the circumstances; for example one early videodisk based course taught welding skills by having the user manipulate the light-pen as a welding torch; another taught CPR by having trainees practice on a dummy wired with sensors that provided input to the computer. To summarize, current AI research results limit ITS instruction to the logical-quantitative and experiential approaches; in the future it may be possible to build tutors capable of the narration, foundational, and esthetic approaches.

7.4 Summary

This chapter has presented results and contributions from the LEAP ITS research project. The project attained its design, implementation, and evaluation objectives, surpassed conventional computer-based instructional alternatives, and the tutor was embraced by its stakeholders.

This research makes four contributions to the intelligent tutoring field: Focused Practice is a new instructional method, LEAP is a benchmark example of a successfully fielded practical tutor, the LEAP project is an example of a design approach taking into account theories of situated cognition, socio-technical systems, and participatory design, and the LEAP project has shown the advantages of a new paradigm for ITS research: shifting from a computational to an instructional focus.

The final section of the chapter suggested a number of improvements in LEAP and proposed five methods to improve the tutoring process, including one which ceases making tutoring decisions for the learner, and instead guides learners in developing good learning practices.
A.1 Introduction

This section is a walkthrough of the LEAP ITS as it was configured at the time of its 1993 field trials. The walkthrough describes LEAP from the user's perspective: for most of the walkthrough the user is a trainee; for the authoring portion of the walkthrough, the user is an author.

This walkthrough will cover the top level screens, the three tutoring methods trainees can use with LEAP, and one part of authoring: revising LEAP's decision-making strategies. While interacting with LEAP, trainees are not only viewing these screens, which contain the same database (called SONAR) they use on the job, they use the same keyboard and wear the same telephone headsets they use on the job, and through the headset's earphones and microphone they converse with simulated customers.

A.2 Top Level

Figure A.1 shows the LEAP log in screen. Individual trainees enter their first and last names, which LEAP uses to establish and maintain their student model, they then click on OK to go on.
Figure A.1.

LEAP Log in screen

Figure A.2 shows LEAP's top level screen. It has two windows, the Commands window and the Topics window. The Commands window presents trainees with the options they have in this part of LEAP. The Topics window presents the list of topics the trainee can study. In the Commands window clicking the button:

Help: pops up a window describing how to use the remaining commands on the screen.
View Conversations: replaces the Topics window and its list of topics with the Conversations window and its list of conversations (or exercises)\(^1\).

Recommend: pops up a LEAP-determined recommendation for what to study next.

Exit LEAP: will end the session, saving the student model.

The remaining three buttons in the Commands window each initiate a particular method of instruction for whichever Topic is selected. Clicking:

- **Study the Guide**: takes the trainee to declarative presentations regarding the selected topic.
- **Rehearse Conversation**: enables the trainee to practice role-plays with simulated customers and a simulated database. Rehearsing a conversation is very close to performing an actual job task.
- **Examine Contact Flow**: lets the trainee see all possible situations and either see or try the appropriate actions for each situation within a Topic.

The Topic window has a list of the topics and subtopics of instruction. Clicking on a topic opens or closes its list of subtopics. Clicking on a topic or subtopic also selects it for instruction.

LEAP posts the trainee's updated proficiency rating for each topic and subtopic whenever the trainee returns to the top level screen (i.e., each time the trainee completes an exercise). The proficiencies range from 'Untried' through 'Needs practice\(^2\),' 'Almost,' and 'Good,' to 'Excellent.' These five labels are based on two numbers; one measuring the percentage of material in the topic the trainee has tried, and the other the trainee's average score on that material. From the proficiency ratings trainees always know their current

---

\(^1\) LEAP's 'conversations' are analogous to exercises in conventional instruction. When trainees practice conversations with each other, they call them role-plays. On the job, conversations with customers are called contacts.

\(^2\) Occasionally trainees would take offense at the phrase 'Needs practice.' The phrase 'Practicing...' might be better since it indicates work in progress instead of judgment of a finished performance.
performance level. Field trial results confirm that trainees believe LEAP rates their proficiencies accurately.

![LEAP Top Level screen - Topics presented](image)

Figure A.2.

LEAP Top Level screen - Topics presented

Figure A.3 shows the Conversations window. Trainees select this window by clicking on the View Conversations button in the Commands window. When the Conversations window appears the View Conversations button is replaced by the View Topics button; otherwise the Commands window remains the same.
The Conversations window displays a list of conversations together with two student model data: the number of times the trainee has seen or practiced each conversation, and its current 'Challenge,' or difficulty, rating. Each conversation has a descriptive title to aid the trainee in selecting specific conversations for practice.

Trainees click on a specific conversation, or exercise, to select it and then click the Rehearse Conversation button to begin practicing it.

![LEAP Tutoring for Voice Messaging](image)

**Figure A.3.**

LEAP Top Level screen - Conversations presented
Figure A.4 shows the top level Help screen. At this level, Help is a one-screen affair describing the trainee's options. First it describes the two-step process for choosing a practice exercise, then it describes the other instruction-type buttons, next it describes the alternative of having LEAP recommend what to study and finally, how to switch back to the Topics window.

To receive instruction:
First, click a conversation from the "Conversations" list.
Then, click one of the instruction activities:
"Study the Guide" will allow you to browse through some course material.
"Rehearse Conversation" will allow you to observe or practice contacts.
"Examine Contact Flow" will allow you to browse possible contact flows.
For a suggestion at any time click the "Recommend" button.
To select instruction based on topic click the "View Topics" button.

Figure A.4.
LEAP Top Level screen - Help presented
Figure A.5 shows how LEAP presents a study recommendation. When trainees want to know what they should study next, they click Recommend and LEAP presents its recommended learning method and recommended topic. LEAP uses its tutoring knowledge to select the topic and method based on the current state of the student model. The trainee can Accept the recommendation and begin practicing or Reject it and make his or her own choices. The next sections describe the three learning methods in more detail.
A.3 Study the Guide

When trainees begin to study a fresh topic, LEAP recommends they first Study the Guide. Figure A.6 presents the opening page of the Guide. For ease of use, the Guide is based on a familiar metaphor, the book. Trainees move through the Guide section by section by clicking on the tabs sticking out from the pages. Trainees move through a section page by page by clicking on the turned up corners of the leaves.

