AD A083432

LEVEL II

(12)
B.S.

TRANSFER OF RULE-BASED EXPERTISE THROUGH A TUTORIAL DIALOGUE

by

William John Clancey

STAN-CS-79-769
September 1979

DEPARTMENT OF COMPUTER SCIENCE
School of Humanities and Sciences
STANFORD UNIVERSITY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DTIC
ELECTE
S APR 23 1980 D
E

DDC FILE COPY

80 4 21 076

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: STAN-CS-79-769, ~~Tech. Rpt. #1~~ TR-1
2. GOVT ACCESSION NO.: AD-A083 432
3. RECIPIENT'S CATALOG NUMBER: Doctoral Thesis
4. TITLE (and Subtitle): Transfer of Rule-Based Expertise through a Tutorial Dialogue
5. TYPE OF REPORT & PERIOD COVERED: technical, September 1979
6. PERFORMING ORG. REPORT NUMBER: STAN-CS-79-769
7. AUTHOR(s): William John Clancey
8. CONTRACT OR GRANT NUMBER(s): N00014-79-C-0302; ARPA-3697
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Computer Science, Stanford University, Stanford, California 94305 USA
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NR 154-430; ARPA Order-3697
11. CONTROLLING OFFICE NAME AND ADDRESS: Personnel and Training Research Programs, Office of Naval Research (Code 458), Arlington, VA 22217
12. REPORT DATE: September 1979
13. NO. OF PAGES: 462
14. MONITORING AGENCY NAME & ADDRESS (if diff. from Controlling Office): ONR Representative - Philip Surra, Durand Aeromautics Building, Room 165, Stanford University, Stanford, California 94305
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this report): Approved for public release; distribution unlimited.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from report)
18. SUPPLEMENTARY NOTES
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This dissertation describes an intelligent, computer-aided instructional (ICAI) program, named GUIDON, with capabilities to carry on a structured case method dialogue, generate teaching material from production rules, construct and verify a model of what the student knows, and explain expert reasoning. The principle objective of this research has been to convert MYCIN, a knowledge-based consultation program, into an effective instructional tool. GUIDON combines the subject matter knowledge of the consultation system with tutorial discourse knowledge, while keeping the two distinct. → (continued)

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE 094120

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19. KEY WORDS (Continued)

20 ABSTRACT (Continued)

MYCIN-like knowledge-based consultation programs are designed to provide expert-level advice about difficult scientific and medical problems. High performance is attained by interpreting a large, specialized set of facts and domain relations that take the form of rules about what to do in a given circumstance. Such a rule base is generally built by interviewing human experts to formulate the knowledge that they use to solve similar problems in their area of expertise. While it is generally believed that these programs have significant educational potential, little work has been done to evaluate the problems of realizing this potential.

Using a rule base for teaching provides a new perspective for showing what production rules have to do with human expertise. This dissertation closely examines the usefulness and adequacy of MYCIN's rules for infectious disease diagnosis as an instructional vehicle: as topics to be discussed in a tutorial, as problem-solving methods for understanding a student's behavior, and as skills to be learned by a student. It is argued that MYCIN-like rule-based systems constitute a good starting point for developing a tutorial program, but they are not sufficient in themselves for making knowledge accessible to a student. Using GUIDON as an interactive medium for transferring expertise provides a larger context about human cognition; this is reflected in our consideration of subject matter representation and principles of tutorial discourse.

The study of subject matter representation focuses on knowledge that allows the tutor to articulate the structure, underlying principles, and strategies of the domain. This dissertation pays particular attention to aspects of human expertise that have not been captured by the MYCIN rule base, a kind of investigation that has not arisen in the construction, maintenance, and use of this knowledge base for consultation.

The study of tutorial discourse principles focuses on managing the dialogue to achieve economical, systematic presentation of problem-solving expertise. In addition, tutoring methods for opportunistically presenting new material and providing hints on the basis of an hypothesis revision strategy are demonstrated. GUIDON's teaching and discourse expertise is represented as explicit rules. These rules comprise strategies for modeling the student, means for sharing initiative, and knowledge of conventional procedures for discussing a problem in a "goal-directed" way.

After the basic set of tutorial expertise was developed using MYCIN's infectious disease rule set, some perspective on GUIDON's generality and domain independence was attained by coupling it to rule sets for other domains, including an engineering application. Two experiments of this type were performed. They reveal the relationship of discourse strategies to the reasoning structure of the problem being discussed.

TRANSFER OF RULE-BASED EXPERTISE THROUGH A TUTORIAL DIALOGUE

William J. Clancey

Department of Computer Science
Stanford University, Stanford, CA 94305

Contract No. N00014-79-C-0302, effective March 15, 1979.
Expiration Date: March 14, 1979
Total Amount of Contract - $396,325
Principal Investigator, Bruce G. Buchanan (415) 497-0935

Sponsored by:
Office of Naval Research,
Personnel and Training Research Programs,
Psychological Sciences Division.
Contract Authority No. NR 154-436
Scientific Officers: Dr. Marshall Farr and
Dr. Henry Halff

and

Advanced Research Projects Agency
ARPA Order No. 3697
Program Code No. 12420G

The views and conclusion contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency, the Office of Naval Research, or the U.S. Government.

Approved for public release; distribution unlimited. Reproduction in whole or in part is permitted for any purpose of the United States Government.

TRANSFER OF RULE-BASED EXPERTISE THROUGH A TUTORIAL DIALOGUE

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By

William John Clancey

August 1979

Accession For
NTIS GRA&I
DDC TAB
Unannounced
Justification
By
A

© 1979

by

William John Clancey

Abstract

This dissertation describes an intelligent, computer-aided instructional (ICAI) program, named GUIDON, with capabilities to carry on a structured case method dialogue, generate teaching material from production rules, construct and verify a model of what the student knows, and explain expert reasoning. The principle objective of this research has been to convert MYCIN, a knowledge-based consultation program, into an effective instructional tool. GUIDON combines the subject matter knowledge of the consultation system with tutorial discourse knowledge, while keeping the two distinct.

MYCIN-like knowledge-based consultation programs are designed to provide expert-level advice about difficult scientific and medical problems. High performance is attained by interpreting a large, specialized set of facts and domain relations that take the form of rules about what do to in a given circumstance. Such a rule base is generally built by interviewing human experts to formulate the knowledge that they use to solve similar problems in their area of expertise. While it is generally believed that these programs have significant educational potential, little work has been done to evaluate the problems of realizing this potential.

Using a rule base for teaching provides a new perspective for showing what production rules have to do with human expertise. This dissertation closely examines the usefulness and adequacy of MYCIN's rules for infectious disease diagnosis as an instructional vehicle: as topics to be discussed in a tutorial, as problem-solving methods for understanding a student's behavior, and as skills to be learned by a student. It is argued that MYCIN-like rule-based systems constitute a good starting

point for developing a tutorial program, but they are not sufficient in themselves for making knowledge accessible to a student. Using GUIDON as an interactive medium for transferring expertise provides a larger context about human cognition; this is reflected in our consideration of subject matter representation and principles of tutorial discourse.

The study of subject matter representation focuses on knowledge that allows the tutor to articulate the structure, underlying principles, and strategies of the domain. This dissertation pays particular attention to aspects of human expertise that have not been captured by the MYCIN rule base, a kind of investigation that has not arisen in the construction, maintenance, and use of this knowledge base for consultation.

The study of tutorial discourse principles focuses on managing the dialogue to achieve economical, systematic presentation of problem-solving expertise. In addition, tutoring methods for opportunistically presenting new material and providing hints on the basis of an hypothesis revision strategy are demonstrated. GUIDON's teaching and discourse expertise is represented as explicit rules. These rules comprise strategies for modeling the student, means for sharing initiative, and knowledge of conventional procedures for discussing a problem in a "goal-directed" way.

After the basic set of tutorial expertise was developed using MYCIN's infectious disease rule set, some perspective on GUIDON's generality and domain independence was attained by coupling it to rule sets for other domains, including an engineering application. Two experiments of this type were performed. They reveal the relationship of discourse strategies to the reasoning structure of the problem being discussed.

Acknowledgments

We sometimes joke about the futility of progress in a field in which scientists stand on each other's toes, rather than building upon past work. I have been privileged in two ways: first, by being able to build my work upon the MYCIN consultation program, to which more than 20 people have contributed their programming and medical expertise over the past 5 years; second, by being advised by some of the most capable researchers in my field.

First, I would like to thank Bruce Buchanan, my principal adviser, whose poise and sense of proportion have taught me more than just how to do AI research. In reviewing my writing, his efforts to simplify, "reduce to known terms," and "bring the main ideas forward" set standards that I have found to be very satisfying.

Second, the medical expertise in MYCIN was provided (chronologically) by Ted Shortliffe (whose thesis suggested the tutorial project to me), Stan Axline, Stan Cohen, Frank Rhame, Rudy Chavez-Pardo, John Foy, Victor Yu, and Bob Blum. A large part of my research involved making an "archeological" study of the 200 Meningitis-oriented rules that Victor wrote (with the help of Larry Fagan, Carli Scott, and Shari Wraith). While I sometimes complain in this dissertation about "inconsistencies" in these rules, I want to make clear that Victor's painstaking effort in researching and organizing the meningitis rule set has been invaluable. In the past year, Victor has answered my questions by mail, helping me to make the study that constitutes Chapter 5.

Third, MYCIN Computer Science colleagues provided well-written code, which I used throughout my own system, and also provided a friendly, if somewhat noisy,

office. Randy Davis' technical contributions are frequently cited in this document. The sense of style that he has tried to convey to me has been no less important. Carli Scott taught me MYCIN's design and helped me to work out some computationally difficult extensions to the code. Bill van Melle was truly a "programmer's assistant." Over a period of two years, he meticulously answered my almost daily messages about the INTERLISP file package, I/O, and hashfiles. His package for refiling modified code, remaking the program, and documenting changes was essential to my research. In addition, he made numerous useful changes to utility routines, often at my demand. Jim Bennett and Avron Barr encouraged me to look beyond my program to the larger issues of transfer of expertise. I am grateful to Avron for allowing me to use an unpublished manuscript as the basis for my history of CAI.

Fourth, John Brown, Adele Goldberg and Terry Winograd served on my oral and reading committees. John has worked with me for several years now, patiently helping me to focus on ICAI research problems. His interest and investment of time was at first puzzling, but ultimately taught me something about the importance of the community in science. Adele provided stylistic help and a useful perspective on previous work in CAI. Discussions with Ira Goldstein led to the section on modules of ICAI programs.

Finally, Greg Cooper made a contribution for which he refuses to be acknowledged. And special thanks to Jan and Thomas, my confidants.

Table of Contents

Chapter / Section			Page
Abstract			iv
Acknowledgments			vi
1	Introduction		1
	1.1	Task and Thesis	2
	1.2	Features of GUIDON	3
	1.3	MYCIN: Why tutor at all?	9
	1.4	Transfer of Expertise	11
	1.5	Design Assumptions	16
	1.6	Method/Guide to the Reader	18
	1.6.1	System Framework	19
	1.6.2	Augmentation of Domain Knowledge	19
	1.6.3	Formalism of Transfer of Expertise Expertise	19
	1.6.4	Discourse Expertise	20
	1.6.5	Experiments with Multiple Knowledge Bases	20
2	Background: Other Approaches to CAI		21
	2.1	Historical Overview of Educational Applications of Computer Technology	22
	2.1.1	Frame-Oriented Systems	23

2.1.2 Intelligent Computer-Aided Instruction 24
2.2 Components for ICAI Systems 27
2.2.1 The Expertise Component 28
2.2.2 The Modeling Component. 30
2.2.3 The Tutoring Component 33
2.3 Medical CAI 36

3 Annotated Protocol 37
3.1 The Experimental Setting 38
3.2 Program Features to Look For 40
3.3 The Protocol 42
3.4 The Student's Reaction to the Program. 85
3.4.1 Definition of the Problem. 85
3.4.2 Strategy 86
3.4.3 Interaction with the Program. 87

4 Knowledge Base and Control Overview: The Expert/Tutor Framework. . . . 91
4.1 Components of the GUIDON System. 92
4.2 The EMYCIN Formalism 94
4.3 Applicability of EMYCIN Systems for Tutorials 95
4.3.1 Assets of the Formalism 95
4.3.2 Limitations of the Original Explanation Capability 99
4.3.3 Other Rule Set Properties Important for Tutoring 100
4.4 GUIDON: Conversion of the Consultation System 103
4.4.1 The Consultation Record and Its Reconfiguration 103
4.4.2 The Communication Model 110

4.4.3 Tutorial Rules and Discourse Procedures 112

5 Meta-levels of Knowledge Required for Tutoring. 115

5.1 Structural Knowledge -- for organization 121

5.1.1 Parameter Separability 121

5.1.2 Parameter Screening Relations 124

5.1.3 Kinds of Rules. 127

5.1.4 Rule Clusters 133

5.1.5 Rule Interactions 139

5.2 Support Knowledge -- for explaining rules. 145

5.2.1 What's in a Rule?. 146

5.2.2 Principles Missing from Optimized Knowledge 150

5.2.3 Implications for Tutoring 159

5.2.4 The Use of Models for Explanation 160

5.2.5 Conclusions about Support Knowledge 180

5.3 Strategical Knowledge -- for planning solutions 182

5.3.1 Surface and Deep Structure of Rule Systems. 183

5.3.2 Representing Planning Knowledge in Meta-rules 185

5.3.3 Meta-strategies 187

5.4 Epilogue: What is Meta-knowledge? 187

3 Dialogue Management Heuristics for Rule-Based Tutorials 193

6.1 Dialogue Management Overview. 195

6.1.1 Scope of the Problem. 195

6.1.2 Management Constraints. 197

6.1.3 Knowledge of Discourse Structure 198

6.2 Rhetorical Structure of the Dialogue 210
6.2.1 Transitions I: Economy 211
6.2.2 Transitions II: Domain Logic 219
6.2.3 Transitions III: Tutoring Goals 222
6.2.4 Situation Transitions 246
6.3 Summary of Tutoring Principles 250
6.4 Dialogue Management Conclusion 251

7 Providing for and Coping with Student Initiative 253
7.1 Sharing and Maintaining Problem-Solving Context. 256
7.1.1 Getting Case Data 256
7.1.2 Inference and Discussion Records 258
7.1.3 Controlling Tutorial Remarks 2.0
7.1.4 Direct Retrieval of Domain Knowledge 262
7.1.5 Dialogue Context 267
7.2 Conveying What You Know: IKNOW & HYPOTHESIS 270
7.3 Requesting Assistance: HELP/HINT, QUIZ, TELLME, ALLRULES . . . 271
7.4 Changing the Topic: DISCUSS, TELLME, control-F interrupt 272
7.5 Standard User Options: COMMENT, PROFILE, ?, OPTIONS 272

8 Contents, Maintenance and Use of the Student Model 275
8.1 Contents of the Model 276
8.2 Sources of Evidence 280
8.2.1 Background Evidence. 280
8.2.2 Implicit Evidence 282
8.2.3 Explicit Evidence 282

Chapter 1

Introduction

CONTENTS

1.1 Task and Thesis: Expertise, Explanation, and Tutoring

1.2 Example: features of GUIDON
- -- assistance in context
- -- performance evaluation/feedback
- -- tutoring presentations
- -- responsive to student initiative
- -- exploratory freedom

1.3 MYCIN: Why tutor at all?

1.4 Transfer of Expertise
- -- Acquisition, Advice, and Tutoring
- -- What should GUIDON be trying to teach?
- -- ICAI tutorial paradigm
  - > Dialogue Strategies
  - > Diagnostic modeling
  - > Tutoring Strategies

1.5 Design Assumptions
- -- start with MYCIN rules
- -- student: memory, ability, conventions, preferences
- -- nature of the dialogue

1.6 Method/Guide to the reader
- 1.6.1 System framework
- 1.6.2 Augment domain knowledge
- 1.6.3 Formalism of transfer of expertise expertise
- 1.6.4 Discourse expertise
- 1.6.5 Experiments with multiple knowledge bases

8.3 Growth and Revision of the Model 283
8.3.1 The Components of the Model 283
8.3.2 The Overlay Model 285
8.3.3 Revision of Belief 304
8.4 Uses of the Model 307
8.5 Inadequacy of Performance Knowledge for Modeling Misconceptions 309

9 Results: Experiments with Other Knowledge Bases 315
9.1 The PUFF Tutorial 317
9.2 Discussion of the PUFF Tutorial 324
9.3 The SACON Tutorial 326
9.4 Discussion of the SACON tutorial 336

10 Summary and Conclusions 339
10.1 Accomplishments 340
10.2 Dimensions for Comparing and Critiquing ICAI Research 342
10.2.1 Formal Clarity 343
10.2.2 Articulateness and Understanding the Student 344
10.2.3 Mixed-Initiative Dialogue and Task Difficulty 346
10.3 Contribution to Medical CAI 349
10.4 Future Directions for the GUIDON System 350
10.4.1 Dialogue Management 351
10.4.2 Tutoring 353
10.4.3 Modeling 354
10.4.4 Knowledge Representation 356

10.4.5 Learning 357

10.5 Speculation about the Future: Cognitive Science 357

Appendix A

Formation of the AND/OR Solution Tree 361

Appendix B

Question Generation and Response 369

Appendix C

GUIDON Handbook 375

Appendix D

Introductory Orientation for New Students 382

Appendix E

Representation of Discourse Procedures 384

Appendix F

Discourse Procedures and Tutorial Rules 389

References 438

Author Index 449

1.1 Task and Thesis

This dissertation makes contributions to several areas of research in Intelligent Computer-Aided Instruction (ICAI), including means for structuring a case-method dialogue, generating teaching material from performance rules, constructing and verifying a model of what the student knows, and explaining expert reasoning.

The ICAI tutorial program described here, named GUIDON, has been built to explore the problem of converting a knowledge-based consultation program, MYCIN, into an effective instructional tool. Knowledge-based consultation programs [Shortliffe, 1976] [Lenat, 1976] [Pople, 1975] are designed to provide expert-level advice about difficult scientific and medical problems. High performance is attained by interpreting a large, specialized set of facts and domain relations according to the demands of a particular problem. These facts and relations constitute a knowledge base that is generally built by interviewing human experts to formulate the knowledge that they use to solve similar problems in their area of expertise. While it is generally believed that these programs have significant educational potential, little work has been done to evaluate the problems of realizing this potential.

The GUIDON system uses subject matter knowledge and tutorial discourse knowledge, while keeping the two distinct. Basic research centers on two areas: (1) representation of subject material to be used in teaching, so that the tutor can articulate the structure, underlying principles and strategies of the domain, (involving a study of the epistemology of MYCIN's knowledge) and (2) management of the

dialogue so as to achieve economical, systematic presentation of problem-solving expertise (involving the creation of an appropriate and precise theory of tutoring). Some of basic themes of this work, constituting the "thesis" we argue, are:

- ***On Expertise***

Current knowledge-based programs like MYCIN have superficial chains of reasoning that capture little more than the I/O behavior of human expertise.

- ***(Corollary$_1$) On Explanation***

Teaching the expertise represented by a knowledge-based program requires consideration of underlying models of the domain that justify the rules, structural patterns that organize them, and strategies for using them to construct lines of reasoning.

- ***On Tutoring***

The natural language issues in computer-aided instruction go beyond parsing student input. There is a body of expertise for managing a dialogue that takes into account conventional reasoning patterns, sharing initiative, and understanding the student.

1.2 Features of GUIDON

To give some idea about the nature of the tutoring problem and what is difficult about it, we will illustrate the basic design features of the GUIDON program with

tutorial excerpts.[1] Later in this chapter we will consider assumptions that we have made in converting MYCIN to a tutorial program and the paradigm for constructing knowledge-based tutors that has guided our research.

1) The program should provide problem-solving ***assistance in context***. The tutor's guidance should be based upon the student's partial solution. In general, this is a difficult problem because it requires that the tutor be sensitive to the student's current problem-solving strategy, the kind of advice he prefers (a hint? full details?), and be able to articulate problem-solving methods that might be applied (a problem of knowledge representation).

In this example, GUIDON provides assistance by applying a solution method (rule507) that is relevant to the evidence discussed so far. In this case the selected method was alluded to in an earlier hint. The program has many ways to present a rule to the student, such as forming a question or discussing each clause of the rule explicitly. Here GUIDON demonstrates the applicability of the solution method by showing how truth of the single precondition that remains to be considered can be inferred from known evidence (rule509). The inference is trivial, so it is given directly rather than opened up for discussion. GUIDON then applies the original method (rule507) and comments about the status of the current subproblem.

Thus, providing assistance can involve applying a teaching strategy that carries the solution of the problem forward. This in turn requires being able to articulate reasoning on the basis of what the student knows, according to principles of economical presentation.

[1] To help the reader get past the medical jargon to see the instructional issues, these excerpts have been cleaned-up, chiefly by substituting pronouns for the identifier "pt538" and rephrasing parameter descriptions; the original text is given in Chapter 3 and Chapter 9.

** HINT
Hint: Look for strong, positive evidence that Diplococcus-pneumoniae is one of the organisms that might be causing the infection.

.
.
.

** HELP

Let's consider whether J.Smith has a head injury.

We already know that the patient has not had an injury to the central nervous system; this is evidence that he does not have a head injury [RULE509].

We now have strong evidence that Diplococcus-pneumoniae and Neisseria-meningitidis are organisms that might be causing the infection (considering the age of J.Smith and the fact that he does not have a head injury [RULE507]).

It remains for us to consider other factors for determining the organisms that might be causing the infection.

.
.
.

2) The program should provide ***feedback*** to the student that enables him to evaluate his competency. Here the tutor must be able to account for the student's behavior in terms of problem-solving methods and strategies. This may involve interpreting wrong, but consistent behavior in terms of misconceptions. In correcting the student, the tutor may need to refer back to past student actions and compare alternative approaches. Providing elaboration in a mixed-initiative exchange poses technical problems in dialogue management, such as maintaining context while sharing initiative.