Figure A.6.

LEAP “Study the Guide” entry screen
In Figure A.7 the Guide is open to a typical section. Trainees can either read the text themselves or click on it and hear it read to them. By clicking on the camera icon they can see a short video on a related topic. Finally, clicking on the yellow note marked Exit returns the trainee to the top level of LEAP.

In this chapter, you'll learn about our Voice Messaging Service and how it works. By understanding your customer's needs, and then telling them about benefits of Voice Messaging that meet those needs, you'll be cultivating loyal customers for US WEST - and helping to grow our business.

Voice Messaging answers the phone when the customer can't.

Voice Messaging Service (VMS) is a service comprised of the Basic Elements (BSE): Call Forward, Busy Line, Don't Answer, and the Message Waiting Indication. Calls to the customer's line are transferred to a multi-line hunt group by adding a call forwarding number to the order. The information is transmitted over a private line between the central office switch and US WEST Voice Messaging equipment. Messages are stored in a voice mailbox.

Figure A.7.

LEAP study “page” screen

A.4 Rehearse Conversation

Once trainees have studied LEAP's presentation of declarative information, LEAP will recommend they begin practicing the skills they must ac-
quire. LEAP will recommend a topic, select the most suitable conversational exercise for that topic and recommend that the trainee begin by observing LEAP’s expert service representative, or rep, perform the conversation.

Figure A.8 shows the Rehearse Conversation screen. The upper window contains a set of commands for using LEAP: Help, Hints, Show History, Related Info, and Exit. Clicking on:

Help: brings up help about how to use LEAP in the Rehearse Conversation mode.

Hints: gives the trainee situation-specific information about the content, in this case, information about Voice Messaging Service (VMS).

Show History: pops open the History window which contains the script of the conversation up to the current point.

Related Info: brings up the Guide, opened to the location pertaining to the current point in the conversation.

Exit: returns the trainee to LEAP’s top level.

---

3 Exit, both here and in the Guide, would more appropriately be called "Return to Top Level."
The two lower windows handle the simulated conversation. The left window gives the trainee control over the spoken part of the exercise, the right window simulates SONAR, the database application. Both windows have a set of three conversation study method buttons at their top: Observe, Focused Practice, and Full Practice.

When the Observe button is depressed, trainees observe LEAP's expert service rep perform the conversation. The expert will talk with the customer, determine the customer's needs, and provide a solution, all the while retriev-
ing pertinent information from the customer's account and entering new information into the account.

When the Full Practice button is depressed, the expert disappears and the trainee must do the entire exercise him- or herself. Finally, when the Focused Practice button is selected, LEAP first determines which portions of the conversation the trainee would most benefit from by practicing (thus focusing the practice). LEAP then presents the whole exercise step by step, having the trainee practice in the areas of focus and having LEAP's expert rep do the other steps of the exercise.

The actions the trainee performs in each window are quite different from each other so the actions for each of the windows will be discussed separately.

In the left window the trainee manages the speaking parts of the exercise. Exercises are made up of situation-action pairs: the trainee must learn to take the appropriate action for each situation.

In Observe mode, the trainee first hears a situation, i.e., a customer statement, in his or her headset, and sees a description of the appropriate action in the Conversation window. The action description has a button; by clicking the button, the trainee will hear the expert rep's spoken response. The trainee clicks through the situation-action pairs until the conversation is complete.

In Full Practice mode, and in Focused Practice when the trainee must practice the response, the procedure for completing a situation-action pair is a four-step process (the three visible steps are shown in Figure A.9).

---

4 Field trial results indicate that the complexity of this sequence causes some users difficulty. Focusing attention on this step during the initial LEAP training session does prevent the problem, but a simpler design is preferable.
• First, the situation is presented, that is, the trainee hears the customer's statement.

• Second, the trainee records a response: LEAP presents the Start Recording Response button in the Conversation window, Figure A.9, first screen. The trainee presses the button and begins recording his or her response. As the button was pressed its label changed to Stop Recording Response, as in Figure A.9, second screen. The trainee presses Stop Recording Response when he or she is through recording. LEAP saves the recorded response for later review by the trainee and for possible review by the instructor.

• Third, when the trainee presses the Stop Recording Response button, LEAP presents a list of possible responses, as in Figure A.9, third screen, to the trainee, who selects the one closest to his or her actual response. Trainees are required to indicate their responses this way because LEAP cannot understand spoken language. If the trainee selects an incorrect response he or she is given feedback to that effect and must select another response. LEAP updates the student model based on the first response the trainee selects from the list.

• Fourth, when the trainee selects the correct response, LEAP plays the expert's response and goes on to the next situation-action pair.

The SONAR simulation in the right hand window of Figure A.8 replicates the relevant features of the SONAR database application. In the SONAR simulation window the trainee encounters the same sequence of screens and moves among them in the same way as with the real SONAR (the simulation is based on captured SONAR screens). Within a screen, the trainee moves from field to field and inputs data as with the real SONAR. When the trainee performs as expected, SONAR responds as expected; when the trainee makes an error, LEAP responds with an error message detailing the expected versus actual response (e.g., “Expecting '785' in the Exchange field, you typed 'Jo/se/ph' in the Date field.”).

The description above contains the simplifying assumption that conversational situations have conversational responses and SONAR situations
have SONAR actions; in fact, both types of situation have both types of response. For example, the trainee might see that the SONAR Due Date field needs to be filled in, and discuss the due date with the customer. When the customer states a due date the trainee would type it into the Due Date field.

Figure A.9.
LEAP Rehearse Conversation response recording process
In Figure A.10, the Show History button has been pressed and the Conversation History window is open. The window shows the lines the rep and the contact (the customer) have spoken. Any lines recorded by the trainee have the word “[Recorded]” to their left. By default, the History window opens displaying only the spoken lines (i.e., the Conversation Only button is depressed). The trainee can click Show All to see both the conversation and the SONAR I/O. The trainee can click on any line in the conversation to select it and then click Hear Original to hear the expert rep (or customer) speak the line. If the trainee has recorded a line, s/he can also click Hear Myself to hear the recording of him or herself speaking the line. Comparing and contrasting one’s own responses with the expert’s is a powerful learning tool. Close History hides the History window.