In this excerpt, the student's hypothesis for a subproblem is correct, but after analyzing the previous history of interaction with the student, GUIDON is not sure that the student really knows why this hypothesis is correct. Therefore, instead of accepting this answer, GUIDON requests relevant factors to support the hypothesis, and probes further to see if the student knows why an alternative hypothesis is discredited.

What is the type of the infection?
** BACTERIAL

What facts about this case tell you that the type of the infection is bacterial?

(Enter one factor per line.)
** CEREBRAL-FLUID GLUCOSE VALUE
** PMNS
The percent of PMN's in the CBC is not relevant for determining the type of the infection.
** PMNS IN THE CSF
** PROTEIN OF THE CSF
**

You're right about the CSF protein, the percent of PMNs in the CSF and the CSF glucose value, however you should have mentioned the fact that the meningitis is partially treated, the WBC from the CSF and the fact that the simultaneous blood glucose value is not known.

.
.
.

MYCIN rejected some possibilities you did not mention:

What facts about this case tell you that the type of the infection is not viral?

.
.
.

3) The program should make ***tutorial presentations*** for the purpose of introducing new material to the student. In a knowledge-based tutor, presentations are generated solely from the knowledge base of rules and facts. This requires that the tutor have presentation methods that opportunistically adapt material to the needs of the dialogue. In particular, the tutor has to be sensitive to how a tutorial dialogue fits together, what kinds of interruptions and probing are reasonable and expected in this kind of discourse.

The following example is taken from a GUIDON tutorial that uses PUFF's knowledge base for the problem of pulmonary function analysis. Here GUIDON first interrupts the student's data collection to suggest that he make an hypothesis; but the student refuses to do so. The program then observes that there is a particular problem-solving method that can be applied that is probably known to the student (rule020). However, he probably can't apply the method to this case because he doesn't know how to verify a necessary precondition. GUIDON presents the inference that it believes is unknown to the student (rule028), and then asks him to take this evidence forward.

** TOPIC
We are discussing the degree of reversibility of airway obstruction of PATIENT-7.
Hypotheses you should be considering are: none, insignificant, slight, good and significant.

** WHAT IS THE CHANGE IN FORCED EXPIRATORY VOLUME?

The change in Forced-Expiratory-Volume (1sec.) due to dilation of M.Smith is 31.

** WHAT IS THE CHANGE IN RESISTANCE?

The change in Resistance due to dilation of M.Smith is 20.

We have substantially completed our discussion of the degree of reversibility of airway obstruction of M.Smith.
Would you like to make a hypothesis now (as opposed to gathering more information)?
** NO

We will now consider evidence that leads to a conclusion about the degree of reversibility of airway obstruction of M.Smith.

Here is a conclusion that you may not have considered....

The fact that the change in Forced-Expiratory-Volume (1sec.) due to dilation of M.Smith and the change in Resistance due to dilation are known is evidence that there are post bronchodilation test results [RULE028].

Given the post bronchodilation test results, does this tell you that the degree of reversibility of airway obstruction is good?
** YES
No, given this fact, we can conclude that the degree of reversibility of airway obstruction of M.Smith is significant [RULE020].

The conclusion in the question holds when the change in Forced-Expiratory-Volume (1sec.) due to dilation of the patient is between 5 and 10.

---

4) The program should be ***responsive to the student's initiative*** in

expressing what he knows and wants to discuss. In replying to student initiative, a tutor needs to determine what response makes sense, based on what it knows about the student's knowledge and shared goals for the tutorial session. For example, when the student takes the initiative by saying he knows something, the tutor may want to hold a detailed response in abeyance, simply acknowledge the student's remark, or probe him for a proof. Selection among these alternative dialogues might require determining what the student could have inferred from previous interactions and the current situation.

Here GUIDON decides that there is sufficient evidence that the student knows the solution to a relevant subproblem, so detailed discussion and probing is not necessary.

** I KNOW THE INFECTION THAT REQUIRES THERAPY

Good. MYCIN has also made a decision.

That fact that the cerebral spinal fluid of J.Smith has been cultured by attending physicians is evidence that the infection that requires therapy is meningitis [RULE333].

Back to our discussion of the organisms that might be causing the infection....

5) The program should give the student the ***freedom to explore*** MYCIN's reasoning as he desires. In laying out MYCIN's reasoning, the program should not act like a passive information retrieval system. In addition to clearly laying out data and inferences, the tutor has to reason about what constitutes reasonable, expected elaboration on the basis of what has been previously discussed.

In this excerpt GUIDON provides details for an inference (rule148) by offering to support necessary preconditions that were not considered in the dialogue up to this point, though they can be inferred from known data.

Summary of evidence that the type of the infection is bacterial (.98) viral (-.76) fungal (-.83) tb (-.83):

29a. The fact that the meningitis is partially treated (.95) is evidence that the type of the infection is bacterial (.19)

[RULE148]

29b. The WBC from the CSF and the percent of PMNs in the CSF is evidence that the type of the infection is bacterial (.9) viral (-.6) fungal (-.7) tb (-.7) [RULE501]

.
.
.

** DETAILS 29A
For which of the following factors would you like to see a review of the evidence?
(select from the following):
1) the infection which requires therapy
2) the fact that the meningitis is partially treated (.95)
** 2

Summary of evidence that the meningitis is partially treated (.95):

32a. The infection which requires therapy, the fact that organisms were not seen on the stain of the pending csf culture and the time since therapy with the cephalothin was started is evidence that the meningitis is partially treated (.95) [RULE145]

Do you want to see RULE148?
** NO

1.3 MYCIN: Why tutor at all?

MYCIN is a knowledge-based program that provides consultations about infectious disease diagnosis and therapy [Shortliffe, 1976] [Davis, Buchanan, & Shortliffe, 1977]. In MYCIN, domain relations and facts take the form of rules about what to do in a given circumstance. A principle feature of this formalism is the separation of the knowledge base from the interpreter for applying it. This makes the knowledge accessible for multiple uses, including application to particular problems

(i.e., for "performance") and explanation of reasoning [Davis, 1976]. Converting the MYCIN system into a tutorial program was suggested naturally by this capability. This dissertation closely examines the usefulness and adequacy of MYCIN's rules as an instructional vehicle: as topics to be discussed in a tutorial, as problem-solving methods for understanding a student's behavior, and as skills to be learned by a student. It is argued in this paper that MYCIN-like rule-based expert systems constitute a good basis for tutorial programs, but they are not sufficient in themselves for making knowledge accessible to a student.

One can follow MYCIN's reasoning during a consultation by using the explanation system (by asking WHY case data are being sought by the program and HOW goals will be (were) achieved). However, we believe that this is an inefficient process for learning the contents of the knowledge base. The MYCIN program is only a passive "teacher": it is necessary for the student to ask an exhaustive series of questions, if he is to discover all of the reasoning paths considered by the program. We believe that most students would not have this persistence, so the wealth of expertise in the knowledge base would be lost to them.

The capabilities of existing ICAI programs suggested that it would be desirable to have an active, intelligent agent that kept track of the knowledge that has been presented to the student in previous sessions and attempted to measure and record his competence. Using this individual record with strategies for teaching, the program could progressively and systematically present the knowledge base to the student according to his interests and capability to advance. Moreover, it turned out that to understand the student and provide a progressive series of lessons, it would be

necessary for the tutor to know facts and relations that played no part in a consultation. "Tutoring" the knowledge base became a focus for defining problems of human reasoning and explanation that went beyond those that were important for merely giving advice.

1.4 Transfer of Expertise

GUIDON is an example of a *transfer of expertise* program. Previous work in transfer of expertise that we are building upon investigated the problem of constructing and maintaining MYCIN's knowledge base through interactive sessions with human experts [Davis, 1976]. This is transfer of expertise from a human to a program. From this perspective, the original consultation project investigated the transfer of expertise from program to human in the context of giving advice (see Figure 1.1). The goal of the GUIDON project is to extend the transfer of expertise theme in yet another direction--from the program to a student.

Observe that a single knowledge base is coupled to different programs that must interact with a human: this human connection provides an important focus to our research. The question we ask is, what should an interactive program know if it is to effectively learn from, give advice to, or teach a human? How should the single knowledge base be organized if it is to be used for all three of these tasks? While this dissertation is concerned primarily with tutorial interactions, these questions provide a larger context about human cognition that will be reflected in our consideration of subject matter representation and principles of tutorial discourse.

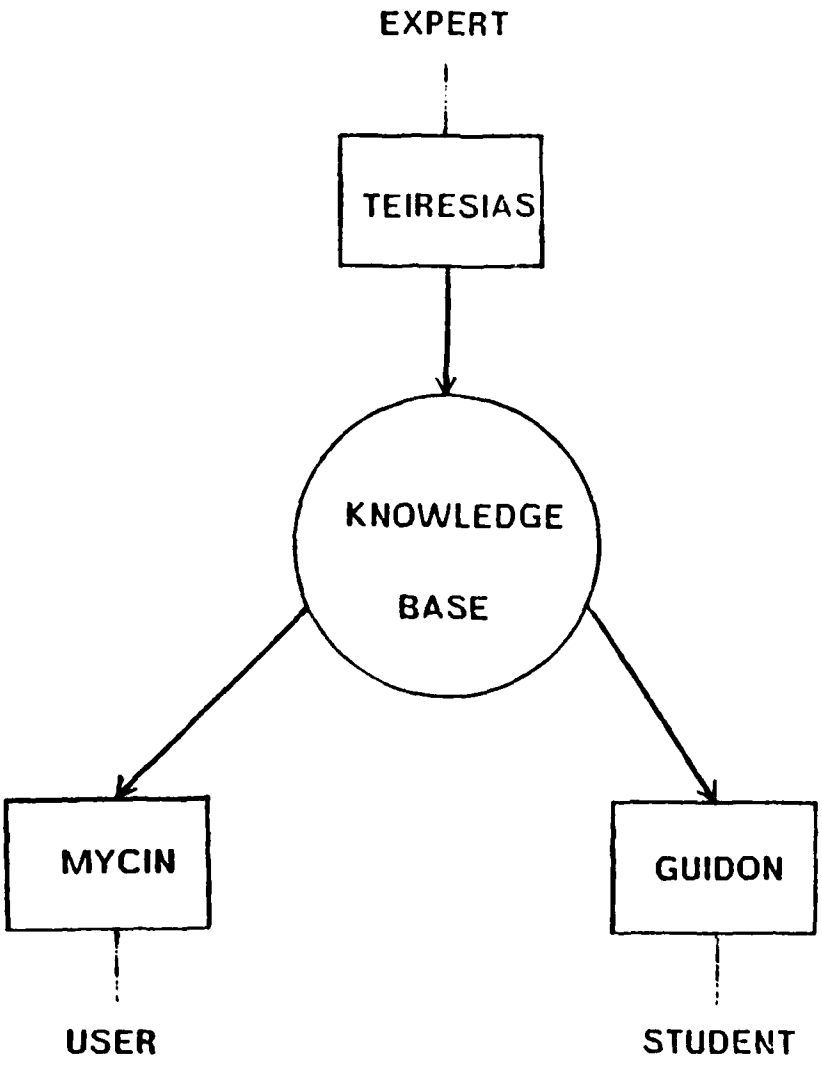


Figure 1.1. Transfer of Expertise: Learning, Advising, and Teaching

GUIDON is not the first program that might be characterized as a tutor for transferring expertise. Indeed, research in individualized, computer-aided instruction has been greatly stimulated in the 1970s by knowledge-based research. A large store of specialized facts and relations has been used for teaching geography [Carbonell, 1970], meteorology [Brown, Burton, & Zydbel, 1973] [Collins, 1976], and electronic circuit debugging [Brown, Burton, & Bell, 1974b][Brown, Rubinstein, & Burton, 1976]. The GUIDON system represents the first attempt to construct a complete tutorial program through *conversion* of a knowledge-based system. For this reason, our research pays particular attention to studying the aspects of human expertise that have been captured by the MYCIN rule base, a kind of investigation that has not arisen in the construction, maintenance, and use of this knowledge base for consultations.

What Should GUIDON be Trying to Teach?

One of the reasons for doing this work is because we don't fully understand the problems of learning expertise like MYCIN's. Indeed, we don't know how MYCIN's rules correspond to what an expert really knows or uses to solve a problem. It is not clear that the organization and level of abstraction of this performance knowledge is suitable for use in a tutorial program. Moreover, the rules may only capture the superficial "input/output" behavior of experts: what they observe, a few intermediate concepts, and the advice they give.

A naive view would be that MYCIN's rules are all that a student needs to know if he is to become an expert. However, we can easily think of other kinds of knowledge

that people use all the time when reasoning about a problem or learning new material. For example, there are facts (E.coli has the shape of a rod), strategies (consider the accuracy of the data), models for justifying rules (some drugs interfere with immunosuppression), patterns in rules (the shape of an organism and its staining characteristics tend to be considered together), examples (consider a 20 year old male who...), prototypical cases, and so on. In addition, we know that human teachers talk in terms of overviews or frameworks for structuring subject matter. Courses are frequently designed around notions of difficulty, sophistication, and prerequisite connections of the subject.

In developing a tutor for MYCIN, we will want to consider these various forms of knowledge. A basic question is: What is it essential for a tutor to be able to articulate if a problem solution is to be understood and made memorable? Transferring expertise to a student requires that we go beyond features in the rule set that were incorporated just to make sure that the consultation program got the right answer. In Chapter 5 we examine MYCIN's rules from this perspective.

The ICAI Tutorial Paradigm

In addition to the domain knowledge of the expert program, a tutorial program requires teaching expertise, such as the ability to tailor the presentation of domain knowledge to the student's competence and interests [Brown, 1977b]. Even given an accurate representation of a human expert's knowledge and reasoning procedures, we don't know how to present it to a student, or what special instructional information that may play no part in problem-solving itself is useful.

In all ICAI programs, a student is given some task to solve. It may be a game requiring reasoning skills, a diagnostic problem (faulty electronic circuit or patient showing symptoms of infection), or a factual puzzle to be explained (causes of heavy rainfall). Each of these systems has an underlying "expert" program that can talk about problems that are posed to the student. While the student works on the task, the instructional program uses the expert program as a relative measure of the consistency and efficiency of the student's solution (his answer and the overt steps he took to reach this answer). If the expert program can articulate its solution of the task (talk about strategies for taking alternative steps to solve the task), the instructional program can provide guidance when the student has difficulty with a subproblem or does not know what to do next. One of the major research problems is the design of tutorial strategies: when to say something and what to say.

The following three considerations seem central to tutorial discourse:

1) ***Dialogue Strategies - Problem-solving***

> How is the tutor to manage the discussion of a long and difficult problem? What dialogue conventions enable people to guide a discussion coherently through lines of reasoning? When there are too many rules and topics to discuss in the allotted time, how is the tutor to decide which to present and which to omit? How are summaries constructed? What sort of planning is necessary?

2) ***Diagnostic Modeling***

> How is the tutor to determine what knowledge a student is using to solve a problem? When expert reasoning paths are not unique or a limited window into the student's thinking prevents monitoring each step of a chain of reasoning, how is the tutor to apportion credit and blame for the student's observable behavior among the different knowledge sources? How is a program to construct theories that account for student misconceptions?

3) *Tutoring Strategies - Correcting Misconceptions*

> What are pedagogically effective methods for bringing an inadequacy to the student's attention? What strategies can usefully elicit information about a student's thinking process without causing interference? How will these strategies be related to the content of the knowledge base? How will they be represented?

Observe that these questions arise in converting a consultation system designed for high performance into a tutorial program. The additional body of expertise that we must formulate for tutoring can be characterized as expertise for transferring expertise. Representation and development of this expertise is a basic focus for GUIDON research.

1.5 Design Assumptions

The GUIDON system is designed to be built on top of any EMYCIN knowledge base [1] [van Melle, 1979]. However, we have used the original MYCIN infectious disease knowledge base for developing GUIDON. Besides making GUIDON separate from the rule base, we have not modified the rules in any way during this initial implementation of the tutorial system.

Assumptions about the students who will use GUIDON and the nature of the tutorial dialogue are discussed below.

---

[1] This domain-independent framework for building MYCIN-like consultation systems is described in Section 4.2.

The Student

In a GUIDON tutorial, a student plays the role of a consultant. The dialogue deals exclusively with a particular consultation that has already been presented to MYCIN. Thus, for the infectious disease domain, the student is given some information about a patient suspected to have an infectious disease, and is expected to request more case data, as he deems necessary, to draw conclusions about the patient. The purpose of a GUIDON tutorial is to make the student aware of gaps or inconsistencies in his knowledge, and to correct these deficiencies.

The problems to be solved by the student require technical, specialized training. We assume that the student has some background in the area to be tutored: he knows what problem is to be solved (e.g, to reach a diagnosis for a patient suspected to have an infectious disease), the real world sources for observations to solve problems (e.g., organisms found on cultures taken from a patient), and the nature of the problem-solving methods (e.g., uncertain judgments as opposed to numerical calculations). Thus, he is prepared to practice a basic set of already acquired skills. GUIDON will introduce "factors" to the student (representing a parametrization of the real world sources) and judgmental rules for making inferences that lead to a solution of the problem.

We assume that the students will be well-motivated adults who are capable of a serious, mixed-initiative dialogue. They will be willing to follow the program's guidance, but will want to share in the decision of what is discussed.

Furthermore, we assume that there are basic human limitations and preferences that affect tutoring effectiveness. These include considerations like memory and learning capability, social conventions for carrying on a discussion, and individual styles for sharing initiative and approaching new material.

Nature of the Dialogue

We limit communication between the program and student to a computer terminal that prints one line at a time, like a teletype; in particular, graphical methods of presenting information will not be considered. The tutorial is expected to require between 30 and 90 minutes. In fact, the basic limitation of time is an important practical consideration that can be expected to constrain the tutor's handling of the dialogue. We are more concerned with the tutor's choice and relation of topics than its ability to parse and understand natural language input.

The general structure of the case method dialogue is shown in Figure 1.2.

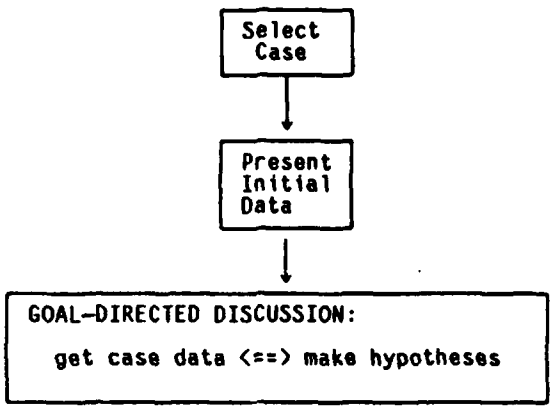


Figure 1.2. General Structure of GUIDON Tutorials

In the current version of GUIDON, the student must select a case from the library of patients (there are over 100 meningitis cases, but as described in Section 4.4.1, a consultation must be run and the results reconfigured before a tutorial can begin). Presentation of initial data involves describing the objects of the case (e.g., cultures, organisms) in general terms.

The dialogue from this point is goal-directed: Topic transitions *descend* to pursue deeper subgoals and *pop* back, returning conclusions. The current subgoal determines the context in which student options are interpreted. For example, requests for case data that do not pertain to this subgoal are considered to be irrelevant, and the student is told so. However, the student can change the goal under discussion at any time (Section 7.4). The topics of the dialogue are precisely the goals that are determined by applying MYCIN rules, e.g., the type of the infection.

During discussion of a goal, the student repetitively requests additional case data, and has a large number of options available for exploring MYCIN's solution tree. He is expected to state hypotheses which GUIDON analyzes in terms of the case data he has received and the conclusions made by MYCIN from this information.

1.6 Method/Guide to the Reader

Here we survey the key methods for approaching the problems we have outlined. Pointers are provided to relevant chapters as a guide to the reader.

1.6.1 System Framework

MYCIN communicates its solution of a problem as an AND/OR tree of goals and rules. Chapter 4 describes how this tree is generated from the output of a consultation, and relates this data structure to the modules of the GUIDON system.

1.6.2 Augmentation of Domain Knowledge

In GUIDON we have augmented the performance knowledge of rules by adding three other levels: a *support level* to justify individual rules, a *structural level* to characterize patterns in the rule set, and a *strategical level* to control the application of rules. In addition, the program makes use of representational meta-knowledge [Davis & Buchanan, 1977] that enables it to pick apart MYCIN's rules and use them to guide the conversation, model the student's understanding, and quiz him. Chapter 5 describes these levels of knowledge and their tutorial application in detail.

1.6.3 Formalism of Transfer of Expertise Expertise

GUIDON teaching expertise is represented as explicit rules. We view the set of tutorial procedures as a knowledge base, and have developed it in much the same way that MYCIN's rule set was constructed. That is, formulation of dialogue, modeling, and tutoring methods is an iterative process that requires frequent changes to the program. For this reason, the representation of tutoring rules was designed to make them easy to enumerate and modify. The formalism of tutoring rules is described in Chapter 6.

1.6.4 Discourse Expertise

GUIDON can be considered to be a kind of discourse program. Discourse expertise includes knowledge of conventional procedures for discussing a problem. In GUIDON these are procedures for a goal-directed case method dialogue (Chapter 6). In addition, mixed-initiative tutorial dialogue involves sharing initiative with the student (Chapter 7) and constructing a model of what he knows and is trying to do (Chapter 8).

1.6.5 Experiments with Multiple Knowledge Bases

After the basic set of tutorial expertise had been developed using the infectious disease rule set, some perspective on GUIDON's generality was attained by coupling it to rule sets for other domains. Two experiments of this type were performed; they are described in Chapter 9.

Chapter 2

Background: Other Approaches to CAI

CONTENTS

2.1 Historical Overview of Educational Applications of Computer Technology:
-- environmental: free-style programming
-- experimental: playing with games and simulations
-- instructional: direct teaching (CAI)
2.1.1 Frame-Oriented Systems
2.1.2 Intelligent Computer-Aided Instruction

2.2 Components of ICAI Systems
2.2.1 The Expertise Component
2.2.2 The Modeling Component
2.2.3 The Tutoring Component

2.3 Medical CAI

[Note: The historical overview and description of ICAI system components will appear in A. Barr & E. A. Feigenbaum (Eds.), *The Handbook of Artificial Intelligence*, 1979.]

2.1 Historical Overview of Educational Applications of Computer Technology

Educational applications of computer technology have been under development since the early 1960s. These applications have included scheduling courses, managing teaching aids, and grading tests. The predominant application, however, has involved using the computer as a device that interacts with the student directly, rather than serving as an assistant to the human teacher. For this kind of application, there have been three general approaches.

The *ad lib* or *environmental approach* is typified by Papert's LOGO laboratory [Papert, 1970], that allowed students more or less free-style use of the machine. Students are involved in programming; it is conjectured that learning problem-solving methods takes place as a side effect of using tools that are designed to suggest good problem-solving strategies to the student. The second approach uses *games and simulations* as instructional tools; once again the student is involved in an activity--for example, doing simulated genetic experiments--for which learning is an expected side effect. The third computer application in education is *computer-assisted instruction* (CAI). Unlike the first two approaches, CAI makes an explicit attempt to instigate and control learning [Howe, 1973]. This third use of computer technology in education is the focus of the following discussion.

The goal of CAI research is to construct instructional programs that incorporate well-prepared course material in lessons that are optimized for each student. Early programs were electronic "page-turners" that printed prepared text and simple, rote drills; and practice monitors, which printed problems and responded to the student's

solutions using pre-stored answers and remedial comments. In the Intelligent CAI (ICAI) programs of the 1970s, course material is represented independently of teaching procedures so that problems and remedial comments can be generated differently for each student. Research today focuses on the design of programs that can construct a truly insightful model of the student's strengths, weaknesses, and preferred style of learning. It is believed that AI techniques will make possible a new kind of learning environment. In this overview, we survey CAI techniques used by past programs and discuss how AI techniques became useful for ICAI programs.

2.1.1 Frame-Oriented Systems

The first instructional programs took many forms, but all adhered to essentially the same pedagogical philosophy. The student was usually given some instructional text (sometimes without using the computer) and asked a question that required a brief answer. After the student responded, he was told whether his answer was right or wrong. The student's response was sometimes used to determine his "path" through the curriculum, or the sequence of problems he was given (see [Atkinson & Wilson, 1969]). When the student made an error, the program branched to remedial material.

A courseware author attempted to anticipate every wrong response and prespecified branching to other material based on the underlying misconceptions that the author associated with each wrong response. Branching on the basis of response was the first step toward individualization of instruction [Crowder, 1962]. This style of CAI has been dubbed ad-hoc, frame-oriented (AFO) CAI by [Carbonell, 1970], to

stress its dependence on author specified units of information.[1] Design of ad-hoc frames was originally based on Skinnerian stimulus/response principles. The branching strategies of some AFO programs have become quite involved, incorporating the best learning theory that mathematical psychology has produced [Atkinson, 1972] [Fletcher, 1975] [Kimball, 1973]. Many of these systems have been used successfully and are available commercially.

2.1.2 Intelligent Computer-Aided Instruction

In spite of the widespread application of AFO CAI to many problem areas, many researchers believe that most AFO courses do not make the best use of computer technology:

> In most CAI systems of the AFO type, the computer does little more than what a programmed textbook can do, and one may wonder why the machine is used at all....When teaching sequences are extremely simple, perhaps trivial, one should consider doing away with the computer, and using other devices or techniques more related to the task. ([Carbonell, 1970], pp. 32; 193)

In this pioneering paper, Carbonell goes on to define a second type of CAI that is known today as knowledge-based or intelligent CAI (or ICAI). Knowledge-based systems and the previous CAI systems both have representations of the subject matter they teach, but ICAI systems also carry on a natural language dialogue with the student, and use the student's mistakes to diagnose his misunderstandings. ICAI has

[1] The term frame as it is used in this context predates the more recent usage [Minsky, 1975], and refers to a block or page or unit of information or text. Its use here does not refer to some general form of schema.

also been called *generative* CAI [Wexler, 1970] since it is typified by programs that present problems by generating them from a large database representing the subject material to be taught. (See [Koffman & Blount, 1973] for a review of some early generative CAI programs and an example of the possibilities and limitations of this kind of program.)

However, the kind of program that Carbonell was describing in his paper was to be more than just a problem generator. Rather, it was to be a computer-tutor that had the inductive powers of its human counterparts and could offer what Brown [Brown, Rubinstein, & Burton, 1976] calls a *reactive learning environment*, in which the student is actively engaged with the instructional system, and his interests and misunderstandings drive the tutorial dialogue. This goal was expressed by other researchers trying to write CAI programs that extend the medium beyond the limits of frame selection:

> Often it is not sufficient to tell a student he is wrong and indicate the correct solution method. An intelligent CAI system should be able to make hypotheses based on a student's error history as to where the real source of his difficulty lies. [Koffman & Blount, 1973]

The realization of the computer-tutor has involved increasingly complicated computer programs and has prompted CAI researchers to use artificial intelligence techniques. Artificial intelligence work in natural language understanding, the representation of knowledge, and methods of inference, as well as specific AI applications like algebraic simplification, calculus and theorem proving, have been

applied by various researchers toward making CAI programs that are more intelligent and more effective. Early research on ICAI systems focused on representation of the subject matter. The benchmark efforts include the geography tutor of Carbonell and Collins [Carbonell, 1970] the Logic and Set Theory tutors by Suppes et al. [Suppes & Morningstar, 1972] and the electronics troubleshooting tutor of Brown and Burton [Brown, Burton, & Bell, 1974b]. The high level of domain expertise in these programs permits them to be responsive in a wide range of problem-solving interactions.

These ICAI programs are quite different from even the most complicated frame-oriented, branching program.

> Traditional approaches to this problem using decision theory and stochastic models have reached a dead end due to their oversimplified representation of learning.... It appears within reach of AI methodology to develop CAI systems that act more like human teachers. [Laubsch, 1975]

However, an AI system that is expert in a particular domain is not necessarily an expert teacher of the material--"ICAI systems cannot be AI systems warmed over" [Brown, 1977b]. A teacher needs to understand what the student is doing, not just what he is supposed to do. AI programs often use very powerful problem-solving methods that do not resemble those used by humans. In many cases, CAI researchers borrowed AI techniques for representing subject domain expertise, but had to modify them, often making the inference routines less powerful to force them to follow human reasoning patterns, so as to better explain their methods to the student, as well as to understand his methods [Goldberg, 1973].

In the mid-1970s, a second phase in the development of generative tutors has been characterized by the inclusion of expertise in the tutor regarding the student's learning behavior, as well as tutorial strategies [Brown & Goldstein, 1977]. AI techniques are used to construct models of the learner that represent his knowledge in terms of *issues* [Burton & Brown, 1976]) or *skills* [Barr & Atkinson, 1975] that should be learned. This model then controls tutoring strategies for presenting the material to be learned. Finally, some ICAI programs are now using AI techniques to explicitly represent these tutoring strategies, gaining the advantages of flexibility and modularity of representation and control [Brown, Rubinstein, & Burton, 1976] [Goldstein, 1977].

2.2 Components for ICAI Systems

The main components of ICAI systems are *problem-solving expertise* (the knowledge that the system tries to impart to the student), a student model (a diagnosis of what the student does and does not know), and tutoring expertise (methods and strategies for presenting problem-solving knowledge to the student).[1] Not all of these components are fully developed in every system. Because of the size and complexity of intelligent CAI programs, most researchers tend to concentrate their efforts on the development of a single part of what would constitute a fully usable system. Each component is described below.

[1] See [Self, 1974], for an excellent discussion of the differences and interrelations of the types of knowledge needed in an intelligent CAI program.

2.2.1 The Expertise Component

The *expert* module of an ICAI system generates problems and measures the correctness of student solutions. Knowledge of the subject matter was originally envisioned as a huge static database that incorporated all the facts to be taught. This idea was implicit in the early drill-and-practice programs and was made explicit in generative CAI. Representation of subject area expertise in this way, using semantic nets, has been useful for generating and answering questions involving causal or relational reasoning [Carbonell & Collins, 1973] [Laubsch, 1975].

Recent systems have used procedural knowledge of the subject matter to show the student how to do things (e.g., take measurements, make deductions). This knowledge is represented as procedural experts that correspond to subskills a student must learn in order to acquire the complete skill [Brown, Burton, & Bell, 1974b]. Production rules have been used to construct modular representations of skills and problem-solving methods [Goldstein, 1977]. In addition, Brown has shown that multiple representations (e.g., a semantic net of facts about an electronic circuit and procedures simulating the functional behavior of the circuit) are sometimes useful for answering student questions and for evaluating partial solutions to a problem. Stevens and Collins [Stevens & Collins, 1978] have considered an evolving series of metaphorical "simulation" models that can be used to reason about the behavior of causal systems.

It should be noted that not all ICAI systems can actually solve the problems they pose to a student. For example, BIP, the BASIC Instructional Program [Barr, Beard, &

Atkinson, 1976] can't write programs, but sample input/output pairs (supplied by the program authors) permit it to test a student's program. Similarly, the procedural experts in SOPHIE-I could not debug an electronic circuit. In contrast, the production system representation of subject knowledge used in WUMPUS [Goldstein, 1977] [Stansfield, Carr, & Goldstein, 1976] and GUIDON enables the programs to solve problems independently, as well as to criticize student solutions. Being able to solve problems, preferably in all possible ways, correctly and incorrectly, is important if the ICAI program is to make fine-grained suggestions about the completion of partial solutions. In this respect, the ability to *articulate* reasoning [Goldstein & Papert, 1977] in an understandable way is a useful dimension for comparing ICAI systems (Section 10.2.2).

All ICAI systems are distinguished from earlier approaches by the separation of teaching strategies from the subject expertise to be taught. However, the separation of subject-area knowledge from instructional planning requires a structure for organizing the expertise that captures the difficulty of various problems and the interrelationships of course material. Modeling a student's understanding of a subject is closely related conceptually to figuring out a representation for the subject itself or for the language used to discuss it.

Trees [Koffman & Blount, 1973] and lattices showing prerequisite interactions have been used to organize the introduction of new knowledge or topics. In BIP this lattice took the form of a *curriculum net* that related the skills to be taught to example programming tasks that exercised each skill. Goldstein called the lattice a *syllabus* in the WUMPUS program and emphasized the developmental path that a learner takes in

acquiring new skills. For arithmetic skills used in WEST, Burton and Brown use levels of *issues*. Issues proceed from the use of arithmetic operators to strategies for winning the game, to meta-level considerations for improving performance. Burton and Brown believe that when the skills are "structurally independent," the order of their presentation is not particularly crucial. This representation is useful for modeling the student's knowledge and coaching him on different levels of abstraction. Stevens and Collins have argued further that a good human tutor does not merely traverse a predetermined network of knowledge in selecting material to present. Rather, it is the process of ferreting out student misconceptions that drives the dialogue.

2.2.2 The Modeling Component

The modeling module is used to represent the student's understanding of the material to be taught. Much recent ICAI research has focused on this component. The purpose of modeling the student is to make hypotheses about his misconceptions and nonoptimal strategies so that the tutor module can point them out and suggest why they are wrong. It is advantageous for the system to be able to recognize alternate ways of solving problems, including the incorrect methods that the student might use based on systematic misconceptions about the problem or on inefficient strategies.

Some of the original frame-oriented systems used mathematical stochastic models of the student, but this approach failed because it only modeled the probability that a student would give a specific response to a stimulus. In general, knowing the probability of a response is not the same as knowing what a student is thinking about; it has little diagnostic power [Laubsch, 1975].

Typical use of AI techniques for modeling student knowledge includes simple pattern recognition applied to the student's response history, and flags in the subject matter semantic net or rule base representing areas that the student has mastered. In these ICAI systems, a student model is formed by comparing the student's behavior to that of the computer-based "expert" in the same environment. The modeling component marks each skill according to whether evidence indicates that the student knows the material or not. Goldstein has termed this component an *overlay* model because the student's understanding is represented completely in terms of the expertise component of the program.

In contrast, another approach is to model the student's knowledge not as a subset of the expert's, but rather as a perturbation or deviation from the expert's knowledge, that is, in terms of *bugs*. There is a major difference between the overlay and "buggy" approaches to modeling: In the latter approach it is not assumed that the student reasons as the expert does, but simply knows less. Thus, the student's reasoning can be substantially different from expert reasoning. How to represent, diagnose, and generate these deviations is a major problem.

Other elements that might be included in the student model are preferred modes for interacting with the program, a rough characterization of the student's level of ability, a consideration of what he seems to forget over time, and an indication of what his goals and plans seem to be for learning the subject matter.

Major sources of evidence used to maintain the student model can be characterized as: (a) implicit, from student problem-solving behavior; (b) explicit,

from direct questioning of the student; (c) historical, from assumptions based on the student's experience; and (d) structural, from assumptions based on some measure of the difficulty of the subject material [Goldstein, 1979]. Historical evidence is usually determined by asking the student to rate his level of expertise on a scale from "beginner" to "expert". Early programs like SCHOLAR used only explicit evidence. Recent programs have concentrated on inferring implicit evidence from the student's problem-solving behavior. This approach is complicated because it is limited by the program's ability to recognize and describe the strategies being used by the student. Specifically, when the expert program indicates that an inference chain is required for a correct result, and the student's observable behavior is wrong, how is the modeling program to know which of the intermediate steps are unknown or incorrectly applied by the student? This is the *apportionment of credit/blame problem*; it has been an important focus of WEST research.

Because of inherent limitations in the modeling process, it is useful for a *critic* in the modeling component to measure how closely the student model describes the student's behavior. Extreme inconsistency or an unexpected demonstration of expertise in solving problems might indicate that the representation being used by the program does not capture the student's approach. Finally, Goldstein has suggested that the modeling process should attempt to measure whether or not the student is actually learning and to discern what teaching methods are most effective. Much work remains to be done in this area.

2.2.3 The Tutoring Component

The tutoring module of ICAI systems must integrate knowledge about natural language dialogues, teaching methods, and the subject area to be taught. This is the module that communicates with the student: selecting problems for him to solve, monitoring and criticizing his performance, providing assistance upon request, and selecting remedial material. This module must deal with questions like "When is it appropriate to offer a hint?" or "How far should the student be allowed to go down the wrong track?"

> These are just some of the problems which stem from the basic fact that teaching is a skill which requires knowledge additional to the knowledge comprising mastery of the subject domain. [Brown, 1977b]

This additional knowledge, beyond the representation of the subject domain and the student knowledge, is about how to teach.

Most ICAI research has explored teaching methods based on diagnostic modeling in which the program debugs the student's understanding by posing tasks and evaluating his response [Collins, 1976] [Brown, et al., 1975] [Koffman & Blount, 1973]. The student is expected to learn from the program's feedback of which skills he uses wrongly, which he does not use (but could use to good advantage), etc. Recently, there has been more concern with the possibility of saying just the right thing to the student so that he will realize his own inadequacy and switch to a better method [Carr & Goldstein, 1977] [Burton, 1979] [Norman, Gentner, & Stevens, 1976]. This new direction is based on attempts to make a bug "constructive" by

establishing for the student that there is something inadequate in his approach, and giving enough information so that the student can use what he already knows to focus on the bug and characterize it so that he avoids this failing in the future.

However, it is by no means clear how "just the right thing" is to be said to the student. We do know that it depends on having a very good model of his understanding process (the methods and strategies he used to construct a solution). Current research is focusing on means for representing and isolating the bugs themselves [Stevens, Collins, & Goldin, 1978] [Brown & Burton, 1978].

Another approach is to provide an environment that encourages the student to think in terms of debugging his own knowledge. In one BIP experiment [Wescourt, et al., 1978], explicit debugging strategies (for computer programming) were conveyed in a written document and then a controlled experiment was undertaken to see whether this fostered a more rational approach for detecting faulty use of (programming) skills.

Brown, Collins and Harris [Brown, 1977b] suggest that one might foster the ability to construct hypotheses and test them (the basis of understanding in their model) by setting up problems in which the student's first guess is likely to be wrong, thus "requiring him to focus on how he detects that his guess is wrong and how he then intelligently goes about revising it."

The Socratic method used in WHY [Collins, 1976] involves questioning the student in a way that will encourage him to reason about what he knows, and so modify his conceptions. The tutor's strategies are constructed by analyzing protocols of real-world student/teacher interactions.

Another teaching strategy that has been successfully implemented on several systems is called *coaching* [Goldstein, 1977]. Coaching programs are not designed to cover a predetermined lesson plan within a fixed time (in contrast with SCHOLAR). Rather, the goal of coaching is to develop the acquisition of skill and general problem solving abilities by by engaging the student in a game and unobtrusively making suggestions for improving his play. In a coaching situation, the immediate aim of the student is to have fun, and skill acquisition is an indirect consequence. WUMPUS and WEST are both coaching programs.

Socratic tutoring and coaching represent different styles for communicating with the student. All mixed-initiative tutoring involves following some dialogue strategy. This will include decisions about when and how often to question the student, and methods for presentation of new material and review. For example, by design, a coaching program is not intrusive, and only *rarely* lectures. On the other hand, a Socratic tutor questions repetitively, requiring the student to pursue certain lines of reasoning. Recent ICAI research has turned to making explicit these alternative dialogue principles. Collins has pioneered the careful articulation and investigation of teaching strategies, using production rules as a stylistic convention for describing the strategies used by his program.

In general, ICAI programs have only begun to deal with the problems of representing and acquiring teaching expertise and of determining how this knowledge should be integrated with general principles of discourse.

2.3 Medical CAI

The literature for medical CAI systems is extensive. [Hoffer, et al., 1975] provides a good overview. Not all of the programs reported can be characterized as frame-oriented; some programs use probability tables to generate "cases" (a patient with a specific problem), and use differential diagnosis to analyze the student's response and provide assistance [Entwisle & Entwisle, 1963][Steele, et al., 1978]. Most researchers address the following set of issues: realism of the case, nature of the dialogue, feedback and pedagogy, and the problem of assembling a variety of cases. The successful use of the case method in these programs [Harless, et al., 1971] [Swets & Feurzeig, 1965] [Weber & Hagamen, 1972] has most directly influenced the design of GUIDON.

Chapter 3

Annotated Protocol

CONTENTS

3.1 The Experimental Setting

3.2 Program Features to Look for

3.3 The Protocol

3.4 The Student's Reaction to the Program
- 3.4.1 Definition of the Problem
- 3.4.2 Strategy
- 3.4.3 Interaction with the Program

This chapter includes the entire typescript of a GUIDON tutorial session. The tutorial shown here is used to illustrate the program's use of domain knowledge, tutoring rules, student initiative, and modeling discussed in later chapters. These issues are brought out in the typescript by annotations signified by braces ({}); the student's responses to GUIDON are in BOLDFACE and follow the double asterisks (**). Later references to this typescript will usually mention "the protocol" and a page number. The item of interest on a referenced page is indicated by the pointing hand (☜). A good way to become familiar with the GUIDON's features is to scan the protocol, reading these comments. Other sections of this chapter describe how the typescript was produced and a summary of the student's (verbal) comments during the tutorial.

3.1 The Experimental Setting

The purpose of the experiment was to produce a typescript that illustrated most of the program's capabilities, specifically for inclusion in this dissertation. It was deemed important that the student have an adequate background to actually work out a case posed by GUIDON. This allows us to directly observe how the student's approach meshes with the program's initiative and available options.

The student who used GUIDON in this tutorial was a second-year medical student. He had no previous experience with this program or MYCIN's rules, though he had an undergraduate background in computer science. Consequently, the student was familiar with the terminal and the mechanics of interacting with a computer

program, so he was able to focus directly on the problem-solving task itself. The author monitored his use of the program, sometimes helping him to translate his questions into program options. A large cardboard diagram that summarized the 30 options available to the student was always on display. In addition, the student made frequent reference to the GUIDON handbook for MYCIN (Appendix C) which he read the night before.

The session took nearly 4 hours because the student was encouraged to verbalize his thoughts throughout. The author and student also spent some time discussing pros and cons of the program's remarks (these are summarized in Section 3.4). The reader will observe that the student first exhibits uncertainty about what to do, shifts to an aggressive attack on the problem, and finally runs out of leads, too tired to explore MYCIN's solution in much detail.

The typescript shown here was produced by correcting several program bugs [1] and then rerunning the tutorial, repeating the student's original input exactly. There is one exception: since the student answered all of the program's quizzes correctly, one answer was modified to prove that GUIDON could actually deal with an incorrect response (page 62).

Finally, the phrase "organisms (other than those seen on cultures or smears) which might be causing the infection" has been replaced by "organisms that might be

[1] For example, a new constraint was added to the t-rules for selecting quiz types to prevent a nonsensical question, the control-O option for aborting output was fixed, and other similar, local changes were made that did not change the basic flow of the dialogue. No t-rules for modeling the student or guiding the dialogue were modified.

causing the infection." This is one of MYCIN's fixed-text parameter descriptions that a tutorial system may need to word in different ways in different contexts.

3.2 Program Features to Look For

The current version of GUIDON can be characterized as an unobtrusive reasoning assistant. Through a wide variety of simple options, the program provides easy access to a large knowledge base, helping the student to measure the adequacy of his understanding of the problem. For example, observe how the "data retrieval" options USE, CONCLUDE, FACTORS, PENDING (protocol page 74, page 68, page 65, page 72) and statements of some of MYCIN's conclusions (preceded by ">>>") indirectly challenge the student to address problem subtasks.