In real life, the rep has only the SONAR window on his/her terminal. In LEAP, trainees need a means of controlling the conversation flow as well, thus the Conversation window. Because conversing is the dominant task, trainees come to focus their attention on the conversation window. When LEAP expects trainees to change focus to the SONAR window the message “Do something in SONAR” appears in the Conversation window, as in Figure A.10.

---

1 One improvement on the current approach would be to employ a structured method for comparing oneself with the expert, i.e., a self-critique. One can envision reviewing each line of the conversation, comparing oneself with the expert, with the aid of a check-list of salient features for that particular line. One part of the review would be the opportunity to re-record - just for practice - the line and replay it until one was satisfied.

Another improvement: change the button names to their best current meaning. Some are now obsolete, e.g., Rehearse Conversation should be Practice, and Examine Contact Flow should be: Explore Contact Possibilities.
A.4.1 Hint

Hints give the trainee clues about what to do next with respect to the content area (in this case, Voice Messaging Sales and Service to existing customers). Hints in LEAP are triple-layered; each layer provides an increasingly more specific, detailed hint. Hints for the Conversation and SONAR windows, though similar, are different enough to merit separate descriptions.

Figure A.11 shows the first level SONAR hint. The hint tells the trainee the current topic or context, the screen name, and the label of the field
where action is expected. If the trainee can proceed, s/he clicks OK and acts. If not, s/he clicks Show Answer and the next level Hint appears, as in Figure A.12. It additionally describes the data the trainee should enter in the field; again, if the trainee can proceed from here, s/he clicks on OK and acts. Otherwise, s/he clicks Demonstrate and LEAP goes on to level 3, an actual demonstration of the step, doing it for the trainee.

Figure A.11.

SONAR-related level 1 hints
Figure A.12.

SONAR-related level 2 hints

Figure A.13 shows the first level conversation hint. Again, LEAP presents the current context, and the situation to which the trainee should respond. The trainee can click OK if s/he knows what to do or click Show Answer if necessary to get the next level of Hint.
Figure A.13.

Conversation-related level 1 hint

Figure A.14 shows the level 2 conversation hint. This time the hint contains a description of what the trainee should say; the trainee can click OK or continue to the level 3 hint: Hear Expert, whereupon the Hint window closes and LEAP’s expert rep models the appropriate response. The trainee can then record the response him/herself.
The current response is: Explain: With Voice Messaging your Call Waiting will still work fine.

Figure A.14.
Conversation-related level 2 hint

A.4.2 Help

Figure A.15 shows the first Help screen that appears in Rehearse Conversation. The Help screen orients the trainee, provides specific instruction for the current study method, tells the trainee how to change methods, and points the trainee to more specific help via the buttons at the bottom of the Help screen: Windows, Commands, Instruction Styles, SONAR, and History. Figures A.16 through A.20 display the screens each of these buttons, respectively, brings up. Each Help screen displays these buttons for direct access to the other Help screens.
Figure A.15.
Rehearse Conversation top level Help screen

Figure A.16.
Rehearse Conversation “Windows” Help window
Help: Brings you here; gives info about how to use LEAP.
Hints: Hints about what to do next.
Show History: Displays a history of the contact up to the current point.
Related Info: Jumps into appropriate section of Guide to Voice Messaging.

See "Instruction Styles" to change the way instruction works.

Figure A.17.
Rehearse Conversation "Commands" Help window

Conversation and SONAR Instruction Styles

With the buttons "Observe," "Focused Practice" and "Full Practice," set the level at which you practice your skills.

You can change the level for both Conversation and SONAR activities.

Observe: Shows you how an expert rep does the task.
Focused Practice: You practice the task. Often, LEAP does some steps for you.
Full Practice: You practice the task throughout the contact.

Figure A.18.
Rehearse Conversation "Instruction Styles" Help window
SONAR Simulation Help

To use SONAR, the cursor MUST be in the SONAR window.

Move cursor to desired field and enter appropriate data.

Move cursor using:
- Arrow keys: Move: Up, Down, Left, Right
- Tab, Shift+Tab Goes to next/previous input field.
- Home Goes to command line at top left of screen
- Mouse click Places cursor at clicked point
- Enter (on keyboard) Goes to input field on next line

Go to next screen using:
- Enter (on keypad): Goes to next screen

Function keys:
- F1 Next Data
- F2 Previous Data
- F7 Issue Order
- F11 Next Info (csr)
- F12 Previous Info (csr)

Create Conversation “SONAR Simulation” Help window
When the trainee reaches the end of a conversation exercise, LEAP changes the Conversation window as shown in Figure A.21, and brings up the History window. (In the History window, as described earlier, the trainee can review his/her performance, comparing it with the expert rep’s to see where it’s good enough, and where it needs more practice.) The Conversation window now informs the trainee s/he has completed the exercise and describes his/her options: Summary, Repeat Conversation, and Exit. Clicking:

Summary: brings up a short video in which LEAP’s expert rep recaps the salient points of the just-completed conversation\(^2\).

Repeat Conversation: brings the trainee to the start of the same conversation so s/he can redo it immediately. (In contrast to LEAP’s built-in strategy which is to cycle through all the conversations that touch on a topic before repeating any.)

\(^2\) The video summary might also be presented before the exercise, as an advanced organizer.
Exit: functions exactly the same as the Exit in the Commands window, it returns the trainee to LEAP’s top-level. The Exit button is replicated here solely for the trainee’s convenience.

Figure A.21.
Rehearse Conversation feedback screen

A.5 Examine Contact Flow

We turn now to the last of LEAP’s three study methods: Examine Contact Flow. From LEAP’s top-level screen, clicking on a Topic and on Examine Contact Flow brings up the display shown in Figure A.22. In contrast to Rehearse Conversation, which takes linear paths through the space of pos-
sible conversations, Examine Contact Flow allows the trainee to traverse all paths in the conversation space. Whenever the conversation could branch, all branches are displayed, and the trainee may select any one of them, traverse it as far as s/he likes, back up, select another branch, traverse that, etc. In this way trainees can study a selected topic in detail, learning how to respond to all the situations that could occur at every point in the topic.

Figure A.22, for example, displays many different service options that customers might have on their telephone lines. Trainees must learn how each of these service options interacts with or conflicts with Voice Messaging, and what s/he should do or say about each.

As in Rehearse Conversation, the Examine Contact Flow and SONAR windows have three buttons: Observe, Focused Practice, and Full Practice, enabling the trainee to pick the desired study method.

In Examine Contact Flow the Commands window has two additional buttons, one to toggle the Audio on/off, and one to Back Up through the conversation space.
**SONAR CSR CALL WAITING, CONTROLLABLE**

**SONAR CSR CALL WAITING, ENHANCED**

**SONAR CSR CUSTOM RINGING 1AESS**

**SONAR CSR SPEEDCALL**

**SONAR CSR CUSTOM RINGING, C.O. switch Is DMS or SESS**

**SONAR CSR VACATION SERVICE**

**SONAR CSR CALL WAITING, CONTROLLABLE**

**SONAR CSR CALL WAITING, ENHANCED**

**SONAR CSR CUSTOM RINGING 1AESS**

**SONAR CSR SPEEDCALL**

**SONAR CSR CUSTOM RINGING, C.O. switch Is DMS or SESS**

---

**Click on a possible situation:**

1. ADDITIONAL LINE
2. ASSISTANCE MENU
3. BANK PLAN
4. BUSINESS TO RESIDENCE
5. CHANGE ADDRESS
6. CHANGE BILL MAILING
7. CHANGE CARRIER DATA
8. CHANGE LISTED NAME
9. CHANGE RESPONSIBILITY
10. CHANGE SERVICE
11. CHANGE TEL. #
12. CONCESSION
13. CSR ONLY
14. DIRECTORY DELIVERY INFO
15. DISCONNECT
16. HELD NEGOTIATION - CHANGE

---

**Figure A.22.**

Examine Contact Flow screen #1

---

389
In Figure A.23, the trainee is Observing; she has picked one of the possible situations (the contact’s account information reveals she has Custom Ringing, etc.), and LEAP’s expert has displayed the appropriate response for the trainee to read; the trainee can also click on the response to hear the expert say it.

Figure A.23.
Examine Contact Flow: Observe
In Figure A.24, the trainee is in the Full Practice method rather than Observe, the Situation is the same as before, but this time LEAP presents a list of responses from which the trainee must choose the appropriate one.

Figure A.24.

Examine Contact Flow: Full Practice

Figure A.25 presents the Help screen for Examine Contact Flow. The first line of Help tells the trainee his or her current context. The second line
tells him or her what to do in that context. The third line tells him or her how to change the context, and the last line, more about Help.

Figure A.25.
Examine Contact Flow: Help screen

Figure A.26 shows the display when the trainee reaches the end of a Topic in Examine Contact Flow. The History window opens so that trainees can review and replay the whole sequence, and some buttons appear in the Conversation window. Clicking:

Back Up: backs the trainee through the conversation, one SA pair at a time. All branches reappear.
Repeat Flow: takes the trainee to the beginning of the topic.
Exit: returns the trainee to the top level screen.

The History window works differently in Examine Contact Flow than it did in Rehearse Conversation; since no specific conversation has been rehearsed, selecting a line and clicking Hear Example once plays a random instance of the line, and clicking Hear Example repeatedly plays all the different recordings of that line, from all the different conversations that contain it. Trainees can see that there are many ways to make the same point or accom-
plish the same task, some of which are context dependent and some which are not.

This is the end of the sections describing what trainees see. The remaining section walks through one part of the authoring process.

**A.6 Revising LEAP’s Decision-making Strategies**

One step of LEAP’s authoring process requires weighting various factors that determine the tutor’s decision-making strategies. LEAP has several screens for setting these instructional decision-making factors. Figure A.27 shows the screen the author uses to adjust LEAP’s topic selection factors. The upper portion of the screen has two windows, the window on the left displays the list of topics and subtopics, the window on the right displays the conversations. Just below the two windows are buttons for moving among the four factor weighting screens. Clicking:

- **Topic Choice**: brings up the Topic Choice screen (Figure A.27).
- **Conversation Choice**: brings up the Conversation Choice screen (Figure A.28).
- **Student Modeling**: brings up the Student Modeling screen (Figure A.29).
- **Question Probabilities**: brings up the Question Probabilities screen (Figure A.30).
LEAP study parameters - “Topic Choice” selected

When LEAP recommends the trainee study a particular topic, it bases its recommendation on several weighted factors; the weight of each factor in the recommendation is adjustable by the author; to adjust the weight the author simply moves the slider beside the factor’s name; the factors are displayed on the lower portion of the screen in Figure A.27. Thus, when recommending the next topic for study, LEAP will, given the weightings in Figure A.27, weight the trainee’s proficiency most, followed by a consideration of the topic the trainee last studied, followed by a consideration of the topic’s sequence in the list.
LEAP’s ‘spiral curriculum’ is implemented with the Use of Proficiency Category... The Use of Proficiency Category... allows authors to set LEAP’s preference for one topic over another based on the trainee’s current proficiency ratings for the topics.

At the bottom of the Topic Choice screen are four buttons. Clicking:

Reset Default Parameters: changes the parameters (factors) back to the pre-programmed weights.

Save Parameters: sets the values of the parameters (factors) to those indicated on the sliders, and displays the currently recommended topic.

Practice Conversation: jumps to a conversational exercise for the recommended topic.

Exit: returns the author to LEAP’s top level screen. (When LEAP is opened in Author mode, the top level screen has a button to call up the factor weighting screens.)