The student usually controls the solution of the problem: he determines when a topic is changed and the order in which case data are considered. Specifically, the DISCUSS option changes the topic with later return to the current topic. The TELLME option effectively terminates discussion of a topic. These features demonstrate how GUIDON provides for and copes with student initiative, the subject of Chapter 7.

GUIDON takes the initiative by offering orientation to new topics at several times. It presents:

1) the "goal rule" of the task (see Appendix D)

2) a sketch of the entire subgoal tree (Appendix D)

3) a "model" of the typical factors to consider for a "major" goal (page 47)

4) FACTORS to consider, the first time help is requested for a goal (page 49)

5) introduction of a rule in terms of the evidence it will contribute to what is already known (page 68).

Tutorial rules explicitly control when this information is offered. For example, default heuristics now limit presentation of the goal rule to a new user of the program. In addition, following a method we call *opportunistic tutoring*, GUIDON interrupts the student to question him about what he is doing (page 67) and to present information that broadens his knowledge beyond the needs of the case he is solving (page 77). Tutoring rules and principles for guiding the dialogue in this way are considered in Section 6.2.3.3.

Finally, the program maintains a model of what the student knows, as well as some limited information about topics in which he is interested. Presentation methods are tuned to this model, as evidenced by a statement of a result rather than a quiz about it (page 76) and response to a student hypothesis (page 56) that reflects the tutor's measure of the student's understanding. The contents, maintenance and use of the student model are considered in Chapter 8.

3.3 The Protocol

March 25, 1979 12:57PM in <CLANCEY>GUIDON.SAV;90324 by CLANCEY

This is the tutor for MYCIN.

Instructions (Y or N)?
** YES

Brief Description of the Tutorial Session

You are to play the role of consultant for a patient suspected to have an infection. After initial data has been presented, you should request more information about the patient, decide which organisms (if any) require therapy, and prescribe treatment. The tutor will direct you, according to the profile you set. Its behavior can vary from leaving you alone until you are ready to make a diagnosis, to directing your thinking at every step. Feel free to change your profile or request HELP to solve the problem at any time. Use a question mark (?) whenever you want to see the list of options open to you. Remember that the purpose of this session is to present MYCIN's rules of infectious disease diagnosis and, in the second part, the criteria of therapy selection.

Type control-G to call GUIDON. You will receive the initiative after the next question is asked.

Please enter your initials:
** STU

Well, STU, it looks like we don't have a file on you....
Please enter your name and address:
(terminate with control-Z)
** ↑Z

-1 Session Profile

{The profile is used to initialize the student model for a new student and control the tutor's remarks about MYCIN's reasoning. Options include printing nothing, a hint (or general statement about a result), or the actual conclusion made by the program.}

The following questions concern the SESSION PROFILE.

Do you want instructions for using the profile?
** NO

Standard setting? (type ? for help)
** NO
On a scale from 0 to 4, how do you rate your problem-solving experience in this field [0=naive (no training); 1=some experience (beginning student); 2=considerable experience (advanced student); 3=practitioner; 4=specialist]?
Suggested setting: 1
** 1
Should MYCIN tell you when a rule succeeds?
Suggested setting: NO
** NO
Should MYCIN tell you when a rule fails?
Suggested setting: NO
** NO
Which rule conclusions should MYCIN mention during the session?
Suggested setting: HINTONLY
** HINTONLY
What should MYCIN tell you when a final conclusion for a factor is known?
Suggested setting: HINTONLY
** HINTONLY
Should the default lesson plan be used?
Suggested setting: YES
** YES
Should DEBUGGING information be printed?
Suggested setting: NO
** NO
okay

{Next, files containing case data, an AND/OR tree representation of MYCIN's solution, and records from the consultation are loaded. After a few minutes wait, a basic outline of the case is given to the student. This is done using MYCIN's list of "main properties"--identifying data it always collects about an object, e.g., the site of a culture, the name, age, and sex of a patient. In this case there is one culture that has been sent to the labs, and one drug that the patient is still receiving; an infection is suspected. The format for displaying initial data is simple in this first version of GUIDON. Later development will need to take into account methods for more concise presentation, perhaps according to domain-specific conventions.}

Tutorial Session 26-Mar-79 12:59:18

Which case?
** 538
Loading <CLANCEY>PATIENT.538;5...
Loading TUTORIAL.538...
File created 5-Feb-79 18:49:34
Loading DIAGNOSIS.538...
File created 5-Feb-79 18:50:01

[consultation of 30-Jan-77 10:15PM]

Initial Case Data

PATIENT-538:
The name of PATIENT-538 is J.Smith.
The age of J.Smith is 34 years.
The sex of J.Smith is male.
The race of J.Smith is Caucasian.

There was 1 pending cultures of J.Smith, as follows...
CULTURE-1:
The site of CULTURE-1 is csf.
The time since the specimen for the pending csf culture was obtained is 10.2 hours.
A smear of the pending csf culture was examined.

There was 1 current drugs of J.Smith, as follows...
DRUG-1:
The name of DRUG-1 is cephalothin.
The route of administration for the cephalothin is iv.
The time since therapy with the cephalothin was started is 1.06 weeks.

There was 1 suspected infections of J.Smith, as follows...
INFECTION-1: [No initial data will be presented]

{Using the ANDIOR tree and initial case data, GUIDON fires domain rules in a forward direction. Directed by the session profile, GUIDON mentions some of MYCIN's conclusions (2a-4a) (">>>" indicates such a remark throughout this protocol). However, note that simplification rules have reduced the number of statements, such as omitting comments about intermediate conclusions when there is only one rule that succeeds. Comment 4a is an artifact of MYCIN's representation: "organisms of the patient" is the set union of organisms associated with each culture.}

[Now giving initial case data to MYCIN... this may take a few minutes.]

>>> 2a. MYCIN has made a FINAL CONCLUSION about organisms noted on smears of the pending csf culture.

>>> 3a. MYCIN has made a FINAL CONCLUSION about whether organisms were seen on the stain of the pending csf culture.

>>> 4a. MYCIN has made a FINAL CONCLUSION about organisms noted on smears of J.Smith.

[MYCIN is now considering additional questions to ask....]

{Each parameter in the ANDIOR tree is now scanned to determine if it is now known, or whether case data will need to be requested. 5a and 6a are known because they are not relevant to a case that doesn't have positive cultures.}

>>> 5a. MYCIN has made a FINAL CONCLUSION about the associated infections for which therapy should cover.

>>> 6a. MYCIN has made a FINAL CONCLUSION about the organisms which require therapy.

{The tutor has begun consideration of the top goal: determining a regimen of antibiotics for the patient. The student has never used the program before, so he is shown the goal rule and, next, a representation of the AND/OR tree that shows only "important" subgoals. This orientation has been omitted here and moved to Appendix D.}

We will now begin to discuss this case, limiting our discussion to one topic at a time.

{The tutor observes that this topic corresponds to the goal rule, and the student has never discussed it before, so it will be discussed clause-by-clause....}

(Now considering RULE092.)

We have already determined that information has been gathered about organisms isolated from J.Smith, organisms noted on smears of J.Smith, negative cultures of J.Smith, suspected infections without microbiological evidence, current drugs of J.Smith and prior drugs of J.Smith.

{The procedure for discussing a rule clause-by-clause indicates whether a clause is done or not. If not, the procedure for discussing a goal is invoked.}

We must now determine whether an attempt has been made to deduce the organisms which require therapy.
We are not able to make a conclusion about the organisms which require therapy, because there are no organisms in this case.

{One goal is to determine which organisms from positive cultures require therapy (some may be contaminants). The tutor notes that there is nothing to discuss here because we have no organisms to consider. This patient will be treated on the basis of clinical data (social history, age, etc.) alone.}

(Continuing our discussion of the therapeutic regimen of J.Smith [RULE092].)

We must now determine whether you have given consideration to organisms that might be present, and found at least one organism which requires therapy.

{The main topic of the tutorial has begun. A heuristic based on the number of rules associated with this topic indicates that it is important, so orientation for pursuing it is presented: a definition, list of important subgoals, indication of the extent of MYCIN's knowledge, and a list of the kinds of answers to look for.}

You should now proceed to ask questions that will enable you to make an hypothesis about the organisms that might be causing the infection.

A pathogenic organism which was not seen on a culture or smear but which might be causing an infection in the patient and therefore requires antimicrobial therapy is an organism to "cover for."

Important factors for a major topic

CULTURE-1 is a pending culture. In this context, when we are considering the organisms that might be causing the infection, we generally find it useful to consider:

8a. the infection which requires therapy
8b. the type of the infection
8c. whether organisms were seen on the stain of the culture
8d. whether a smear of the culture was examined

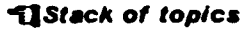
There are 93 rules used by MYCIN to conclude about this topic. Altogether there are 29 factors considered in these rules.

Some sample values for the organisms that might be causing the infection are: proteus-non-mirabilis, streptococcus-group-b, proteus-mirabilis and neisseria-meningitidis, etc.

{The student begins by asking for information about the ORIENT option. After using it, he asks to be quizzed. Options available to the student are described in Chapter 7 (see Figure 7.1 and Appendix C).}

** HELP ORIENT

The tutor recites all of the topics that are "on the stack." Starting with "discussing a case," it moves down through the topics and rules that have been mentioned, to explain how the dialogue got to its current point. Finally, if possible, it will indicate the NEXT steps that will be taken by the tutor. Equivalent to typing control-F.



{The history of active topics and tutoring rules is read to the student. T-rule1.04 is retrieved from the source file to describe what will be done next.}

** ORIENT

Discussing a case with the student.
(Applying tutorial rule 3.01).

-> Discussing the therapeutic regimen of J.Smith.
(Applying tutorial rule 1.04).
Discussing rule092 as it relates to finding out about the therapeutic regimen of J.Smith.

-> Discussing the organisms that might be causing the infection.

The remaining steps in tutorial rule 1.04 are:
Substep v. Say: goodbye
Substep vi. Record that the goal of the consultation has been mentioned in the dialogue
Substep vii. Exit this procedure

{No rules for this topic can be configured into one of the available question formats. No rules have succeeded yet; the rules that failed have incomplete subgoals.}

** QUIZ

Sorry, no quizable rules have been determined to fail or succeed, based on information available to you. Request HELP to discuss other rules.

First assistance is list of factors to consider

** HELP

The following factors will be useful to determine the organisms that might be causing the infection:

9a. whether the infection was acquired while the patient was hospitalized
9b. whether J.Smith has a head injury defect
9c. whether J.Smith has a tb risk factor
9d. whether the results of a recent CBC of J.Smith are available
9e. the white count from J.Smith's peripheral CBC (in thousands)
9f. whether J.Smith has been seriously burned
9g. whether J.Smith has sickle cell disease
9h. whether J.Smith has had neurosurgery
9i. whether J.Smith has had a severe head trauma
(*) 9j. organisms isolated from J.Smith
9k. the infection which requires therapy
9l. the type of the infection
9m. whether J.Smith is a compromised host
(*) 9n. the race of J.Smith
9o. whether J.Smith has evidence of serious skin or soft tissue infection
9p. whether organisms were seen on the stain of the pending csf culture
9q. whether J.Smith has evidence of ocular nerve dysfunction
9r. whether J.Smith has undergone recent radiation therapy to the CNS
(*) 9s. the age of J.Smith
9t. whether there is evidence of congenital malformation involving the central nervous system
9u. the diagnoses of J.Smith
9v. whether J.Smith's X-ray is abnormal
9w. whether J.Smith does live in a crowded environment
9x. whether J.Smith is an alcoholic
9y. the epidemic diseases to which J.Smith has been exposed recently
9z. the types of rash which J.Smith has
9aa. whether J.Smith has received corticosteroids just prior to or during this infection
9ab. the diagnoses which are consistent with J.Smith's clinical history
9ac. whether J.Smith has had a splenectomy

Factors marked with '*' have already been stated. Based on what we have already mentioned, you should be able to conclude about: 9p.

(This list is produced by the FACTORS option.)

{The first time help is requested for a topic, the tutor invokes the FACTORS option. Note that factor 9p is known to MYCIN. The student can make an hypothesis, say that he knows it, request a detailed discussion, or ask for the answer. Here he decides to offer an hypothesis about the current topic.}

** HYP
What are the organisms that might be causing the infection?
** NEISSERIA-MENINGITIS (5)
=NEISSERIA-MENINGITIDIS

{MYCIN has not made this conclusion, so the student is asked to support his (wrong) hypothesis.}

What facts about this case tell you that Neisseria-meningitidis is one of the organisms that be causing the infection (.5)?
(Enter one factor per line.)
** AGE
**

None of the factors that you mentioned are appropriate:

-> The age of J.Smith is not evidence for the conclusion that Neisseria-meningitidis is one of the organisms that might be causing the infection (.5).

Key factors and context

{The tutor points out that the age of the patient will eventually enable the student to hypothesize several possible organisms--it's just premature to do so now. Observe that only "key factors" of rules are printed (whether J.Smith has a head injury defect) and that the "context" of these conclusions (two rules) depends on the infection and its type. GUIDON has in fact given away too much information at this point.}

After gaining more information about this case, the age of J.Smith will enable us to conclude that the organisms that might be causing the infection are:
e.coli (considering whether the infection was acquired while the patient was hospitalized [RULE545]),
pseudomonas-aeruginosa (considering [RULE545]),
klebsiella-pneumoniae (considering [RULE545]),
diplococcus-pneumoniae (considering whether J.Smith has a head injury defect and the age of J.Smith [RULE507]),
neisseria-meningitidis (considering [RULE507]),
staphylococcus-coag-pos (considering [RULE545]).

Remember that both decisions are made given that: 1) The infection which requires therapy is meningitis, 2) Organisms were not seen on the stain of the pending csf culture and 3) The type of the infection is bacterial.

{After the hypothesis, a procedure that prepares for return to the data collection mode is followed. The student decides to explore the topic in more detail.}

Note: From case data we have not mentioned, you will find evidence for: diplococcus-pneumoniae, klebsiella-pneumoniae, pseudomonas-aeruginosa and e.coli.
(Use the CONCLUDE option for details.)

Would you like to end discussion of this topic now (i.e., get the chance to see a summary of all evidence that can eventually be collected instead of working it out yourself)?
** NO

{The student asks, "What rashes does this patient have?" The tutor observes that MYCIN never asked about specific rashes, and gives the student the more general information that kept MYCIN from asking.}

** RASH

J.Smith does not have a rash or cutaneous lesions.

{As before, MYCIN rules are fired as new information is given to the student. Thus a record is kept of hypotheses that are consistent with information the student has received.}

>>> 10a. MYCIN has made a FINAL CONCLUSION about whether J.Smith has evidence of serious skin or soft tissue infection.

>>> 11a. MYCIN has made a FINAL CONCLUSION about whether J.Smith has a rash consistent with Herpes Zoster.

>>> 12a. MYCIN has made a FINAL CONCLUSION about whether J.Smith has had recent vesicular eruptions.

>>> 13a. MYCIN has made a FINAL CONCLUSION about the types of rash which J.Smith has.

{The student wants to find out why MYCIN ultimately decided that the infection was caused by bacteria (page 51) topic. However, his first few questions (about neurosurgery, x-ray, and hospital-acquired infection) reflect continued interest in the previous topic, the organisms causing the infection. Finally, he inquires about about a relevant factor (CSF glucose), and gets enough information to conclude that the infection was caused by bacteria.}

** DISCUSS TYPE
You should now proceed to ask questions that will enable you to make an hypothesis about the type of the infection.

Definition of a topic

An infection is classified according to the type of organism causing the infection (e.g. Bacteria, Fungus, Virus, Tuberculosis).

{Orientation is presented for this important topic.}

CULTURE-1 is a pending culture. In this context, when we are considering the type of the infection, we generally find it useful to consider:

14a. the infection which requires therapy

There are 23 rules used by MYCIN to conclude about this topic. Altogether there are 20 factors considered in these rules.

The values for the type of the infection are: bacterial, fungal, viral and tb.

{The student asks, "Has the patient had neurosurgery?" (from page 49). But GUIDON doesn't want to discuss neurosurgery because it is a parameter that shouldn't be asked until rules (and other factors) are considered that might rule it out. The program discovers that only the "screening question" about surgery needs to be considered, so it gives this information to the student and proceeds to answer his original question.}

** 9H

J.Smith has undergone surgery.

It is not known whether J.Smith has had neurosurgery.

Student begins to jump around

{MYCIN's pattern-matching, keyword parser is used by GUIDON. In this case, it can't tell which parameter is being referred to.}

** HAS THE PATIENT HAD AN X-RAY?
Sorry, can't parse that.
The question appeared to use the DATA option, in reference to PATIENT-538.

You may find it easier to use the single command format.
Type ? for assistance.

Response to irrelevant question

{The student's question is related to the current topic.}

** 9V
Whether J.Smith's X-ray is abnormal is not useful for determining the type of the infection. It is a consideration we can use later when we return to our discussion of the organisms that might be causing the infection.

J.Smith's X-ray is not abnormal.
** 9A
Whether the infection was acquired while the patient was hospitalized is not useful for determining the type of the infection. It is a consideration we can use later when we return to our discussion of the organisms that might be causing the infection.

The infection was acquired while the patient was hospitalized.

Typical valid data request

** CSF GLUCOSE

The csf glucose value is 6.

The "CSF findings" are a set of measurements that are usually available after a lumbar puncture has been performed. They include: the CSF protein, the WBC from the CSF, the percent of PMN's in the CSF, and the CSF glucose value.

The CSF findings we have not yet mentioned:

The WBC from the CSF is 100.

{MYCIN rules are applied as new case data comes in. The tutor concludes here (silently) that the student doesn't know any of these "final conclusions" (16a-18a).}

>>> 15a. MYCIN just made a conclusion about the infection which requires therapy.

>>> 16a. MYCIN has made a FINAL CONCLUSION about the infection which requires therapy.

>>> 17a. MYCIN has made a FINAL CONCLUSION about whether the meningitis is partially treated.

>>> 18a. MYCIN just made a conclusion about the type of the infection.

{The csf glucose is one of a block of related test results, so the tutor gives all of them. By requesting details about any numbered remark (e.g.,19a) the student can find out what MYCIN's conclusions are.}

The percent of pmns in the csf is 99.

>>> 19a. MYCIN just made a conclusion about the type of the infection.

The CSF protein is 280.

>>> 20a. MYCIN just made a conclusion about the type of the infection.

The simultaneous blood glucose value is not known.

>>> 21a. MYCIN just made a conclusion about the type of the infection.

(Other factors in this block have already been given.)

{With all of this information, MYCIN's current hypothesis is "close" to the final answer it will deduce. To save time, the student accepts the offer to wrap up discussion.}

We have substantially completed our discussion of the type of the infection.
Would you like to make a hypothesis now (as opposed to gathering more information)?
** YES

Response to hypothesis

{The student makes an hypothesis and GUIDON responds.}

What is the type of the infection?
** BACTERIAL (9)

{The student's hypothesis is correct, but the model indicates that he may have missed some relevant evidence. Various quiz options are considered, but the default (t-rule25.05) was used: ask for relevant factors to support the hypothesis.}

What facts about this case tell you that the type of the infection is bacterial (.9)?
(Enter one factor per line.)
** CSFGLUC
** PMNS
The percent of PMN's in the CBC is not relevant for determining the type of the infection.
** CSFPOLY
** PROTEIN
**

Evidence summary

{Each factor is checked for relevance to the current topic, then the tutor compares the student's set of factors to those mentioned in all of the MYCIN rules that cutrently support the hypothesis.}

You're right about the CSF protein, the percent of PMNs in the CSF and the CSF glucose value, however you should have mentioned the fact that the

meningitis is partially treated (.95), the WBC from the CSF and the fact that the simultaneous blood glucose value is not known.

To summarize, we can conclude that the type of the infection is:
bacterial (considering:
whether the meningitis is partially treated [RULE148],
the WBC from the CSF and the percent of PMNs in the CSF [RULE501],
the CSF protein [RULE500] and
the CSF glucose value and the simultaneous blood glucose value [RULE502]).
Remember that these decisions are made given that the infection which requires therapy is meningitis.

MYCIN rejected some possibilities you did not mention:

{There is more than one rule that supports the rejected hypothesis (viral), so again the student is asked to list relevant factors (t-rule26.01).}

What facts about this case tell you that the type of the infection is not viral (.87)?
(Enter one factor per line.)
** CSFGLUC
** CSFPOLY
** PROTEIN
**

[1] *Missing factors*

{Observe that the student again forgets to mention the WBC from the CSF and the simultaneous glucose value. The student model is again modified to show that he doesn't know these rules (501 & 502).}

You're right about the CSF protein, the percent of PMNs in the CSF and the CSF glucose value, however you should have mentioned the WBC from the CSF and the fact that the simultaneous blood glucose value is not known.

To summarize, we can conclude that the type of the infection is:
not viral (considering:
the WBC from the CSF and the percent of PMNs in the CSF [RULE501],

the CSF protein [RULE500] and
the CSF glucose value and the simultaneous blood glucose value [RULE502]).
Remember that these decisions are made given that the infection which requires therapy is meningitis.

{The remaining rejected values are concluded by a single rule that, a priori, is considered to be more than one level above the student's sophistication, so these conclusions were stated directly (t-rule26.02). This estimation of the student's knowledge is borne out by the interaction above in which the student twice neglects to refer to rule501.}

The WBC from the CSF and the percent of PMNs in the CSF is evidence that the type of the infection is not tb (.7) [RULE501].
The WBC from the CSF and the percent of PMNs in the CSF is evidence that the type of the infection is not fungal (.7) [RULE501].