Figure A.28 displays the Conversation Choice authoring screen. This screen is identical to the Topic Choice screen except the factors affecting topic choice are replaced by those affecting conversation choice. Some of the factors are characteristic of the conversations themselves, others are characteristic of the trainee’s current model.
### LEAP study parameters - “Conversation Choice” selected

Figure A.29 shows the Student model factor weighting screen. In the student model, each Situation-Action pair has a value attached to it that indicates the trainee’s knowledge of that pair. The factors affecting the value have sliders beside them so that the author can weight them according to his or her understanding of the domain knowledge, student characteristics and instructional goals.
The screens in Figures 30 and 31 are identical except that the screen in Figure A.30 pertains to the current topic and the screen in Figure A.31 pertains to all the other topics in a conversation. In both cases, the line in the upper half of the screen indicates the likelihood that the trainee will be asked to perform the action for an SA pair. In general, the higher the challenge (the less the trainee has performed the action correctly) the greater the likelihood she will be asked to practice it. The author manipulates the line by moving the sliders in the lower half of the screen.
The line in Figure A.31 contrasts with that in Figure A.30 in the following ways: the likelihood of being asked to respond to a situation not in the topic is: lower in general, and, if the SA pair is unknown to the trainee (i.e., has a challenge of "a lot," and is not in the topic, she will not be asked to perform it at all.
LEAP study parameters - “Out of Topic Question Probabilities” selected

This completes the LEAP Walkthrough.
APPENDIX B
LOGGED DATA: SUMMARY AND SAMPLE

This appendix presents a table summarizing logged student data (Table B.1), and a sample of a trainee’s logged data (Table B.2). All trainees’ actions were logged as they used LEAP. The summary table was prepared from the individual logs. The individual log is a detailed, sequential record of an hour of one trainee’s actions as recorded by LEAP’s logging facility.
Table B.1.  
Summary of Logged Student Data

| a  | b  | c  | d  | e  | f  | g  | h  | i  | j  | k  | l  | m  | n  | o  | p  | q  | r  | s  | t  | u  | v  | w  | x  | y  | z  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1  | 1  | 4  | 3  | 11 | 20 | 10 | 209 | 2  | 3  | 7  | 65 | 41 | 2  | 1  | 1  | 21 | 2  | 1  | 33 | 2  | 11 | 24 | 60 | 16 |
| 1  | 2  | 7  | 3  | 18 | 11 | 13 | 142 | 2  | 11 | 22 | 47 | 29 | 9  | 2  | 3  | 5  | 4  | 0  | 40 | 3  | 7  | 1  | 0  | 3  |
| 1  | 3  | 2  | 1  | 3  | 28 | 7  | 201 | 4  | 2  | 7  | 63 | 47 | 4  | 1  | 2  | 2  | 29 | 2  | 0  | 39 | 3  | 8  | 7  | 28 | 6  |
| 1  | 4  | 4  | 3  | 11 | 20 | 8  | 162 | 4  | 3  | 14 | 57 | 22 | 7  | 1  | 3  | 6  | 0  | 14 | 5  | 25 | 8  | 21 | 43 | 26 | 2  |
| 1  | 5  | 7  | 1  | 8  | 29 | 7  | 202 | 5  | 2  | 9  | 55 | 43 | 16 | 4  | 3  | 4  | 5  | 15 | 5  | 149 | 4  | 13 | 3 | 2  | 6  |
| 1  | 6  | 10 | 1  | 13 | 14 | 13 | 183 | 3  | 2  | 7  | 63 | 34 | 10 | 3  | 3  | 5  | 10 | 6  | 9  | 38 | 1  | 15 | 8 | 25 | 9 |
| 2  | 1  | 2  | 12 | 25 | 21 | 11 | 227 | 7  | 7  | 51 | 51 | 44 | 17 | 1  | 4  | 8  | 2  | 10 | 4  | 29 | 2  | 12 | 29 | 0 | 13 |
| 2  | 2  | 2  | 12 | 24 | 24 | 8  | 186 | 24 | 3  | 67 | 63 | 40 | 10 | 1  | 2  | 3  | 22 | 8  | 4  | 14 | 10 | 26 | 6 | 16 | 3 |
| 2  | 3  | 1  | 7  | 7  | 27 | 9  | 247 | 3  | 7  | 21 | 71 | 44 | 4  | 1  | 2  | 4  | 5  | 11 | 2  | 14 | 12 | 20 | 29 | 57 | 2 |
| 2  | 4  | 3  | 0  | 1  | 16 | 11 | 169 | 11 | 3  | 38 | 55 | 41 | 16 | 1  | 4  | 4  | 12 | 7  | 0  | 55 | 0 | 16 | 10 | 3 | 0 |
| 2  | 5  | 2  | 7  | 14 | 24 | 10 | 231 | 5  | 4  | 21 | 71 | 38 | 14 | 1  | 3  | 4  | 9  | 3  | 2  | 27 | 9 | 20 | 13 | 10 | 7 |
| 2  | 6  | 3  | 2  | 7  | 31 | 8  | 259 | 1  | 10 | 10 | 71 | 39 | 14 | 0  | 5  | 6  | 1  | 4  | 3  | 7  | 16 | 28 | 63 | 50 | 6 |
| 3  | 1  | 11 | 1  | 13 | 25 | 8  | 208 | 12 | 2  | 22 | 57 | 74 | 8  | 2  | 8  | 10 | 8  | 23 | 4  | 24 | 7 | 39 | 4 | 46 | 5 |
| 3  | 2  | 2  | 3  | 7  | 18 | 14 | 246 | 1  | 8  | 8  | 58 | 52 | 13 | 3  | 1  | 2  | 2  | 3  | 1  | 33 | 4 | 21 | 6 | 4 | 1 |
| 3  | 3  | 4  | 2  | 7  | 19 | 10 | 195 | 25 | 2  | 47 | 61 | 71 | 4  | 2  | 3  | 4  | 2  | 1  | 2 | 16 | 19 | 60 | 35 | 6 | 3 |
| 3  | 4  | 3  | 2  | 6  | 20 | 11 | 229 | 4  | 3  | 12 | 27 | 61 | 2  | 1  | 2  | 4  | 0  | 17 | 1  | 25 | 7 | 30 | 4 | 46 | 1 |
| 3  | 5  | 2  | 11 | 22 | 10 | 18 | 180 | 17 | 4  | 70 | 39 | 47 | 2  | 5  | 1  | 2  | 21 | 4  | 3  | 15 | 1 | 6 | 7 | 4 | 5 |
| 4  | 1  | 3 | 2  | 7  | 20 | 12 | 260 | 0  | 0  | 0  | 59 | 59 | 18 | 2  | 4  | 4  | 3  | 1  | 9  | 44 | 3  | 5  | 0 | 6 | 5 |
| 4  | 2  | 5  | 5  | 26 | 17 | 14 | 242 | 0  | 0  | 0  | 55 | 69 | 2  | 1  | 9  | 9 | 5 | 12 | 0  | 16 | 2 | 12 | 3 | 49 | 6 |
| 4  | 3  | 2  | 3  | 6  | 25 | 9  | 232 | 3  | 2  | 5 | 51 | 75 | 10 | 3  | 13 | 16 | 10 | 15 | 3 | 11 | 2 | 71 | 3 | 14 | 3 |
| 4  | 4  | 2  | 13 | 27 | 15 | 14 | 214 | 6  | 2  | 13 | 60 | 54 | 2  | 3  | 5  | 5 | 17 | 2 | 2 | 15 | 3 | 8 | 8 | 7 | 5 |
| 4  | 5  | 2  | 4  | 9  | 18 | 15 | 263 | 0  | 0  | 0 | 42 | 68 | 3  | 2 | 3 | 5 | 3 | 16 | 1 | 24 | 1 | 14 | 1 | 3 | 7 |
| 4  | 6  | 5  | 2  | 8  | 19 | 11 | 213 | 1  | 6 | 6 | 56 | 67 | 1  | 1  | 11 | 12 | 2 | 19 | 3 | 22 | 1 | 6 | 11 | 15 | 2 |

Continued, next page
| a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z |
| 5 | 1 | 8 | 2 | 18 | 23 | 7 | 155 | 7 | 4 | 26 | 51 | 52 | 3 | 2 | 5 | 6 | 31 | 5 | 1 | 90 | 2 | 23 | 49 | 47 | 0 |
| 5 | 2 | 3 | 3 | 8 | 19 | 10 | 188 | 1 | 9 | 9 | 48 | 55 | 4 | 0 | 7 | 9 | 10 | 6 | 0 | 4 | 5 | 26 | 30 | 35 | 6 |
| 5 | 3 | 3 | 5 | 14 | 20 | 9 | 174 | 2 | 5 | 11 | 59 | 44 | 3 | 0 | 2 | 4 | 1 | 19 | 3 | 37 | 4 | 35 | 54 | 47 | 6 |
| 5 | 4 | 8 | 5 | 38 | 19 | 9 | 174 | 5 | 4 | 18 | 47 | 34 | 3 | 1 | 4 | 6 | 3 | 18 | 3 | 25 | 4 | 16 | 24 | 72 | 7 |
| 5 | 5 | 4 | 3 | 10 | 19 | 10 | 198 | 1 | 9 | 9 | 61 | 53 | 3 | 0 | 4 | 4 | 14 | 2 | 0 | 53 | 5 | 25 | 20 | 62 | 1 |
| 5 | 6 | 2 | 3 | 6 | 15 | 15 | 231 | 1 | 8 | 8 | 47 | 64 | 14 | 0 | 6 | 6 | 7 | 3 | 1 | 7 | 1 | 17 | 1 | 0 | 2 |

Note. aColumn titles from left to right are:

- Group code
- Individual code
- Guide, number of times entered
- Guide, average time per entry, minutes
- Guide, total time, minutes
- Rehearse, number of times entered
- Rehearse, average time per entry, minutes
- Rehearse, total time, minutes
- Explore, number of times entered
- Explore, average time per entry, minutes
- Explore, total time, minutes
- Student model, score on SA pairs tried
- Student model, percent tried
- Recommend requests
- Top-level help requests
- Change view to Topic list requests
- Change view to Conversation list requests
- Select instruct from topics
- Select instruct from conversations
- Practice-level help requests
- Hint requests
- Repeat conversation requests
- Change instructional style requests
- Hear original requests
- Hear myself requests
- Play conversation summary video requests
Table B.2.
Logged Trainee Data (Excerpted)

Logging in as Trainee 5 5
Date: 09/03/93
Time: 8:30:03
Host machine: binary
Display machine: meteetse

Instruct Top Level Button Pressed: View Conversations
Instruct Top Level Button Pressed: View Topics
Instruct Top Level Button Pressed: Recommend
Instruct Top Level Button Pressed: Study the Guide

Study the Guide at: 8:34:24
Topics: "Change Voice Messaging Ring Cycle".
Running vaps on declarative topic: RING.
Returning from the Study Guide at: 8:38:34

Instruct Top Level Button Pressed: View Conversations
Instruct Top Level Button Pressed: Rehearse Conversation

Starting Rehearse Conversation at: 8:40:31
Instruct Mode = rehearse
Conversation = addvms-custom-ring-1aess
Topic = Order Voice Messaging

Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (CONCLUDE) asked and correct.
Application command correctly input.
Application command demonstrated.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command demonstrated.

Continued, next page
Table B.2., continued

Application command demonstrated.
Application command demonstrated.
Conversation element (CONCLUDE) shown.
Conversation element (CONCLUDE) shown.
Conversation element (TALK) shown.
Conversation element (PLAYTHROUGH) shown.

Completed conversation at: 8:50:39
History Radio Button Pressed: Show All
History Radio Button Pressed: Conversation Only
History Button Pressed: Hear Original
Summary of conversation shown.
Setting Instruct style in PRACTICE-WINDOW to FULL-PRACTICE at: 8:54:59
Setting Instruct style in SIMULATION-WINDOW to FULL-PRACTICE at: 8:55:02

Repeating conversation.