Detecting topic termination

It remains to collect further evidence for determining the type of the infection.
However, we have substantially completed our discussion of the type of the infection.
Would you like to end discussion of this topic now (i.e., get the chance to see a summary of all evidence that can eventually be collected instead of working it out yourself)?
** NO

Would you like to see a summary of the evidence we have discussed before going on to ask more questions?
** NO

{The topic continues to be the type of the infection. GUIDON considers quizzing about two rules that recently failed (but they are too advanced for the student), and four rules that recently succeeded (but they were just discussed above). Nothing is said.}

Indirectly useful question

{The student's question about exposure to contagious diseases implicitly brings into focus a deeper subgoal, the actual diseases to which the patient has been exposed. (If the patient had been exposed to mumps, one could conclude that the type of the infection is viral.) Rather than opening up discussion of the deeper subgoal, the tutor states the only relevant rule, one that is probably known by the student (rule463). Compare the dialogue here to what occurred when abnormal x-ray, an "irrelevant" question, was asked on page 54.}

** EXPOSED

J.Smith has not been recently exposed to a contagious disease.

>>> 22a. MYCIN has made a FINAL CONCLUSION about the diseases to which J.Smith has been exposed.

>>> 23a. MYCIN has made a FINAL CONCLUSION about the epidemic diseases to which J.Smith has been exposed recently.

>>> 24a. MYCIN just made a conclusion about the organisms (other than those seen on cultures or smears) which might be causing the infection.

Your question is indirectly useful for determining the type of the infection.

Whether J.Smith has been recently exposed to a contagious disease can be used to determine the diseases to which J.Smith has been exposed...

and this will enable us to determine the type of the infection.

The fact that J.Smith has not been recently exposed to a contagious disease is evidence that mumps is not one of the diseases to which J.Smith has been exposed [RULE463].

Back to our discussion of the type of the infection....

{The student now requests that this topic be terminated.}

** TELLME

Our final conclusion is that the type of the infection is bacterial (.98) viral (-.76) fungal (-.83) tb (-.83).
Would you like to see a summary of the evidence that supports this conclusion?
** YES

{MYCIN's final conclusion is not the same as the conclusion that can be made from the case data mentioned so far in the dialogue, so first GUIDON lists the missing relevant data.}

. . . . . .

Factors that have not been mentioned that are relevant to determining the type of the infection:

The white count from J.Smith's peripheral CBC (in thousands) is 1.9.

>>> 25a. MYCIN just made a conclusion about the type of the infection.

J.Smith has had recent neurologic signs.

The duration of the neurological signs is 7.2 hours.

>>> 26a. MYCIN just made a conclusion about the type of the infection.

J.Smith has had recent neurologic symptoms.

The duration of the neurological symptoms is 1 day.

>>> 27a. MYCIN just made a conclusion about the type of the infection.

It is not known whether lymphoma is one of the diagnoses of the patient.
It is not known whether leukemia is one of the diagnoses of the patient.

J.Smith has not shown symptoms of mumps.

J.Smith does not have a tb risk factor.

>>> 28a. MYCIN just made a conclusion about the organisms that might be causing the infection.

. . . . . .

{Rules that made conclusions about this topic are now described in terms of their key factors.}

Summary of evidence that the type of the infection is bacterial (.98) viral (-.76) fungal (-.83) tb (-.83):

29a. The fact that the meningitis is partially treated (.95) is evidence that the type of the infection is bacterial (.19) [RULE148]

29b. The WBC from the CSF and the percent of PMNs in the CSF is evidence that the type of the infection is bacterial (.9) viral (-.6) fungal (-.7) tb (-.7) [RULE501]

29c. The CSF protein is evidence that the type of the infection is bacterial (.4) viral (-.5) fungal (.4) tb (.4) [RULE500]

29d. The csf glucose value and the fact that the simultaneous blood glucose value is not known is evidence that the type of the infection is bacterial (.35) viral (-.35) [RULE502]

29e. The white count from J.Smith's peripheral CBC (in thousands) and the WBC from the CSF is evidence that the type of the infection is bacterial (.2) viral (.15) fungal (.1) tb (.1) [RULE504]

29f. The duration of the neurological signs is evidence that the type of the infection is bacterial (.2) viral (.2) fungal (-.4) tb (-.4) [RULE524]

29g. The duration of the neurological symptoms is evidence that the type of the infection is bacterial (.2) viral (.2) fungal (-.5) tb (-.5) [RULE526]

Remember that these decisions are made given that the infection which requires therapy is meningitis.

*** *** ***

[1] *Forced termination of topic*

{GUIDON now marks the inference and discussion record to show that this goal has been completed. MYCIN proceeds to use this final conclusion (31a). As always, the student model is updated according to whether GUIDON believes that the student is following along.}

>>> 30a. MYCIN has made a FINAL CONCLUSION about the type of the infection.

>>> 31a. MYCIN just made a conclusion about the organisms that might be causing the infection.

Back to our discussion of the organisms that might be causing the infection....

Quiz and response to incorrect answer

{After discussing the type of the infection, the topic pops back to determining the organisms that are causing the infection. Seven rules that have failed are now considered for a quiz. One has two incomplete subgoals; three are too advanced for this student. The remaining three rules have the same "interestingness" score, so the choice is random. The premise/action format of the question is "multiple choice premise"/"given correct conclusion." (Possible question formats are summarized in Appendix B.)

Given the infection which requires therapy, what value for the type of the infection would tell you that the organisms that might be causing the infection are: Cryptococcus (1.0) Coccidioides (1.0)?
(select from the following):

1) VIRAL
2) FUNGAL
3) TB
4) BACTERIAL

** 3

No, you could make this conclusion if the type of the infection were fungal [RULE666].

Student's attempt to follow MYCIN's reasoning

{The student now requests more information about an inference about the previous topic. He will then examine how MYCIN determined a subgoal of this inference (that the infection is meningitis): he says that he knows it, prints the rule that MYCIN used, asks for a justification for the rule, requests to discuss the subgoal, prints relevant factors for the subgoal, and examines a rule that didn't apply to this case.}

** DETAILS 29A
For which of the following factors would you like to see a review of the evidence?
(select from the following):
1) the infection which requires therapy
2) the fact that the meningitis is partially treated (.95)
** 2

{GUIDON observes that this rule which it mentioned earlier requires two subgoals that were not discussed. Consistent with the student model, the student did not know how to determine "partial treatment."}

Summary of evidence that the meningitis is partially treated (.95):

32a. The infection which requires therapy, the fact that organisms were not seen on the stain of the pending csf culture (CULTURE-1) and the time since therapy with the cephalothin (DRUG-1) was started is evidence that the meningitis is partially treated (.95) [RULE145]

Do you want to see RULE148?
** NO

{The student tells GUIDON that he doesn't need to discuss the infection that requires therapy.}

** IKNOW 8A
Good. MYCIN has also made a final decision.

The infection with which the pending csf culture is associated is evidence that the infection which requires therapy is meningitis [RULE333].

Back to our discussion of the organisms that might be causing the infection....

{Rule333 is a complicated way of saying that the name of the infection derives from the body location that is infected.}

** PR 333

RULE333

[This rule applies to cultures, and is tried in order to find out about the infection which requires therapy]

If: 1) The infection which requires therapy is not known,
2) The infection with which the culture is associated is known, and
3) There is no evidence that the infection which requires therapy is not the infection with which the culture is associated

Then: It is definite (1.0) that the infection which requires therapy is the infection with which the culture is associated

Author: Scott & Shortliffe

** JUSTIFY 333

[No info]

1 *Discussing a completed topic*

{The student decides to see if the tutor has anything further to say about the infection that requires therapy.}

** DISCUSS 8A

We have already completed our discussion of the infection which requires therapy.

Our final conclusion is that the infection which requires therapy is meningitis.
Would you like to see a summary of the evidence that supports this conclusion?

** YES

{MYCIN only used one rule.}

Summary of evidence that the infection which requires therapy is meningitis:

33a. The infection with which the pending csf culture is associated is evidence that the infection which requires therapy is meningitis [RULE333]

* * * * * * * * *

Back to our discussion of the organisms that might be causing the infection....

Displaying relevant factors

{Still not satisfied, the student asks to see all of the factors for concluding the infection which requires therapy.}

** FACTORS 8A

The following factors were useful to determine the infection which requires therapy:

(*) 34a. whether J.Smith has had recent neurologic symptoms
(*) 34b. whether J.Smith has had recent neurologic signs
(*) 34c. the WBC from the CSF
(*) 34d. the CSF protein
(*) 34e. the site of CULTURE-1
34f. the infection with which the pending csf culture (CULTURE-1) is associated

Factors marked with '*' have already been stated. Based on what we have already mentioned, you should be able to conclude about: 34f.

{Probing further to understand clause 3 of rule333, the student asks for details for one of the factors. He is told how this factor might have changed the outcome of the case.}

** DETAILS 34C

If J.Smith had not had recent neurologic symptoms, J.Smith had not had recent neurologic signs, the WBC from the CSF were less than 10 and the CSF protein were less than 40, one could conclude that the infection which requires

therapy is not meningitis (.8) [RULE116].

If J.Smith had not had recent neurologic signs and the WBC from the CSF were less than 5, one could conclude that the infection which requires therapy is not meningitis (.7) [RULE118].

{Having explored a topic that GUIDON skipped over (determine the name of the infection), the student reorients himself to the current goal.}

Statement of current topic

** TOPIC
We are discussing the organisms that might be causing the infection.
That is, you should be considering evidence for one or more of the following: coccidioides, bacteroides-fragilis, hemophilus-influenzae and streptococcus-species, etc.

Student begins to use handbook

{At this point, the student began to go down the list of factors that appeared in the hardcopy handbook for MYCIN. He asks for nine pieces of data in this series.}

** GRAM
There are no organisms in this case, so "the stain of the organism" is not meaningful.
** BURNED

J.Smith has been seriously burned.

>>> 35a. MYCIN just made a conclusion about the organisms that might be causing the infection.

Opportunistic tutoring interruption

{Here an "opportunistic tutoring" quiz uses the format of "actual premise facts"/"wrong conclusion." The conclusion is chosen randomly from other MYCIN rules.}

Given the fact that J.Smith has been seriously burned, does this tell you that Staphylococcus-coag-pos is one of the organisms that might be causing the infection (.74)?
** NO

That's right (see RULE578).

Given these facts, we can conclude that Pseudomonas-aeruginosa is one of the organisms that might be causing the infection (.49) [RULE578].

** NOSOCOMIAL

You were already given this information...

The infection was acquired while the patient was hospitalized.

Hypothesis revision hint

** HINT

Hint: Look for strong, positive evidence that Diplococcus-pneumoniae is one of the organisms that might be causing the infection.

{Applying an hypothesis revision strategy to order the rules that remain to be considered, GUIDON suggests that the student consider Diplococcus. However, the student decides to look more closely at another hypothesis....}

** CONCLUDE PSEUDOMONOS-ARIGINOSA

I don't recognize PSEUDOMONOS-ARIGINOSA as a value for the organisms that might be causing the infection.

(Type control-O to abort printout)

Choose from ACINETOBACTER-ANITRATUS, ACINETOBACTER-LWOFFI, ACINETOBACTER-SPECIES, ACTINOMYCES-ISRAELII, ACTINOMYCES-ODONTOLYTICUS, ACTINOMYCES-SPECIES:O

Factors relevant to an hypothesis

{Describing how an hypothesis can be supported, GUIDON mentions future conclusions, evidence that has not been discussed, and offers to list discussed evidence.}

** CONCLUDE PSEUDOMONAS-AERUGINOSA

Using the following factors, it will be possible to conclude that Pseudomonas-aeruginosa is one of the organisms that might be causing the infection:

36a. the white count from J.Smith's peripheral CBC (in thousands)

[RULE557]

- - - - - - - - - - -

Using the following factors, it is possible to conclude that Pseudomonas-aeruginosa is one of the organisms that might be causing the infection:

37a. the fact that the infection was acquired while the patient was hospitalized [RULE545]

- - - - - - - - - - -

Do you want information about evidence that we have already discussed (if any)?
** NO

{Returning to the list of factors in the handbook, the student is now looking for evidence of Diplococcus. The current version of GUIDON does not monitor the student's strategies.}

** CSRSUG

You were already given this information...

J.Smith's X-ray is not abnormal.

Premature question

{Adhering strictly to MYCIN's rules, the tutor refuses to give information about alcoholism until the student has considered whether the patient is a compromised host, a more general condition. This tutoring rule needs to be modified.}

** ALCOHOLIC
While you will eventually find it useful to ask about whether J.Smith is an alcoholic, it is premature to do so now. There is still a possibility that other data will enable you to deduce a value.
** CROWD

J.Smith does not live in a crowded environment.

If J.Smith did live in a crowded environment, would this tell you that Virus is one of the organisms that might be causing the infection (.89)?
** NO
That's right (see RULE533).

If J.Smith did live in a crowded environment, one could conclude that Neisseria-meningitidis is one of the organisms that might be causing the infection (.2) [RULE533].

{The student asks, "Does the patient have a fever?"}

** FEBRILE
MYCIN never needed to inquire about whether J.Smith is febrile.
Whether J.Smith is febrile is not relevant to our discussion of the organisms that might be causing the infection.

{The student now asks about a splenectomy because it is evidence for diplococcus. The tutor's question about a splenectomy reminds the student that sickle cell disease was also evidence for diplococcus, and he types this in his excitement to find out about sickle cell disease. Then, still anxious to find out about sickle cell disease, he mistakes the summary of rule539 to be another quiz, and types "Y" in agreement with the conclusion.}

** SPLENECTOMY

J.Smith has not had a splenectomy.

If J.Smith had had a splenectomy, would this tell you that Streptococcus-group-b is one of the organisms that might be causing the infection (.69)?
** SICKLECELL
Yes, No, or Dontknow, please.
** NO
That's right (see RULE559).

If J.Smith had had a splenectomy, one could conclude that Diplococcus-pneumoniae is one of the organisms that might be causing the infection (.3) [RULE559].

** Y
Can't make sense of: Y.
Enter PARAMETER/CONTEXT pair or index, e.g., 14a.

{GUIDON observes that MYCIN never asked about sickle cell disease, and states the rule (281) that gives the answer, a rule which the student almost certainly knows.}

** SICKLECELL
The race of J.Smith is evidence that J.Smith does not have sickle cell disease [RULE281].

{A question is posed using the format is "actual facts"/"correct conclusion."}

Given the fact that J.Smith does not have sickle cell disease, does this tell you that Diplococcus-pneumoniae is one of the organisms that might be causing the infection (.75)?
** YES
That's right (see RULE531).

Remaining factors

{The student stops to see what remains to be done. This leads him to request more data and to investigate why certain factors are relevant.}

** PENDING

We need to discuss the following subgoals before our discussion of the organisms that might be causing the infection will be complete:

38a. whether J.Smith has received corticosteroids just prior to or during this infection
38b. whether J.Smith is an alcoholic
38c. the diagnoses of J.Smith
38d. whether there is evidence of congenital malformation involving the central nervous system
38e. whether J.Smith has undergone recent radiation therapy to the CNS
38f. whether J.Smith has had a severe head trauma
38g. whether J.Smith has a head injury defect

We need to discuss the following case data that is directly relevant to the organisms that might be causing the infection:

39a. the diagnoses which are consistent with J.Smith's clinical history
39b. whether J.Smith has evidence of ocular nerve dysfunction
39c. whether J.Smith is a compromised host
39d. whether the results of a recent CBC of J.Smith are available

You should be able to make an hypothesis about the following subgoals:

40a. the types of rash which J.Smith has
40b. the epidemic diseases to which J.Smith has been exposed recently
40c. whether organisms were seen on the stain of the pending csf culture
40d. whether J.Smith has evidence of serious skin or soft tissue infection

{The tutor's reasoning for presenting information about a skin infection is the same as for sickle cell disease....}

** SKININFECT
The fact that J.Smith does not have a rash or cutaneous lesions is evidence that J.Smith does not have evidence of serious skin or soft tissue infection [RULE369].

{No attempt has been made to document all of MYCIN's parameters at this time....}

** DEFINE DIAGNOSES
Sorry, there is no entry for DIAGNOSES in MYCIN's definition file.

{The student asks to discuss the diagnoses that are consistent with the patient's clinical history. There are no rules to discuss, so GUIDON gives the case data and moves on.}

** DISCUSS 39A
The diagnoses which are consistent with J.Smith's clinical history is information about a case that is usually available to the consultant.

Epiglottitis is not one of the diagnoses which are consistent with the patient's clinical history.
Otitis-media is not one of the diagnoses which are consistent with the patient's clinical history.

>>> 41a. MYCIN just made a conclusion about the organisms that might be causing the infection.

Back to our discussion of the organisms that might be causing the infection....

{This quiz format is "wrong premise hypothesis"/"correct conclusion"....} information about a case that is usually available to the consultant.

Given the infection which requires therapy, if coccidioidomycosis were one of the diagnoses which are consistent with J.Smith's clinical history, would this tell you that Virus is one of the organisms that might be causing the infection (.2)?
** NO
That's right (see RULE580).

You could make this conclusion if epiglottitis were one of the diagnoses which are consistent with J.Smith's clinical history [RULE580].

** PR 580

RULE580

[This rule applies to all cultures and suspected infections, and is tried in order to find out about the organisms that might be causing the infection]

If: 1) The infection which requires therapy is meningitis,
2) Epiglottitis is one of the diagnoses which are consistent with the patient's clinical history
Then: There is weakly suggestive evidence (.2) that virus is one of the organisms that might be causing the infection.
Author: Yu

** 39B

J.Smith does not have evidence of ocular nerve dysfunction.

{At this point, GUIDON decides not to quiz about a rule that just failed (366) because it is too advanced for the student.}

** USE 39B

If J.Smith had evidence of ocular nerve dysfunction, one could conclude that Mycobacterium-tb is one of the organisms that might be causing the infection (.2) [RULE366].

Why a factor is relevant

{When explaining how a factor is used, GUIDON mentions only inferences relevant to the current topic. For example, the association between compromised host and alcoholism is not mentioned here.}

** USE 39C

J.Smith is a compromised host.

If the type of the infection were fungal, J.Smith were not a compromised host and the race of J.Smith were one of: black asian indian, one could conclude that Cryptococcus is not one of the organisms that might be causing the infection (.5) (other information would also need to be available to make this conclusion)[RULE280].
** USE 39D

The results of a recent CBC of J.Smith are available.

>>> 42a. MYCIN just made a conclusion about the organisms that might be causing the infection.

{As rules like 557 are presented, the student model is updated to show that the student, like MYCIN, may use the stated conclusions in other rules that he knows.}

Using other information available to you now, the fact that the results of a recent CBC of J.Smith are available enables you to conclude that the organisms that might be causing the infection are: Pseudomonas-aeruginosa (.49) E.coli (.74) Klebsiella-pneumoniae (.49) [RULE557].
** FOLEY
MYCIN never needed to inquire about whether J.Smith has an indwelling urinary catheter.
Whether J.Smith has an indwelling urinary catheter is not relevant to our discussion of the type of the infection.

Economical presentation

{The discussion that follows is a bit complex. The student asks for help and rule507 is chosen, following up on the hint ("look for evidence of Diplococcus-Pneumoniae") that the program made earlier (page 68). One subgoal remains to be done (head injury); the presentation method (t-rule31.04) is to tell the student to work on this subgoal. However, GUIDON observes (belatedly!) that this subgoal can be deduced from a rule that student probably knows (rule509), so it states the conclusion, rather than giving him the initiative....}

** HELP

Try to determine whether J.Smith has a head injury defect.

The fact that J.Smith has not had an injury or insult to, or defect in the CNS is evidence that J.Smith does not have a head injury defect [RULE509].

>>> 43a. MYCIN has made a FINAL CONCLUSION about whether J.Smith has had a severe head trauma.

>>> 44a. MYCIN has made a FINAL CONCLUSION about whether there is evidence of congenital malformation involving the central nervous system.

>>> 45a. MYCIN has made a FINAL CONCLUSION about whether J.Smith has a head injury defect.

>>> 46a. MYCIN just made a conclusion about the organisms that might be causing the infection.

Wrapping up discussion

{The procedure for wrapping up discussion of a rule is now followed: quiz about or state the conclusion, summarize cumulative evidence, report on the status of the current goal, discuss related rules, and return....}

We can now conclude that the organisms that might be causing the infection are: Diplococcus-pneumoniae (.74) Neisseria-meningitidis (.73) (considering whether J.Smith has a head injury defect and the age of J.Smith [RULE507]).

The evidence we have discussed so far indicates that the organisms that might be causing the infection are: E.coli (.93) Pseudomonas-aeruginosa (.82) Klebsiella-pneumoniae (.74) Diplococcus-pneumoniae (.74)

Neisseria-meningitidis (.45) Staphylococcus-coag-pos (.29) Mycobacterium-tb (-.4).

It remains for us to consider other factors for determining the organisms that might be causing the infection.

Opportunistic tutoring interruption

{"Related rules" mention one or more of the hypotheses concluded by the rule that was just discussed.}

>>
>>>>>>>>>>> Considering related rules >>>>>>>>>>>
>>>

{This quiz format is "actual facts"/"confirm related evidence." Strictly speaking, only Neisseria is confirmed.}

Does information about the types of rash which J.Smith has tend to confirm the conclusion that the organisms that might be causing the infection are: Diplococcus-pneumoniae (.74) Neisseria-meningitidis (.73) [RULE507]?
** NO
That's right (see RULE539).

{This quiz format (for a rule that fails) is "right hypothesis"/"correct conclusion."}

If meningococcal-disease were one of the epidemic diseases to which J.Smith has been exposed recently, would this tell you that Neisseria-meningitidis is one of the organisms that might be causing the infection (.8)?
** YES
That's right (see RULE537).