Status of Student Modelling Information at: 8:55:13

<table>
<thead>
<tr>
<th>Grammar</th>
<th>score</th>
<th>%tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMON-CONTACT-PROCEDURE-OPENING</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-ADD-VMS</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER</td>
<td>6</td>
<td>94</td>
</tr>
<tr>
<td>ONA-RESTRICTED</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>CHECK-VMS-AVAILABILITY</td>
<td>6</td>
<td>43</td>
</tr>
<tr>
<td>VMS-NOT-YET-AVAILABLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT-HOW-DOES-VMS-WORK</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>HOW-DOES-VMS-WORK</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CLASS-OF-SERVICE</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-FEATURE-COMPATIBILITY</td>
<td>6</td>
<td>15</td>
</tr>
<tr>
<td>ORDER-VMS-ON-SLF01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Continued, next page
<table>
<thead>
<tr>
<th>Process</th>
<th>Times Seen</th>
<th>Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-ADD-VMS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FULL-DISCLOSURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAILBOX-SETUP</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-CHANGE-RING-CYCLE</td>
<td>6</td>
<td>50</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER*</td>
<td>6</td>
<td>94</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>RING-CYCLE-PROS-AND-CONS</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CHARGE-FOR-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-REMOVE-VMS</td>
<td>6</td>
<td>50</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER*</td>
<td>6</td>
<td>94</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>ATTEMPT-SAVE-VMS</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>CLOSE-CONTACT-ON-CALL-TRANSFER</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ATTEMPT-SAVE-ON-RING-CYCLE-PROBLEM</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2DAY-DUEDATE*</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CHARGE-FOR-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BLANK-OUT-VOICE-MESSAGING</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CHANGE-SALES-CODE-TO-OFFICE-SALES-CODE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-REMOVE-VMS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT-ALL</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>HOW-DOS-VMS-WORK</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Total Course Modelling Evaluation: 6 15
<table>
<thead>
<tr>
<th>Command</th>
<th>Count 1</th>
<th>Count 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>addvms-ccw-cf</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>addvms-custom-ring-1aess</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>addvms-custom-ring-5e-dms</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>addvms-ecw-3wy</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>addvms-not-avail</td>
<td>0</td>
<td>47</td>
</tr>
<tr>
<td>addvms-not-yet-expln-vms</td>
<td>0</td>
<td>44</td>
</tr>
<tr>
<td>addvms-not-yet-wcb</td>
<td>0</td>
<td>38</td>
</tr>
<tr>
<td>addvms-ona</td>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>addvms-speed-cl-intracall</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>addvms-vaca-serv</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>chgrcyc-keep</td>
<td>0</td>
<td>48</td>
</tr>
<tr>
<td>chgrcyc-lengthen</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td>chgrcyc-shorten</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>rmvvms-broken</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td>rmvvms-hard2use</td>
<td>0</td>
<td>22</td>
</tr>
<tr>
<td>rmvvms-save-expln</td>
<td>0</td>
<td>31</td>
</tr>
<tr>
<td>rmvvms-save-hlp-ctr</td>
<td>0</td>
<td>44</td>
</tr>
</tbody>
</table>

Conversation element (TALK) asked and incorrect.
Practice Control Button Pressed: Hints
Showing Talk Answer.

Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Practice Control Button Pressed: Hints
Showing Talk Answer.

Practice Control Button Pressed: Show History
Conversation element (TALK) asked and correct.
Conversation element (CONCLUDE) asked and correct.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Practice Control Button Pressed: Exit
### Table B.2., continued

Exiting early from Rehearse Conversation at: 9:06:50

Status of Student Modelling Information at: 9:06:50

<table>
<thead>
<tr>
<th>Grammar</th>
<th>score</th>
<th>%tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMON-CONTACT-PROCEDURE-OPENING</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-ADD-VMS</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER</td>
<td>75</td>
<td>94</td>
</tr>
<tr>
<td>ONA-RESTRICTED</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CHECK-VMS-AVAILABILITY</td>
<td>52</td>
<td>43</td>
</tr>
<tr>
<td>VMS-NOT-YET-AVAILABLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT-HOW-DOES-VMS-WORK</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CLASS-OF-SERVICE</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-FEATURE-COMPATIBILITY</td>
<td>6</td>
<td>15</td>
</tr>
<tr>
<td>ORDER-VMS-ON-SLF01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-ADD-VMS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FULL-DISCLOSURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAILBOX-SETUP</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-CHANGE-RING-CYCLE</td>
<td>75</td>
<td>50</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER*</td>
<td>75</td>
<td>94</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>RING-CYCLE-PROS-AND-CONS</td>
<td>75</td>
<td>9</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CHARGE-FOR-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-REMOVE-VMS</td>
<td>75</td>
<td>50</td>
</tr>
<tr>
<td>GET-NAME-AND-NUMBER*</td>
<td>75</td>
<td>94</td>
</tr>
</tbody>
</table>

Continued, next page
Table B.2., continued

<table>
<thead>
<tr>
<th>Command</th>
<th>Times Seen</th>
<th>Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>75 100</td>
<td></td>
</tr>
<tr>
<td>ATTEMPT-SAVE-VMS</td>
<td>75 11</td>
<td></td>
</tr>
<tr>
<td>CLOSE-CONTACT-ON-CALL-TRANSFER</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>ATTEMPT-SAVE-ON-RING-CYCLE-PROBLEM</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>2DAY-DUEDATE</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>DETERMINE-CHARGE-FOR-CHANGE-RING-CYCLE</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>VERIFY-ORDER-CHANGE-RING-CYCLE</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>BLANK-OUT-VOICE-MESSAGING</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>1DAY-DUEDATE</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>CHANGE-SALES-CODE-TO-OFFICE-SALES-CODE</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>VERIFY-ORDER-REMOVE-VMS</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>CLOSE-CONTACT-ALL</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>HOW-DOES-VMS-WORK</td>
<td>0 0</td>
<td></td>
</tr>
<tr>
<td>VERIFY-ORDER</td>
<td>0 0</td>
<td></td>
</tr>
</tbody>
</table>

Total Course Modelling Evaluation: 54 15

Conversational Times Seen Challenge

<table>
<thead>
<tr>
<th>Command</th>
<th>Times Seen</th>
<th>Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>addvms-ccw-cf</td>
<td>0 24</td>
<td></td>
</tr>
<tr>
<td>addvms-custom-ring-1aess</td>
<td>1 24</td>
<td></td>
</tr>
<tr>
<td>addvms-custom-ring-5e-dms</td>
<td>0 21</td>
<td></td>
</tr>
<tr>
<td>addvms-ecw-3wy</td>
<td>0 18</td>
<td></td>
</tr>
<tr>
<td>addvms-not-avail</td>
<td>0 62</td>
<td></td>
</tr>
<tr>
<td>addvms-not-yet-expln-vms</td>
<td>0 60</td>
<td></td>
</tr>
<tr>
<td>addvms-not-yet-wcb</td>
<td>0 51</td>
<td></td>
</tr>
<tr>
<td>addvms-ona</td>
<td>0 70</td>
<td></td>
</tr>
<tr>
<td>addvms-speed-cl-intracall</td>
<td>0 22</td>
<td></td>
</tr>
<tr>
<td>addvms-vaca-serv</td>
<td>0 9</td>
<td></td>
</tr>
<tr>
<td>chgrcyc-keep</td>
<td>0 61</td>
<td></td>
</tr>
<tr>
<td>chgrcyc-lengthen</td>
<td>0 28</td>
<td></td>
</tr>
<tr>
<td>chgrcyc-shorten</td>
<td>0 33</td>
<td></td>
</tr>
<tr>
<td>rmvvms-broken</td>
<td>0 30</td>
<td></td>
</tr>
<tr>
<td>rmvvms-hard2use</td>
<td>0 30</td>
<td></td>
</tr>
<tr>
<td>rmvvms-save-expln</td>
<td>0 41</td>
<td></td>
</tr>
<tr>
<td>rmvvms-save-hlp-ctr</td>
<td>0 58</td>
<td></td>
</tr>
</tbody>
</table>