{This quiz format (for a rule that fails) is "right premise hypothesis"/"multiple choice conclusion."}

If otitis-media were one of the diagnoses which are consistent with J.Smith's

clinical history, one could conclude that the organisms that might be causing the infection are
(select from the following):

1) staphylococcus-coag-pos -- weakly suggestive evidence (.29)
2) diplococcus-pneumoniae -- suggestive evidence (.7)
3) pseudomonas-aeruginosa -- strongly suggestive evidence (.82)
4) e.coli -- strongly suggestive evidence (.93)
5) streptococcus-species -- weakly suggestive evidence (.09)
6) cryptococcus -- weakly suggestive evidence (.22)
7) not peptostreptococcus -- weakly suggestive evidence (-.23)
8) neisseria-meningitidis -- weakly suggestive evidence (.45)
9) listeria -- strongly suggestive evidence (.82)
10) not mycobacterium-tb -- weakly suggestive evidence (-.4)
11) klebsiella-pneumoniae -- suggestive evidence (.74)
12) hemophilus-influenzae -- suggestive evidence (.75)

** 2 12

Yes, that's right (see RULE549).

>>>

{The student returns to determining the organisms that might be causing the infection.}

** CONCLUDE E.COLI
All evidence for this conclusion has already been discussed.

Do you want information about evidence that we have already discussed (if any)?
** YES
Using the following factors, we have already discussed how to conclude that E.coli is one of the organisms that might be causing the infection:

47a. the white count from J.Smith's peripheral CBC (in thousands) [RULE557]

47b. the fact that the infection was acquired while the patient was hospitalized [RULE545]

** PR 557

RULE557

[This rule applies to pending cultures and suspected infections, and is tried in order to find out about the organisms that might be causing the infection]

If: 1) The infection which requires therapy is meningitis,
2) A: A smear of the culture was not examined, or
B: Organisms were not seen on the stain of the culture,
3) The type of the infection is bacterial,
4) The results of a recent CBC of the patient are available, and
5) The white count from the patient's peripheral CBC (in thousands) is less than 2.6
Then: There is evidence that the organisms that might be causing the infection is pseudomonas-aeruginosa (.5) e.coli (.75) klebsiella-pneumoniae (.5)
Author: Yu

{Now asked to "explain" this rule, GUIDON can account for the contextual clauses (1-3) and the screening clause (4), but it doesn't know how to fit WBC into its model of sufficient evidence for organisms causing an infection. It doesn't know that this white blood count range is termed "leukopenia" and that such a condition is nonspecific evidence of an infection.}

** EXPLAIN 557

Can't explain the clause:
the white count from J.Smith's peripheral CBC (in thousands) is less than 2.5

{Here the program restates the student's question because the parse is ambiguous. The reply shows that GUIDON, like any good human tutor, has a sense of humor.}

** WHAT IS THE PATIENT'S WHITE COUNT?
[I.e., you are asking about the race of J.Smith.]
You were already given this information...

The race of J.Smith is Caucasian.
** WHAT IS THE PATIENT'S WBC?
You were already given this information...

The white count from J.Smith's peripheral CBC (in thousands) is 1.9.

{The student is tired and decides to call it quits.}

** TELLME
Our final conclusion is that the organisms that might be causing the infection are: E.coli (.93) Pseudomonas-aeruginosa (.82) Klebsiella-pneumoniae (.74) Diplococcus-pneumoniae (.74) Neisseria-meningitidis (.45) Staphylococcus-coag-pos (.29) Mycobacterium-tb (-.4) Hemophilus-influenzae (-.4).
Would you like to see a summary of the evidence that supports this conclusion?
** YES

. . . . . .

Factors that have not been mentioned that are relevant to determining the organisms that might be causing the infection:

J.Smith is not an alcoholic.

. . . . . .

Summary of evidence that the organisms that might be causing the infection are: E.coli (.93) Pseudomonas-aeruginosa (.82) Klebsiella-pneumoniae (.74) Diplococcus-pneumoniae (.74) Neisseria-meningitidis (.45) Staphylococcus-coag-pos (.29) Mycobacterium-tb (-.4) Hemophilus-influenzae (-.4):

48a. The infection which requires therapy, the fact that the infection was acquired while the patient was hospitalized and information about the epidemic diseases to which J.Smith has been exposed recently is evidence that Neisseria-meningitidis is not one of the organisms that might be causing the infection (.5) [RULE561]

48b. The infection which requires therapy, the fact that J.Smith's X-ray is not abnormal and the fact that J.Smith does not have a tb risk factor is evidence that Mycobacterium-tb is not one of the organisms that might be causing the infection (.4) [RULE160]

48c. The fact that the infection was acquired while the patient was hospitalized is evidence that the organisms that might be causing the infection are: E.coli (.74) Staphylococcus-coag-pos (.29) Pseudomonas-aeruginosa (.29) Klebsiella-pneumoniae (.49) [RULE545]

48d. The fact that J.Smith has been seriously burned is evidence that Pseudomonas-aeruginosa is one of the organisms that might be causing the infection (.49) [RULE578]

48e. The infection which requires therapy, the fact that the infection was acquired while the patient was hospitalized and information about the diagnoses which are consistent with J.Smith's clinical history is evidence that Hemophilus-influenzae is not one of the organisms that might be causing the infection (.4) [RULE562]

48f. The white count from J.Smith's peripheral CBC (in thousands) is evidence that the organisms that might be causing the infection are: Pseudomonas-aeruginosa (.49) E.coli (.74) Klebsiella-pneumoniae (.49) [RULE557]

48g. The fact that J.Smith does not have a head injury defect and the age of J.Smith is evidence that the organisms that might be causing the infection are: Diplococcus-pneumoniae (.74) Neisseria-meningitidis (.73) [RULE507]

Remember that these decisions (RULE507, RULE557, RULE578 and RULE545) are made given that: 1) The infection which requires therapy is meningitis, 2) Organisms were not seen on the stain of the pending csf culture, and 3) The type of the infection is bacterial.

*** *** ***

>>> 49a. MYCIN has made a FINAL CONCLUSION about the organisms that might be causing the infection.

>>> 50a. MYCIN just made a conclusion about the organisms that will be

considered to be causing the meningitis for the purposes of therapy selection.

>>> 51a. MYCIN has made a FINAL CONCLUSION about the organisms that will be considered to be causing the meningitis for the purposes of therapy selection.

(Continuing our discussion of the therapeutic regimen of J.Smith [RULE092].)

Clause-by-clause discussion

{From now to the end of the session, the student gets the initiative exactly once. The tutor methodically completes discussion of the goal rule....}

We must now consider whether the organisms isolated from the pending csf culture should be considered for therapy.
Let's consider a way to determine this by definition....

(Now considering RULE049.)

We have already determined that the infection which requires therapy is known.

We have already determined that you have told the user which infection will be associated with this culture site.

We must now determine whether an attempt has been made to deduce the time since clinical evidence of the infection was first noted.
There are no rules which successfully apply to this case that can tell us about the time since clinical evidence of the infection was first noted.
Since it is not deducible, a consultant would have to ask to see if the person seeking advice knew about it.

The reported case data:
The time since clinical evidence of the infection was first noted is 1 day.

>>> 52a. MYCIN just made a conclusion about whether the organisms isolated from the pending csf culture should be considered for therapy.

>>> 53a. MYCIN has made a FINAL CONCLUSION about whether the organisms isolated from the pending csf culture should be considered for therapy.

>>> 54a. MYCIN has made a FINAL CONCLUSION about the number of cultures from this site since the onset of the infection.

(Continuing our discussion of whether the organisms isolated from the pending csf culture should be considered for therapy [RULE049].)

We can now conclude that the organisms isolated from the pending csf culture should be considered for therapy (considering the site of CULTURE-1, the infection which requires therapy and the time since clinical evidence of the infection was first noted [RULE049]).

However, there are other rules that we might discuss that use information you have already been given.

(End of our discussion of a rule to determine whether the organisms isolated from the pending csf culture should be considered for therapy [RULE049].)

(Continuing our discussion of the therapeutic regimen of J.Smith [RULE092].)

We must now consider the associated infections for which therapy should cover. We are not able to make a conclusion about the associated infections for which therapy should cover, because there are no organisms in this case.

(Continuing our discussion of the therapeutic regimen of J.Smith [RULE092].)

We must now consider the organisms that will be considered to be causing the meningitis for the purposes of therapy selection.

Try to make a conclusion about the organisms that will be considered to be causing the meningitis for the purposes of therapy selection.
What are the organisms that will be considered to be causing the meningitis for the purposes of therapy selection?
** DONTKNOW

{The student didn't care to answer. This parameter is not meaningful in this case, for it simply involves transferring results from one intermediate concept to another. If there were positive cultures, this clinical evidence would have to be reconciled with likely identities of organisms growing on these cultures.}

Information about the organisms that might be causing the infection is evidence that the organisms that will be considered to be causing the meningitis for the purposes of therapy selection are: e.coli (.37) pseudomonas-aeruginosa (.33) klebsiella-pneumoniae (.3) diplococcus-pneumoniae (.29) neisseria-meningitidis (.18) staphylococcus-coag-pos (.12) [RULE198].

(Continuing our discussion of the therapeutic regimen of J.Smith [RULE092].)
At this point, MYCIN would determine the best therapy recommendation from among the drugs likely to be effective against the significant organisms.
(End of our discussion of RULE092)

{Discussion of the main goal is now complete, and the tutor continues to follow the procedure for discussing a case. Comparison of the student's choice of therapy to MYCIN's has been disabled (see **[Clancey, 1979]** *for details.}*

This completes our discussion of PATIENT-538.
Saving student model...done.

Closing your typescript file <CLANCEY>TYPESCRIPT.CLANCEY/26-MAR-79;2
Shall I list it on the line printer (Yes, No, or Delete it)?
** NO
Do you have any comments?
** NO
Thank you.

3.4 The Student's Reaction to the Program

This single trial of GUIDON is not intended to be a controlled evaluation of the program. Nevertheless, it is useful to consider this student's reaction to the program, particularly remarks that were surprising, revealing tutoring issues to consider in later versions of the program.

3.4.1 Definition of the Problem

The first problem is that the student was unsure of his responsibility for the patient. Was he supposed to play the part of a "resident physician," reviewing a problem that other physicians had handled last week? Or was this one of his own patients who might require life-saving treatment in the next 24 hours? This student felt that knowledge of the setting was important for "getting the proper frame of mind." However, it is not clear whether this setting was a necessary condition for remembering how to solve the case or whether he would have felt simply more at ease if the case were placed in a familiar context.[1]

As the student received information about the patient, he felt uncertain about time relationships. For example, was the patient hospitalized because he was burned? MYCIN is particularly weak in this respect, for its parameters are more like isolated propositions than like facts that form a story.

Perhaps the most difficult problem that the student faced was in understanding

[1] Realism is frequently cited to be an important design criterion for medical CAI programs [Feurzeig, et al., 1964][Harless, et al., 1971].

MYCIN's terms. He asked, "How could a *current* drug have been *discontinued*?" "How could there be 'organisms from a pending culture'?" "How does the 'type of the infection' differ from the 'name of the infection?" "Is a soft tissue infection the same as a wound infection?" These are all good questions that required some explanation before the student could proceed.

3.4.2 Strategy

It was a surprise to see the student make an hypothesis on the basis of the initial case data alone. He knew that there was much more information to consider, but he wanted to see how GUIDON would react to his first thoughts. Indeed, the student followed the basic approach of using what he knew as a basis for gathering more information, rather than exploring what MYCIN knew about the problem. More importantly, he did not even fully read summaries of MYCIN's evidence for an hypothesis. All of this changed when the student said that he exhausted his knowledge of how to proceed. As we might expect, he then began to request help or a quiz. It was at this point that he took up the hardcopy handbook and went down the list, using it to jog his memory about case data to request.

Just as the student's cognitive *style* shifted (from using what he knew to relying on the program's suggestions) as he exhausted his expertise, we found that his problem solving strategy varied at the same time from attempting to confirm an hypothesis, to exhaustively collecting all available data for a goal. As long as data suggested plausible hypotheses, he pursued more data that would build a case. For example to confirm Neisseria-meningitidis, first suggested by the age of the patient,

he asked if the patient had a rash. It was surprising how quickly he felt comfortable with an hypothesis: he did not ask about the white blood count in the CSF because he assumed it was high on the basis of what he already knew.[1]

When the student could no longer think of factors that would confirm his hypothesis, he tried to recall factors for other hypotheses and asked about them (page 54). He characterized his approach as "jumping around, scrambling, knowledge-limited." Next, he used MYCIN's lists of factors (and the handbook) to remind him of other relevant questions (page 67). At this point he was no longer working on single hypotheses, though he restricted his requests to data that he knew how to use. In the final stage the student was exploring MYCIN relations that were new to him.

3.4.3 Interaction with the Program

It is not surprising that GUIDON's initial orientation to the problem is too long. The student said that he was overwhelmed by the goal rule, sketch of the subgoal tree, and orientation for determining the organisms that therapy should cover for. He said that he would have preferred to have four or five sentences to get him started on the problem. A simple statement like, "Your goal is to ...." would have been useful.

It is significant that the student used the ORIENT and QUIZ options immediately. He did not know what to do, and hoped that the program would tip its hand a bit. Unfortunately the ORIENT option was not intended to provide the crisp summary of what had occurred and was about to take place that the student desired. Indeed, at first the student did not know what a "topic" was.

[1] Paradoxically, GUIDON concluded that he left out the WBC in the CSF because it was too advanced for him to consider (page 57).

As has been mentioned, the student did not read everything that GUIDON printed. Sometimes he became weary of reading, and sometimes he skipped over statements in his exuberance to get on with the problem. Consequently, he missed the statement about "partially treated meningitis" (page 56). He missed the definition of "the type of the infection" (page 53) and later asked, "What kinds of answers does the program want?"

The persistence of the student's interest was especially surprising. During the response to the student's hypothesis about the type of the infection, the program went on to quiz him about hypotheses he did not mention, while the student still wanted to talk about the evidence for a bacterial infection. "What was the CSF glucose value?" he asked. The program takes up new subjects more quickly than a human usually does. However, it should be noted that the discussion to wrap-up consideration of a rule (page 76), including mention of related rules (page 77), is designed to imitate the typical way humans slow down before taking up other major subproblems. Rapid fire questioning, as takes place in response to a student's hypothesis, probably does not accommodate a typical student's pace for digesting new information.

The student often reread earlier portions of the dialogue, both to keep track of what had been determined, and to follow-up on remarks made by the program. This suggests that modeling strategies might use the remark indices that the student mentions as clues about what he has read or gone back to reread, contributing to the model of his thinking process (see Section 8.1).

Finally, when asked to summarize his reaction to the program, the student raised a provocative question. He felt that GUIDON was more sophisticated than other tutorial programs that he had used.[1] But, in contrast with simple CAI programs with limited options in which "you are always in control," he began to lose confidence in his model of the program. The complexity of the dialogue (quizzing, stating results, interrupting to present rules, summarizing evidence) gave the tutor an air of unpredictability. Because the student did not feel that he had a good model of the program's methods, he wondered if the program really had a good model of him. It is possible that this student would feel more comfortable (and more in control) in later sessions, after becoming more familiar with the options available and understanding the typical ways in which GUIDON takes initiative.

[1] The student had previously used the CASE system [Harless, et al., 1971] and Massachusetts General Hospital programs.

PAGE NOT FILMED
BLANK

Chapter 4

Knowledge Base and Control Overview: The Expert/Tutor Framework

CONTENTS

4.1 Components of the GUIDON System

4.2 The EMYCIN Formalism

4.3 Applicability of EMYCIN Systems for Tutorials
- 4.3.1 Assets of the Formalism
  - -- Flexible Access
  - -- Bag of Techniques & Dialogue Framework
  - -- QA Development
  - -- Domain-specific meta-knowledge
  - -- Multiple Domains
- 4.3.2 Limitations of the Original Explanation Capability
- 4.3.3 Other Rule Set Properties Important for Tutoring
  - -- Completeness
  - -- Validity
  - -- Size
  - -- Single Form of Expertise
  - -- Weak Model of Inquiry

4.4 GUIDON: Conversion of the Consultation System
- 4.4.1 Consultation Record and Its Reconfiguration
- 4.4.2 The Communication Model
- 4.4.3 Tutorial Rules and Discourse Procedures

The instructional paradigm of the case method involves presenting particular problems to a student, and providing a supportive environment to help him solve the problem. The major objective of the GUIDON project has been to construct a case-method tutor that uses (a) knowledge of the domain and (b) knowledge of tutoring strategies, while keeping the two distinct. This chapter provides an overview of the knowledge representations of both domain expertise (for solving the problems presented to the student) and tutoring expertise (for carrying on a dialogue with the student as he solves a problem). We emphasize here two crucial aspects of this separation of domain knowledge from the tutorial program: 1) How is the problem solution communicated to the tutor? and 2) What knowledge about tutorial dialogues will the program need? Chapter 5 goes on to consider what domain knowledge, in addition to that which MYCIN uses to solve the problem, is useful for tutoring.

GUIDON uses a MYCIN consultation program as a source of problem solving expertise. Below we describe the MYCIN formalism (Section 4.2), and characterize its assets and limitations (Section 4.3). Here we focus on the use of the infectious diseases knowledge base; Chapter 9 reports on the implementation of GUIDON in two other domains.

4.1 Components of the GUIDON System

Figure 4.1 shows the configuration of the GUIDON system. The most important feature is that the the domain knowledge is distinct from the teaching expertise. The "expert" (MYCIN) consists of an interpreter for applying the domain knowledge to particular problems, producing a store of dynamic knowledge which is accessible to

the tutor. The tutor can access all of the knowledge base as well, and uses it to discuss particular aspects of the problem solution with a student. Notice that additional domain knowledge has been provided for use in tutoring. (This is described in Chapter 5.) The main component of the *tutor* module is a collection of tutoring rules, hereafter ***t-rules***, for carrying on a dialogue, including methods for providing assistance, maintaining a student knowledge model, responding to his initiative, and so on. Beginning with the expert system, components of this framework are described further below.

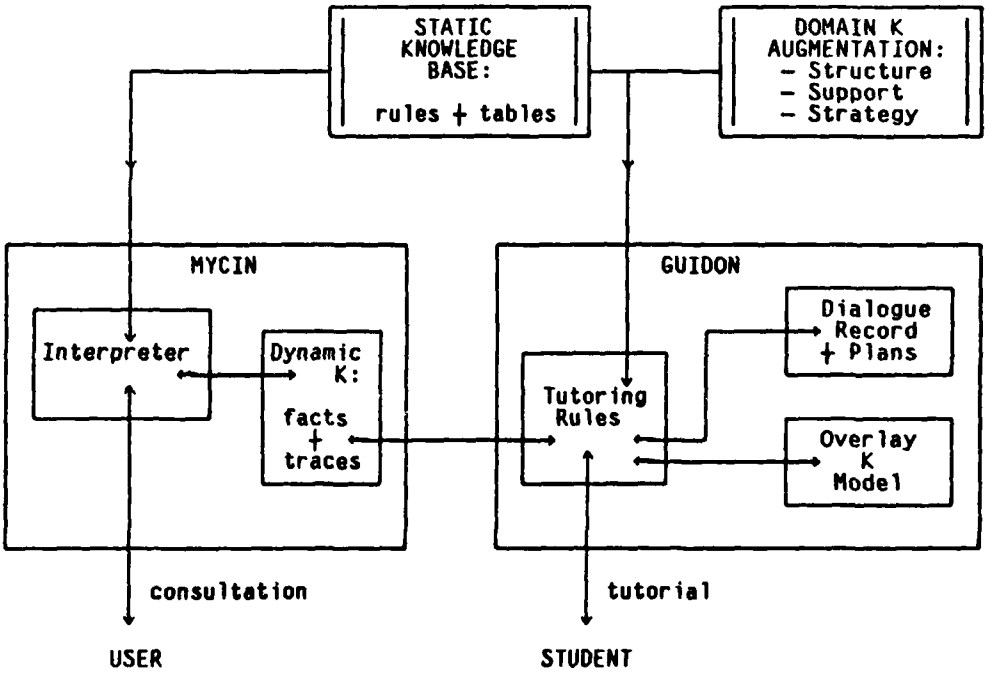


Figure 4.1. Modules for GUIDON: A Domain Independent Tutorial System

("K" = Knowledge)

4.2 The EMYCIN Formalism

The MYCIN knowledge base contains approximately 450 production rules [Davis & King, 1977]. Each rule consists of a set of preconditions (called the *premise*) which, if true, justifies the conclusion made in the *action* part of the rule. An example is shown below. We call each precondition a **subgoal**. If all of the subgoals in the premise can be achieved (shown to be true), then a conclusion can be made about the goal in the action. A classification of the predicate functions used in MYCIN rules is given in Appendix A.

> IF 1) the site of the culture is one of: those sites that are normally sterile, and 2) the gram stain of the organism is gram negative, THEN there is strongly suggestive evidence (.8) that there is significant disease associated with this occurrence of the organism.

```
PREMISE: ($AND (SAME CNTXT SITE (LISTOF STERILESITES))
               (SAME CNTXT GRAM GRAMNEG))

ACTION:  (CONCLUDE CNTXT SIGNIFICANCE YES TALLY 800)
```

Figure 4.2. Sample MYCIN Rule in English and LISP Form

In addition to the rules, there are several hundred facts and relations stored in tables, which are referenced by the rules, e.g., the list of body sites that are normally sterile is referenced by the rule shown above.

It is possible to remove the medical knowledge from MYCIN program and substitute a set of rules about a new domain. The consultation system will provide advice based on the new rules and the explanation system will explain the reasoning behind the advice. The domain-independent package consisting of the rule

interpreter, explanation, and knowledge acquisition modules is called EMYCIN ("essential MYCIN").[1] One EMYCIN program, called SACON [Bennett, et al., 1978],[2] provides advice on structural analysis problems, such as what stress and deflection behaviors should be observed in an airplane wing made of a certain material and subject to certain loading conditions. In addition, the EMYCIN system has been applied to two very different medical domains: interpretation of pulmonary function tests (PUFF) [Kunz, et al., 1978] and drug therapy recommendations for psychiatric patients (HEADMED) [Heiser, Brooks, & Ballard, 1978].

4.3 Applicability of EMYCIN Systems for Tutorials

In this section we make observations relevant to converting MYCIN to a tutorial program. Here assets and properties of the EMYCIN representation and the MYCIN *infectious disease* knowledge base are considered; Chapter 5 considers forms of domain knowledge that are distinct from the performance rules which MYCIN uses for solving problems.

4.3.1 Assets of the Formalism

In what ways are MYCIN-like systems particularly well-suited for use in tutoring?

■ First, the existence of representational meta-knowledge gives the tutorial program the capability to take apart the rules and use them to form quizzes, to guide

[1] EMYCIN is being developed by William van Melle and A. Carlisle Scott [van Melle, 1979].

[2] Collaborative project with the MARC Corporation, Inc.

the dialogue, to summarize evidence, to model the student, and so on.[3] ***Flexible access to the knowledge store*** makes it possible to provide a variety of interactive aids. For example, the student can use the program to evaluate his problem-solving ideas, or use the program to as an information retrieval system see directly what the program knows. In addition, this flexibility permits us to write a variety of tutoring rules that select and present teaching material in multiple ways (including quizzes, direct statements, and hints)--an important consideration if we are to use the MYCIN/GUIDON system for experimenting with different teaching strategies.[1]

■ Second, the rules themselves are well-defined problem-solving methods: the premise states explicitly when the method can be applied, and the action states the evidence that should be added to the knowledge store. It is convenient to think of the list of rules for determining a MYCIN parameter as ***a bag of techniques for achieving a goal***. Rule preconditions can be called "factors," "subgoals," or "deltacts" [Brown, Collins, & Harris, 1977], as we find convenient. Rule actions are goals, steps in diagnostic reasoning, or alternative hypotheses, again according to our perspective at the moment. The subgoal structure of the rule set was designed to break down the problem in a conventional way that would facilitate explanation, and

---

[3] Function templates are the most important kind of representational meta-knowledge in MYCIN systems. Templates "indicate the order and generic type of the arguments in a typical call to the function" [Davis & Buchanan, 1977]. The template for the function SAME (see Figure 4.2) is "(SAME CNTXT PARM VALU)." Using this template to "read the rules," GUIDON can extract "(LISTOF STERILESITES)" from the rule above to determine the value for "the site of the culture" that will satisfy the precondition. Templates are also used to reconfigure the output of the consultation (Section 4.4.1 and Appendix A).

[1] Note that the ability to present domain knowledge in multiple ways is characteristic of a *generative* tutor. Problem-solving knowledge is not "compiled in"; given appropriate tutoring methods, the program can reason about what it knows, adapting subject material to the needs of a particular dialogue with a student.

this turns out to be a useful ***framework for organizing the topic structure*** of a dialogue about a particular problem.

Considering the rules as methods or skills, ***we can superimpose taxonomies*** that relate them according to difficulty and typical usefulness, or group them structurally by syntax, analogy, prerequisite, or specialization/generalization links. Thus, there is a distinction between the "raw" expertise that links situations to actions and more theoretical concepts that show how this knowledge is interrelated. For the purpose of improving the program or experimentation, new organizations can be introduced independently of one another, without modifying the rules themselves.

The (unordered) bags of ***rules require no particular control structure*** for applying them to solve problems. In consultations we can use backward chaining (to restrict the questions asked to just those needed to apply rules in a particular case), while in a tutorial we can run the rules forwards to model the expert and student, or even order them according to a strategy for hypothesis revision (Section 6.2.3.2). Using forward-directed inferencing it is possible to tutor the student about redundancy and relevancy of the information he gathers. When he states and/or supports an hypothesis for a subproblem, it is simple to see if the evidence is consistent with his hypothesis. Thus, some of the important distinctions between the MYCIN consultation system and systems like INTERNIST begin to blur with the addition of control schemes and meta-knowledge for tutoring. In addition, the rules represent uncertain judgments, so it is possible to ***discuss relative pros and cons for each decision.***

■ Third, development of MYCIN's ***question-answer (QA) facility was useful ground work*** for a tutoring system. For example, for the QA system to work properly, it was necessary to keep the rules "clean," by keeping the meta-knowledge that describes them up-to-date, and never changing the formalism in a way that would prevent a program from reading the rules. Moreover, design of the QA system [1] made it necessary to work out in full detail all kinds of program outcomes, for example, all of the reasons why a factor would not be pursued by MYCIN during consultation.

■ Fourth, the development of ***domain-specific meta-knowledge*** [Davis & Buchanan, 1977] in the form of rule models and metarules (Section 5.1.4 and Section 5.3.2) was the seed for all of our subsequent work on meta-knowledge (Chapter 5). Preliminary studies using this meta-knowledge for knowledge acquisition and explanation led directly to its use in GUIDON for orienting the student, summarizing evidence, and so on. Moreover, continuing research with GUIDON is very much directed by the now evident parallel between the knowledge acquisition program (transferring expertise from human to program) and tutorial program (transferring expertise from program to human student).

■ Fifth, the production rule formalism used by MYCIN is ***widely applicable to tasks other than medicine***, although it is by no means a "universal" language.[2] The

---

[1] The initial system was written by Ted Shortliffe. Later improvements were made by A. Carlisle Scott, the author, and William van Melle [Davis, 1976] [Scott, et al., 1977].

[2] Production rules have worked well in several domains [Feigenbaum, 1977]. For example, PROSPECTOR [Hart, 1975], the SRI International program, was designed originally as a MYCIN-like system for consulting with geologists about mineral exploration sites. SU/X is a signal processing program for military applications; it also makes use of a production rule formalism [Nii & Feigenbaum, 1977]. However, other approaches for building knowledge-based systems are possible. For example, Pople successfully uses frame-like *disease hypotheses* [Pople, 1977], and Weiss and Kulikowski use a causal-associational network [Weiss, Kulikowski, & Safir, 1977].

tutoring system will work with cases and rules in any EMYCIN system, assuming some parallels between the structure of the knowledge in the new domain and the structure of the existing medical knowledge (see Chapter 9 for discussion).

4.3.2 Limitations of the Original Explanation Capability

The production rule formalism is the key to MYCIN's explanation capabilities: steps in the problem solution are associated with specific rules whose applicability is well-defined. A typical QA interaction is shown below.[1]

```
** DO YOU EVER USE THE AGE OF THE PATIENT TO CONCLUDE DIPLOCOCCUS?
The following rules use:
     the age of the patient
to conclude about:
     the likelihood that Diplococcus-pneumoniae is one of the organisms
          (other than those seen on cultures or smears) which might be
          causing the infection

535, 534, 554, 507, 341

Which of the rules do you wish to see?
      ** NONE
```

Figure 4.3. General Question for MYCIN's QA System

Experience with MYCIN's QA system leads to the conclusion that reading back a trace of problem-solving steps (the rules) will not be sufficient for a tutorial system. Indeed, this is easily observed by any naive user who tries to unravel MYCIN's reasoning.[2] Some of the lessons we have learned about explanation are summarized below.

---

[1] The system simply does one-level indexing by rule preconditions and conclusions; no chaining of inferences is considered.

[2] It is worth noting that Swartout, using an entirely different representation of problem-solving knowledge (OWL procedures), has encountered some of the same difficulties in providing explanations for the Digitalis therapy advisor [Swartout, 1977].

1) The explanation system should summarize conclusions by ***hierarchically describing key reasoning steps***, rather than providing a list of all of the rules that were applied. The real problem in explanation is to abstract from performance--not merely to say what was done, but why it was done and what alternatives were considered.

2) Strictly speaking, a program that lacks a *user model* cannot be said to have an "explanation" capability. There is an important difference between passively retrieving information and restating it for a particular listener.

3) A naive user doesn't know where to begin asking questions. It is important that an explanation program convey the ***general structure of the solution process*** so that a user has a handle for pursuing details. The vocabulary must be shared.

4) Problem-solving steps may bear only indirect relation to ***underlying mechanism models*** for the domain. Expertise for efficiently cracking a problem is in some cases orthogonal to the general principles that are often taught to a beginner. (See Chapter 5 for discussion.)

These are difficult problems, though some progress has been made with all of them in GUIDON. The main question asked by Chapter 5 is: if teaching involves more than simply reciting expert performance rules, what else does the tutor need to say?

4.3.3 Other Rule Set Properties Important for Tutoring

To some the degree, GUIDON's design and ultimate capability to teach is a reflection on the particular rule set available. Some of the general properties of the infectious disease rule set are considered below.

Completeness

The success and usefulness of GUIDON's tutorial ability depends to some extent on the universality of the rule set, i.e., how the subject matter has been structured by the rule authors. The domain rules were designed to be comprehensible to potential users of the consultation system who are not experts, but it is inevitable that some of the concepts represented in the rules are artifacts of the particular representation and the requirements of making everything explicit. Furthermore, GUIDON acts as if the domain of discourse is closed (i.e. limited to the expert rule base), when in fact it is not: the student may want to discuss controversial diagnostic procedures not known to MYCIN.

Validity

Two formal evaluations of MYCIN's performance have demonstrated that MYCIN's competence in selecting antimicrobial therapy for meningitis and for bacteremia is comparable to that of the infectious disease faculty at Stanford University School of Medicine (where MYCIN was developed) [Yu, et al., 1978]. From this we conclude that MYCIN's rules capture a significant part of the knowledge necessary for demonstrably high performance in this domain. The fact that the formalism of production rules has been exploited in MYCIN to create a rich, high performance knowledge base for solving difficult, real world problems is an important starting point for demonstrating the advantages of using this representation of domain knowledge for tutoring.

Size

The effectiveness of GUIDON's current dialogue strategy of stacking topics partially depends of the depth of the AND/OR solution tree. In MYCIN it is usually only two levels below the top goal of determining therapy for the patient. The other EMYCIN systems generally share this property. (See Chapter 9 for further discussion.)

On the other hand, the very fact that MYCIN has hundreds of rules has made it important to develop tutoring strategies for carrying on the dialogue as economically as possible. There is no time to explicitly discuss with the student each inference that MYCIN attempted during the corresponding consultation. Section 6.2.1 deals with this problem in detail.

Single Form of Expertise

Multiple forms of expertise are sometimes important for tutoring. For example, when mechanistic reasoning is involved, both qualitative and quantitative representations may be useful to solve the problem. DeKleer has found that strategies for debugging an electronic circuit are "radically different" depending on whether one does local mathematical analysis (using Kirchoff's laws) or uses a higher level, functional analysis of components [Brown & Burton, 1975][Brown, Rubinstein, & Burton, 1976] [de Kleer, 1979]. One might argue that a tutor for the electronics domain should be ready to recognize and generate arguments on both of these levels. (See [Carr & Goldstein, 1977] for related discussion.)

The MYCIN program demonstrates that an adequate set of rules for reasoning in infectious disease problem-solving can be based on judgments about empirical information, rather than arguments based on causal mechanisms [Weiss, Kulikowski, & Safir, 1977]. MYCIN's judgments are "cookbook" responses that address the data directly, as opposed to attempting to explain it in terms of physiological mechanisms. Even though MYCIN's domain apparently makes it possible for cases to be solved without recourse to the level of physiological mechanisms, a student may find it useful to know this "support" knowledge that lies behind the rules (see Section 6.2).

Weak Model of Inquiry

There is no strong model for ordering the collection of evidence in MYCIN [1] Medical problem solving has recourse to strategies for building cases, but there are no hard and fast rules for evaluating a student's approach. While there are some conventions that ensure that all routine data are collected, physicians have no agreed upon basis for numerically optimizing the decision of what to do next. (See, for example, [Sprosty, 1963].) The uncertainty of procedures for reasoning in this domain suggests that a tutor will want to accommodate different problem-solving approaches.

[1] In Goldstein's WUMPUS program, for example, it is possible to rank each legal move (analogous to seeking case data in MYCIN) and so rate the student according to *rejected inferior moves* and *missed superior moves*. The same analysis is possible in Burton and Brown's version of the WEST program.

4.4 GUIDON: Conversion of the Consultation System

4.4.1 The Consultation Record and its Reconfiguration

MYCIN's production rules are applied by an interpreter using a backward chaining control scheme. Backward chaining can be described as follows:

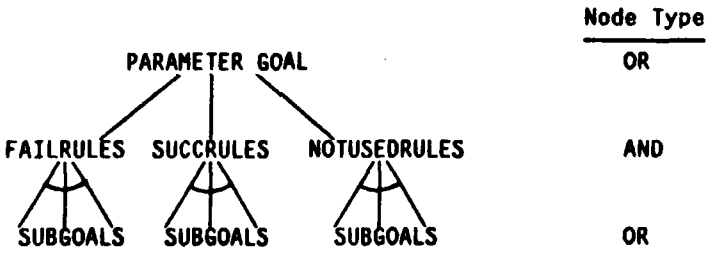
> If a condition in the IF (premise) part of a rule is decidable from the database, the value is noted and used directly; if the condition can be asserted by the THEN (action) part of some other rules, they are applied; otherwise, MYCIN asks the user. (after **[Szolovits & Pauker, 1978]**, p. 136)

The output of the consultation includes case data acquired from the user and conclusions made from rules that were successfully applied. For GUIDON's purposes, MYCIN's compound predicate functions (corresponding to AND, OR, "there exists an X such that P," "there is a maximum (minimum) X such that P," and "there is a subset of L, such that P") have been modified to leave behind traces of their evaluation for later analysis. These traces indicate which clauses of the compound predicates were tried, the context of evaluation, and whether they succeeded. As such, the traces are significantly more complex than those used by the post-consultation question-answer program in MYCIN. (The precise format and recursive construction of the traces is described in Appendix A.)

Next, in batch mode, GUIDON reconfigures the consultation traces to form a complete AND/OR tree representation of the problem solution. The analysis is non-trivial, for it requires that the nesting of predicates be taken into account. For

example, if (NOT (THEREXISTS an X such that P)) is a false rule precondition, then the program must record that every element of X must be known before the rule can be evaluated. Moreover, the tree is complete in that it explicitly records all preconditions of a rule that fail.[1] (This information is crucial for applying the rules in a forward direction during the tutorial to model the student's understanding (see Section 4.4.2 below).) The resulting tree allows the tutorial program to efficiently guide the dialogue without continuously re-analyzing the rules for relationships between subgoals.

Figure 4.4 shows the basic outline of the AND/OR tree. A parameter goal (actually a parameter/context pair, such as SITE/CULTURE-1) is an OR node with three groups of rules, those that failed, succeeded, and were not used. These rules are AND nodes whose preconditions constitute subgoals. Pointers tie goals, rules and subgoals in all directions.

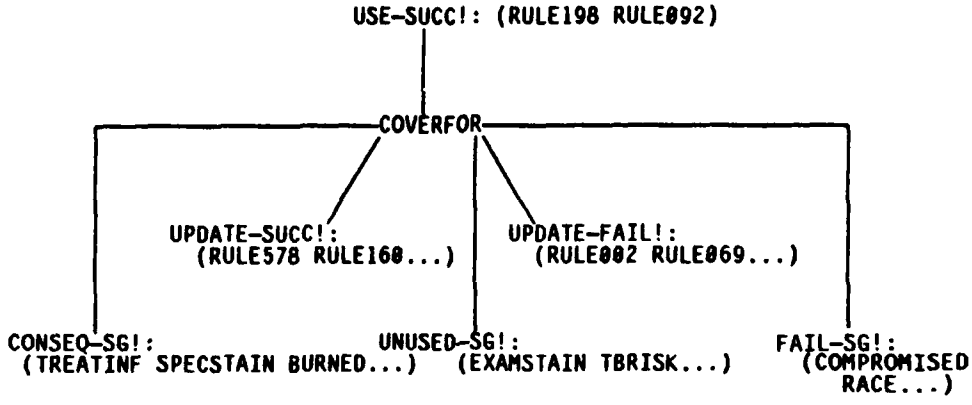


Pointers	Linked Nodes
SG	goal–subgoal
UPDATE	goal–rules
CONCLUSIONS	rule–hypothesis values of goal
USE	rule–subgoals

Figure 4.4. General Form of the AND/OR Tree

[1] But see Appendix A for a discussion of the logical problems involved, and insufficiencies of the current approach.

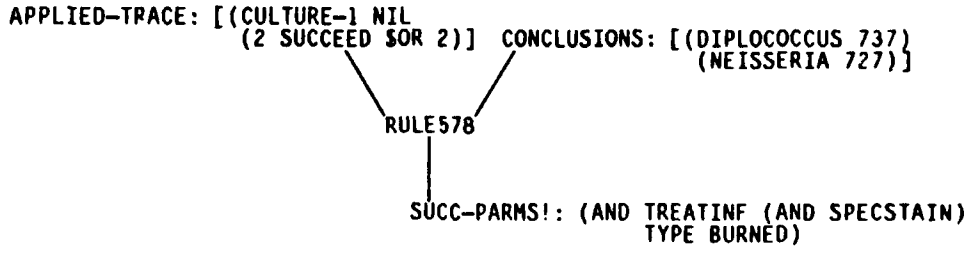
A typical OR node is shown in Figure 4.5. Observe, for example, that the parameter COVERFOR is used successfully in rules 198 and 92, and rules 578 and 160 are two rules that determine COVERFOR. COVERFOR is a subgoal for REGIMEN and TREATINF is one of its subgoals. The RACE of the patient appears in a rule that failed to make a conclusion about COVERFOR in this case. (Note that only parameter and rule names are shown here. In the tree, a context is always included, e.g., ORGANISM-2.)



KEY: USE-SUCC! = subgoal in rules that succeed
UPDATE-SUCC! = rules used to determine goal
UPDATE-FAIL! = rules that failed to apply
CONSEQ-SG! = subgoals in rules that succeed
UNUSED-SG! = subgoals that appear in rules for determining the parameter, but don't need to be known (e.g., true preconditions of rules that fail)
FAIL-SG! = subgoals that cause rules to fail

Figure 4.5. OR node in the reconfigured tree

A portion of the consultation record that was reconfigured to form this tree is shown in Figure 4.6. The components of the record and logical reconfiguration of the rule (SUCC-PARMS!) are described in the key.



KEY: APPLIED-TRACE = consultation record, indicates that clause 2 of the internal OR succeeded.
CONCLUSIONS = consultation record, indicates two organisms and their certainty factors.
SUCC-PARMS! = logical reconfiguration based on the rule's premise and applied-trace; indicates that when the parameters TREATINF, SPECSTAIN, TYPE, and BURNED are known, this rule should be marked as successful and its conclusions applied to other rules.

Figure 4.6. AND node in the reconfigured tree

To illustrate some uses of this tree, Figure 4.7 shows one portion without labels for the pointers. "COVERFOR" signifies the goal to determine which organisms should be considered for treatment; rule 578 concludes about this goal; "TYPE" is a subgoal of this rule.

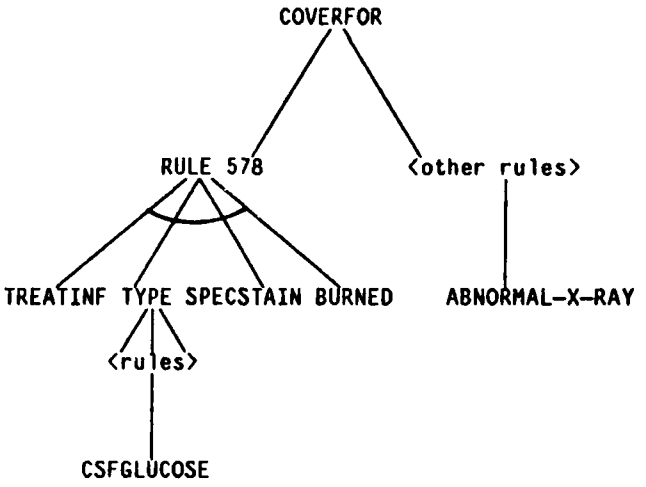


Figure 4.7. A portion of the AND/OR tree of goals and rules

Tutorial rules make frequent reference to this data structure in order to guide the dialogue. For example, the response to the request for help shown in the protocol (page 68, "look for diplo") is based first of all on the rules that were used by the expert program to determine the current goal. Similarly, the t-rules for supplying case data requested by the student check to see if the expert program asked for the same information during the corresponding consultation (e.g., the "csf glucose" on page 54).

Figure 4.7 is useful for illustrating several characteristics of EMYCIN production rules that must be considered during the tutorial dialogue. First, we define a factor of a goal, G, to be those subgoals that appear on the first level in the tree below G. Thus, TREATINF, SPECSTAIN, TYPE, BURNED, and ABNORMAL-X-RAY are all factors of

COVERFOR.[1] Sometimes when we look at a particular rule, say rule 578, we find it useful to distinguish between factors that are determined by case data and those that are determined by applying other rules. We will call factors determined by applying other rules **subgoals** in this context. Thus, TYPE is a subgoal in rule 578, while BURNED is a **case data factor**.

Second, the AND/OR tree is used by the tutor to relate topics to one another in response to student options. We see on page 54 that, during the discussion of the type of the infection, the student requested information about the patient's chest x-ray. Since ABNORMAL-X-RAY is not a factor of TYPE and is not a factor of any subgoal of TYPE, it is defined to be *irrelevant* to the current goal. Given that GUIDON is using a pre-compiled solution, it must cope with student reasoning that is apparently not connected to this tree. See Section 6.2.2.1 and Section 7.1.1.2 for discussion.

In EMYCIN systems, a distinction is drawn between those parameters that are determined by trying rules before asking the user for a value, and those parameters for which one should ask first. The following table shows the four possible kinds of parameters:

[1] The entire list is computed by forming the union of consequent, antecedent, and failure subgoals.