Continued, next page
Instruct Top Level Button Pressed: Recommend
Instruct Top Level Button Pressed: Recommend

Starting Rehearse Conversation at: 9:11:25
Instruct Mode = rehearse
Conversation = addvms-not-yet-expln-vms
Topic = Order Voice Messaging

Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (TALK) shown.
Conversation element (CONCLUDE) asked and correct.
Application command incorrectly input.
Application command correctly input.
Application command demonstrated.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Conversation element (TALK) asked and incorrect.
Practice Control Button Pressed: Hints
Showing Talk Answer.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) shown.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) shown.

Completed conversation at: 9:19:30
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
**Table B.2., continued**

History Button Pressed: Hear Original

Repeating conversation.

---

Status of Student Modelling Information at: 9:20:32

<table>
<thead>
<tr>
<th>Grammar</th>
<th>score</th>
<th>%tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMON-CONTACT-PROCEDURE-OPENING</td>
<td>5</td>
<td>100</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-ADD-VMS</td>
<td>77</td>
<td>100</td>
</tr>
<tr>
<td>GET-NAMESPACE-AND-NUMBER</td>
<td>74</td>
<td>94</td>
</tr>
<tr>
<td>ONA-RESTRICTED</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>78</td>
<td>100</td>
</tr>
<tr>
<td>CHECK-VMS-AVAILABILITY</td>
<td>42</td>
<td>57</td>
</tr>
<tr>
<td>VMS-NOT-YET-AVAILABLE</td>
<td>6</td>
<td>50</td>
</tr>
<tr>
<td>CLOSE-CONTACT-HOW-DOES-VMS-WORK</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>HOW-DOES-VMS-WORK</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>DETERMINE-CLASS-OF-SERVICE</td>
<td>6</td>
<td>29</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>VERIFY-FEATURE-COMPATIBILITY</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>ORDER-VMS-ON-SLF01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-ADD-VMS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FULL-DISCLOSURE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAILBOX-SETUP</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>SERVICE-CHANGE-IS-CHANGE-RING-CYCLE</td>
<td>77</td>
<td>50</td>
</tr>
<tr>
<td>GET-NAMESPACE-AND-NUMBER*</td>
<td>74</td>
<td>94</td>
</tr>
<tr>
<td>SELECT-CHANGE-ORDER</td>
<td>78</td>
<td>100</td>
</tr>
<tr>
<td>RING-CYCLE-PROS-AND-CONS</td>
<td>54</td>
<td>14</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>6</td>
<td>100</td>
</tr>
<tr>
<td>2DAY-DUEDATE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OTHER-DUE-DATE-SCREEN-INFO</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DETERMINE-CHARGE-FOR-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERIFY-ORDER-CHANGE-RING-CYCLE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CLOSE-CONTACT</td>
<td>6</td>
<td>100</td>
</tr>
</tbody>
</table>

Continued, next page
Table B.2., continued

<table>
<thead>
<tr>
<th>CONVERSATION</th>
<th>TIMES SEEN</th>
<th>CHALLENGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>advvms-ccw-cf</td>
<td>0</td>
<td>24</td>
</tr>
<tr>
<td>advvms-custom-ring-laess</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>advvms-custom-ring-5e-dms</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td>advvms-ecw-3wy</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>advvms-not-avail</td>
<td>0</td>
<td>62</td>
</tr>
<tr>
<td>advvms-not-yet-expln-vms</td>
<td>1</td>
<td>60</td>
</tr>
<tr>
<td>advvms-not-yet-wcb</td>
<td>0</td>
<td>51</td>
</tr>
<tr>
<td>advvms-ona</td>
<td>0</td>
<td>70</td>
</tr>
<tr>
<td>advvms-speed-cl-intracall</td>
<td>0</td>
<td>22</td>
</tr>
<tr>
<td>advvms-vaca-serv</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>chgrcyc-keep</td>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>chgrcyc-lengthen</td>
<td>0</td>
<td>28</td>
</tr>
<tr>
<td>chgrcyc-shorten</td>
<td>0</td>
<td>33</td>
</tr>
<tr>
<td>rmvvms-broken</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>rmvvms-hard2use</td>
<td>0</td>
<td>30</td>
</tr>
</tbody>
</table>

Total Course Modelling Evaluation: 41 21
Table B.2., continued

<table>
<thead>
<tr>
<th>Command</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>rmvvms-save-expln</td>
<td>0</td>
</tr>
<tr>
<td>rmvvms-save-hlp-ctr</td>
<td>0</td>
</tr>
</tbody>
</table>

Setting Instruct style in PRACTICE-WINDOW to FULL-PRACTICE at: 9:20:55
Setting Instruct style in SIMULATION-WINDOW to FULL-PRACTICE at: 9:21:00

Conversation element (TALK) shown.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Practice Control Button Pressed: Hints
Showing Talk Answer.
Conversation element (TALK) asked and correct.
Conversation element (CONCLUDE) asked and correct.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command incorrectly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Application command correctly input.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.
Conversation element (TALK) asked and correct.

Completed conversation at: 9:29:27
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Original
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself

Continued, next page
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Myself
History Button Pressed: Hear Original
History Button Pressed: Hear Myself
History Button Pressed: Hear Original
History Button Pressed: Hear Myself
History Button Pressed: Hear Original
History Button Pressed: Close History

Exiting at end of Rehearse Conversation at: 9:34:30

<log continues...>
BIBLIOGRAPHY


417