	Are there rules? Yes	No
Data usually known by the user	Ask, then try rules	Ask only
Requestable	Rules, then ask	X
Can't ask	Rules only	X

Figure 4.8. Kinds of Parameters

GUIDON has to determine the appropriateness of asking for case data at particular times in the problem solution by checking the kind of parameter being requested by the student and examining the status of the expert program's solution. For example, on page 70, we see that the student requested ALCOHOLIC, a parameter that one should try to deduce from rules before asking the user. Since the expert program had not yet been able to try these rules (it is not yet known whether the patient is IMMUNOSUPPRESSED), it is premature for the student to request this information. Section 6.2.2.1 discusses how GUIDON copes with this situation and others similar to it.

Finally, we will frequently refer to an hypothesis for a goal. For true/false parameters, the *value* of the hypothesis can be either "yes" or "no," as in "I think that the patient has had a splenectomy." For parameters like "the type of the infection," an hypothesis must be a single value from finite list of possibilities; so we might say

that "the student is considering that the type of the infection is bacterial." For a third kind of parameter, it is possible for more than one hypothesis to be correct. "The organisms that are causing the infection" is an example--more than one organism might actually be present. These distinctions play a role in hypothesis evaluation and formation of questions from the rules. (On the most basic implementation level, they are important considerations when constructing tutorial remarks from text strings.) Section 6.2.2 gives many other examples of the use of domain logic in the tutorial dialogue.

4.4.2 The Communication Model

During the tutorial session, GUIDON maintains a record of the communication between tutor and student. The most important component of the record is the student knowledge or *overlay* model. This model indicates for every rule: a) the belief that the student knows the rule, and b) the belief that he has applied it in his solution of the case under discussion. For each goal, the model indicates the combined conclusion that the student can draw, given the rules he has applied. This conclusion is the hypothesis that the tutor believes that the student is ready to support. GUIDON's overlay model is used to relate student behavior to what MYCIN knows; it is not expected that the rules will be precise model of the student's thinking. Determining the differences and the extent to which they matter is one objective of continuing research.

The model is maintained as follows. When case data are given to the student, the tutor uses the AND/OR tree to determine the rules that the expert has been able

to apply, given this new information. Thus, domain rules (hereafter, ***d-rules***) are applied in a forward (antecedent) direction to maintain an exhaustive, dynamic record of rule applications and subgoal conclusions. Section 8.3.2.2 describes this in more detail.

When the expert applies a d-rule, special t-rules for modeling are used to decide whether the student has also applied the d-rule, based on his knowledge of the rule and his ability to resolve subgoals in the premise. Then, just as the tutor maintains the record of the expert program's conclusions, it recursively considers the student's use of this new knowledge by examining the d-rules that use it for further modifications to the knowledge model. Furthermore, when the tutor explicitly tells the student about a rule's application, the student model is updated to reflect the student's possible use of this information. This model is called an "overlay" model because it characterizes the student's knowledge solely in terms of the expert knowledge base--it is an overlay on the expert rules [Goldstein, 1977] [Carr & Goldstein, 1977]. (Details and discussion of limitations appear in Chapter 8.)

The overlay student model is used for *opportunistic tutoring* (page 67) (presenting material to fill in possible gaps in the student's knowledge as opportunity arises during the dialogue), evaluating a student's hypothesis (page 56) determining which d-rules the student used to make an hypothesis), and for guiding the dialogue (page 76) (selecting d-rules and presentation strategies as a form of help for solving the problem).

Other components of the communication model include a session profile (Section

7.1.3) to control the tutor's remarks (page 43), a focus and discussion record (Section 7.1.1.2) to maintain coherency, and a record of tutorial remarks (Section 7.1.4.3) to allow the student to follow up on what the program says to him. Finally, GUIDON's t-rules often reference the *lesson plan* of topics that represent the program's goals for the session (topics and rules to be sure to mention). In the current system there is simply a single default strategy that says a rule is "on the lesson plan" if it is not more than two difficulty levels above the student's sophistication (see Section 8.2.1).

The communication model of GUIDON borrows from theory developed in natural language research (see, for example, **[Grosz, 1977] [Winograd, 1977] [Bobrow, et al., 1977]**), but makes no attempt to be a complete model of human interaction in itself. We use natural language research terminology where appropriate to describe features we have found useful to incorporate into GUIDON.

4.4.3 Tutorial Rules and Discourse Procedures

Performance knowledge for carrying on a case dialogue is represented in GUIDON as sequences of rules. Proceeding from an extensive hand simulation, patterns in the rule-based tutorial dialogues were formalized into procedures that constitute sequences of action options for the tutor, and conditions for controlling their use. Figure 4.9 shows a typical tutorial rule.

```
T-RULE31.04

If: 1) The number of factors appearing in this rule which need to be
    asked by the student is zero, and
    2) The number of subgoals remaining to be determined before this
    rule can be applied is equal to 1

Then: Substep i. Say: suggest-subgoal
      Substep ii. Discuss the goal with the student in a
                  goal-directed mode [Proc001]
                   -> A prompt should not be given before starting
                  discussion of this goal
      Substep iii. Wrap up the discussion of the rule being
                   considered [Proc017]
```

Figure 4.9. Tutorial rule for Presenting a Domain Rule

A range of tutorial methods have been formulated in t-rules, consisting of ways to discuss a domain rule, respond when the student requests case data, end discussion of a topic, respond to the student's partial solution of a subgoal, and so on. In addition, special t-rules are used to maintain the overlay model; other t-rules select valid question types when quizzing about a rule; and some represent domain-independent strategies for diagnostic reasoning. These procedures and tutorial rules have been written in stylized code so they are translatable into an English form. This makes it easy to show other researchers and educators the specific rhetoric patterns, domain knowledge, and teaching strategies used by GUIDON. There are about 45 such procedures containing almost 200 t-rules in the current formalization of the case method dialogue. The formalism and procedures are described in considerable detail in Chapter 6, Chapter 7, and Chapter 8. The current set of t-rules is listed in Appendix F.

PRECEDING PAGE NOT FILMED
BLANK

Chapter 5

Meta-levels of Knowledge Required for Tutoring

CONTENTS

5.1 Structural Knowledge -- for organization
- 5.1.1 Parameter separability
- 5.1.2 Parameter screening relations
  -- implicit & explicit
- 5.1.3 Kinds of rules
  -- iterative, tabular, definitional
- 5.1.4 Rule clusters
  -- models, schemas, subgoal structure
- 5.1.5 Rule interactions
  -- cascading CFs, programming

5.2 Support Knowledge -- for explaining rules
-- The Tetracycline example; the need for other knowledge

- 5.2.1 What's in a rule?
  -- what's there and why
  -- rules are optimized: they can be rewritten to include intermediate concepts
- 5.2.2 Principles Missing from Optimized Knowledge
  - 5.2.2.1 DATA ABSTRACTION principles
    -- premise references empiric data rather than conceptual interpretation
  - 5.2.2.2 INFERENCE principles
    -- conclusion patterns are hidden in a specific table of CFs
  - 5.2.2.3 MECHANISTIC principles
    -- causal connection that ties premise to conclusion is not explicit
- 5.2.3 Implications for tutoring
  -- Teaching wrong level: demanding explanations on level of detailed data vs. general concepts and relations; thus, can teach how to subdivide the problem, but not able to provide explanatory principles.

5.2.4 Explanation Models
-- Additional knowledge that gives GUIDON the capability to say why a rule makes sense

5.2.4.1 Determining key factors
5.2.4.2 Evidence classes
5.2.4.3 Critical events
5.2.4.4 Recap: Algorithm for explaining a rule
5.2.4.5 Completeness
-- Explaining the events themselves
5.2.4.6 More sophisticated reasoning
-- Multiple-clause arguments
5.2.4.7 Summary
5.2.4.8 Speculation: Knowledge acquisition

5.2.5 Conclusions about Support Knowledge

5.3 Strategical Knowledge -- for planning solutions
5.3.1 Surface and Deep Structure of Rule Systems
5.3.2 Representing Planning Knowledge in Meta-rules
5.3.3 Meta-strategies

5.4 Epilogue: What is Meta-knowledge?

To use GUIDON for teaching students how to do diagnostic reasoning, we must be clear about what a student must learn in order to become an expert, and how this relates to the knowledge that is encoded in the MYCIN consultation system. In addition, there may be domain knowledge that a tutor might need for ancillary tasks like posing valid questions to the student, conforming to discourse conventions, and modeling the student's understanding. These considerations should be of interest to other researchers who wish to use a rule-based expert system for tutoring.

This section considers domain knowledge that is required for tutoring that is not explicit in MYCIN's rules. The point of the exposition is that in the development of MYCIN's knowledge base, using the simple, uniform rule formalism, certain distinctions that are important for tutoring have not been made. For example, it is possible to make distinctions among types of rule clauses in order to account for why a rule is written the way it is. Moreover, given the focus on "getting the right answer" during MYCIN's development, facts and relations were left out--the understanding of processes and abstract concepts that an expert knows which lead him to believe that the rules make sense. These omissions apparently do not hinder MYCIN's problem-solving competence, but they severely limit a tutorial program's ability to reason about the rules; preventing it from discussing mechanistic principles with the student, summarizing evidence by abstracting key points, using analogies to explain inferences, and so on.

The additional domain knowledge we have found useful for tutorials is of three types: (1) SUPPORT knowledge (or knowledge that justifies a rule), (2) STRUCTURAL knowledge (patterns in the rules and parameters of the knowledge base), and (3)

STRATEGICAL knowledge (meta-rules for controlling lines of reasoning) (see Figure 5.1). The examples below are intended to support our claim that transfer of expertise requires more than a regurgitation of problem-solving methods, which is all that the original MYCIN explanation system could do. The discussion of strategical knowledge in Section 5.3 develops this point most thoroughly, drawing upon some related work in education and psychology.

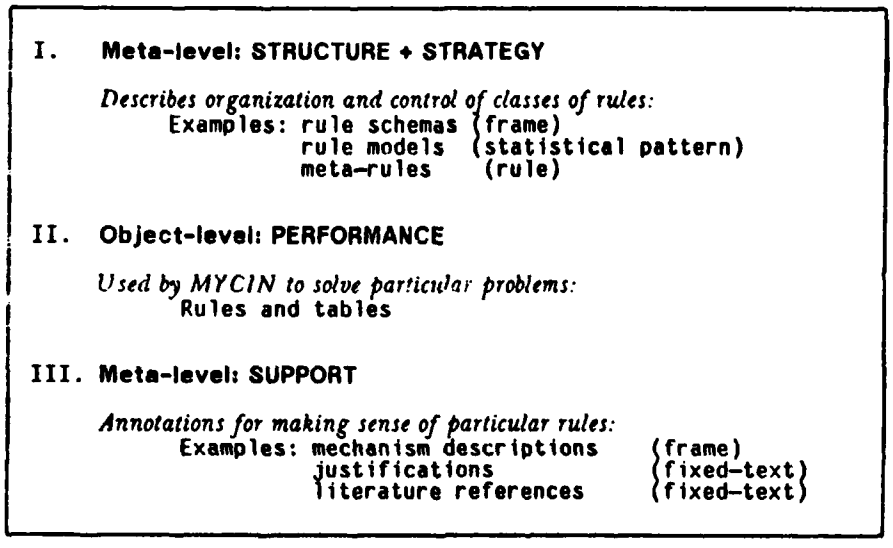


Figure 5.1. Knowledge Base Organization into Three Tiers

Figure 5.1 places the meta-levels of knowledge in relation to the performance rule base that we described in Section 4.2. We find it convenient to group strategical and structural knowledge "above" the level of rules because this meta-level knowledge describes the organization and use of classes of rules. In general we will find it convenient to consider support knowledge as being "attached" or specific to each rule. However, as we will see, mechanism descriptions for the domain are an

important part of support knowledge that could be used to form analogies among groups of rules as well. Figure 5.2 illustrates the three tiers associated with a particular rule. Note that rule models, meta-rules, and the author annotation were added by Davis for use in the consultation system [Davis, 1976]; their use in GUIDON for tutoring complements Davis' work. The rule schema, key factor, dual, and mechanism description are additions that we have made in the course of developing GUIDON.

Briefly, the rule schema and key factor allow GUIDON to refer to a rule without fully printing it, as in summaries of evidence for an hypothesis (page 56). The dual rule relationship, which relates analogous rules, is not currently used. The mechanism description (property of the key factor) is used to produce a rudimentary explanation of the rule (Section 5.2.4). Fixed-text properties are printed at the student's request.

Subsequent sections describe the three kinds of knowledge that we have detected to be missing from MYCIN's rules (these follow the levels *structure*, *strategy*, and *support* in Figure 5.1); specific examples are given in each section. The section on support knowledge goes into considerable detail to demonstrate the feasibility of augmenting MYCIN's rule representation. An epilogue considers the concept of meta-knowledge, distinguishing between the notion of levels that we have used here and other uses of the term.

<organization and control level>

RULE-SCHEMA: MENINGITIS.COVERFOR.CLINICAL
RULE-MODEL: COVERFOR-IS-MODEL
DUAL: D-RULE577

<performance or object level>

D-RULE578

If: 1) The infection which requires therapy is meningitis, and
2) A: A smear of the culture was not examined, or
B: Organisms were not seen on the stain of the culture,
3) The type of the infection is bacterial, and
4) The patient has been seriously burned
Then: There is suggestive evidence (.5) that pseudomonas-aeruginosa is one of the organisms (other than those seen on cultures or smears) which might be causing the infection

UPDATES: COVERFOR
USES: (TREATINF SPECSTAIN EXAMSTAIN TYPE BURNED)

<support level>

KEY-FACTOR: BURNED
MECHANISM-DESCRIPTION: (PORTAL-PATHWAY "a severe burn" "air" "wound")

JUSTIFICATION: "For a very brief period of time after a severe burn the surface of the wound is sterile. Shortly thereafter, the area becomes colonized by a mixed flora in which Gram pos organisms predominate. By the 3rd post burn day this bacterial population becomes dominated by Gram neg organisms. By the 5th day these organisms have invaded tissue well beneath the surface of the burn. The organisms most commonly isolated from burn patients are Pseudomonas, Klebsiella-Enterobacter, and Staph. Infection with Pseudomonas is frequently fatal."

LITERATURE: MacMillan BG: Ecology of Bacteria Colonizing the Burned Patient Given Topical and System Gentamicin Therapy: a five-year study, J Infect Dis 124:278-286, 1971.
AUTHOR: Dr. Victor Yu
LAST-CHANGE: Sept. 8, 1976

Figure 5.2. Domain Rule 578 and Its Associated Documentation

[All information is provided by a domain expert, except for the "key-factor" which is computed by the tutor from the rule schema and contents of the particular rule. See Section 5.1.4.]

5.1 Structural Knowledge -- for organization

GUIDON makes frequent use of knowledge about the structure of MYCIN'S knowledge. Most of this structural knowledge plays no part in MYCIN consultations; it has been added specifically for its usefulness in tutoring. This knowledge is classified below into: 1) concept definitions, 2) kinds of parameters, 3) parameter clusters, 4) kinds of rules, 5) rule clusters, and 6) rule interactions. In some cases, constructs were added to the MYCIN system during its development for their organizational benefit, e.g., tabular rules (Section 5.1.3). Other constructs, e.g., rule schemas (Section 5.1.4) might have been useful during the knowledge acquisition phase if they had been available. Discussion of the use of structural knowledge in tutorials appears in Section 6.2.3.1.

5.1.1 Parameter Separability

Perhaps the first step in designing a MYCIN-like system is to define the concepts that will be named by the parameters used in rules. When parameters are abstract and can be inferred from one another, it may be challenging to keep them well-defined. Overlapping or partially redundant parameters will complicate a student's ability to make sense of the rule set, and may seriously hinder the extent to which he can incorporate them in his own problem-solving approach.

Consider the "clostridia rule" (rule 32) and its justification. This rule uses information about visible characteristics of an unknown organism colony growing in a culture taken from the patient to infer whether this evidence of disease is

"significant." Significance is an important criterion for deciding whether to prescribe therapy to combat the organism. For example, if this is a naturally occurring organism that has contaminated the culture, then it should not be treated.

RULE032

```
If:  1) The site of the culture is one of: those sites that are normally
        sterile, and
     2) The stain of the organism is grampos, and
     3) The morphology of the organism is rod, and
     4) The aerobicity of the organism is anaerobic
Then:  There is weakly suggestive evidence (.4) that there is significant
         disease associated with this occurrence of the organism

COMMENTS: This rule is directed at the possibility that Clostridia species
      may be the pathogen.  Further helpful information that is not in the
      rule (but should be) is the relative size of the rod and its
      configuration.
      (Clostridia are big rods, diphtheroids are small and pleomorphic.)
```

Figure 5.3. Rule that Exhibits Overlapping Use of Parameters

Notice that this rule does not mention the organism species Clostridia, yet the rule author's comments indicate that he was considering the possibility that a gram-positive anaerobic rod might belong to the species Clostridia. So is this rule about the identity of the organism, its significance, or something else?

The medical justification for this rule is that the species Clostridia is rarely found as a contaminant in a culture, i.e., when it grows in a culture, it usually has come from the body site that was sampled, rather than, say, introduced to the sampling device as it passed through the skin (thus picking up organisms that are found on the skin surface of any healthy adult or child, i.e., diphtheroids). It so happens that "contaminant" is another concept in MYCIN. In fact, this rule should be rewritten to conclude that Gram-positive rods are not contaminants.

The point is that whether you are a student or an expert coming to MYCIN intending to write new rules, you will encounter some difficulty sorting out some of the intermediate concepts (see the subgoal tree structure below).

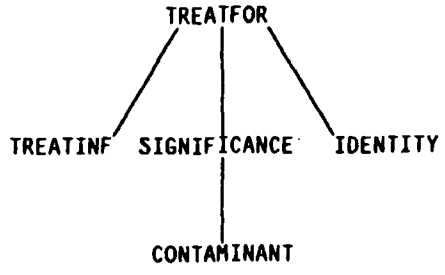


Figure 5.4. Portion of MYCIN's Inference Structure

Since there is a rule that a contaminant is not significant, how do the two concepts really differ? Part of the problem is that prior choices of concepts and rules to include in the knowledge base affect later additions. Thus, there is a rule

(1) contaminant => not significant *but not a rule*
(2) not contaminant => significant.

So if you want to say that gram-positive anaerobic rods require therapy (which is what SIGNIFICANCE ultimately amounts to), you can write a rule like the clostridia rule currently appears or else add the missing rule (2) and rewrite the clostridia rule to conclude "not contaminant." (Grampos, anaerobic rod => not contaminant => significant) Indeed, the multiplicity of ways to get the same final result has contributed to inconsistencies that one sometimes finds in the rule base. Some sort of constraint or discipline is missing that would make the definitions of concepts more precise. We speculate that providing explicit justifications for the rules according to

models like those described in Section 5.2.4 would help sharpen the rule authors' use of concepts in MYCIN.

The sample protocol illustrated how a student could be easily confused about MYCIN's concepts. In this dialogue, the concept "type of infection" was confused with "name of infection"; the student told the experimenter that simply stating the possible alternative hypotheses to be determined would have helped (i.e., "bacterial, viral, tb, or fungal").[1] We have provided GUIDON with a text definition file for many of MYCIN's parameters. Moreover, *reprompts* used in a MYCIN consultation to rephrase a question for a user have been modified so that all clarifications are contained in the definition file, allowing this information to be used more readily in tutorials. The definition lists examples of situations when the concept holds:

```
** DEFINE TBRISK
Definition of:  whether the patient has a tb risk factor

        The risk factors for tb are: a) positive intermediate strength PPD,
b) history of close contact with a person with active tb, c) a household
member with a past history of active tb, d) apical scarring on chest xray,
e) history of a granulomas on biopsy of liver, lymph node, or other organs.
```

Figure 5.5. Defining a MYCIN Concept by a List of Qualifying Criteria

5.1.2 Parameter Screening Relations

When a parameter can be used to determine the applicability of one or more other parameters in a particular case, we say that the parameter serves as a *screen*

[1] GUIDON did this in its orientation (page 53), but the student evidently missed the remark. From this we conclude that information like this must be highlighted and repeated.

for the other parameters. MYCIN has two forms of screening relationships: those that are implicit in rules, and those that are explicit in the form of data tables.

Implicit Screening Relations

MYCIN's reasoning sometimes requires simple common sense knowledge about the world. For example, one shouldn't ask if a male is "pregnant or breastfeeding," and there is probably a (realistic) minimum age for being an alcoholic. Thus, we say that the sex of the patient screens "motherhood" and age screens "alcoholism." Unfortunately, this knowledge is not explicit in the MYCIN system; proper performance depends solely on the ordering of clauses in rules that use parameters like motherhood or alcoholism. The "alcoholic rule" (rule535) illustrates this design principle. Clause 4 will only be considered if the patient is older than 17.

RULE535

```
If:  1) The infection which requires therapy is meningitis,
     2) A: A smear of the culture was not examined, or
        B: Organisms were not seen on the stain of the culture,
     3) The age of the patient is greater than 17 years,
     4) The patient is an alcoholic, and
     5) The type of the infection is bacterial
Then:  There is evidence that the organisms (other than those seen on
          cultures or smears) which might be causing the infection is
          diplococcus-pneumoniae (.3) e.coli (.2)
```

Figure 5.6. Rule that Uses Implicit Screening

It is important for GUIDON to know about these relationships so that it can explain, for example, why MYCIN never considered whether the patient was an alcoholic ("because he is not older than 17"). To make these relationships explicit, the knowledge base has been augmented by adding a relationship "P1 screens P2"

and the inverse "P2 screened-by P1."[1] Specific tutorial strategies for using this information are given in Section 6.2.2.1.

Explicit Screening Relationships: blocks of data

Blocks of data are a second form of parameter relation. When MYCIN was expanded to deal with meningitis infections, the number of questions asked of users increased to over 100 and it became useful to group questions into tables for more convenient entry of data. For example, one table contains the "findings from the csf." The property list representation of this table is shown in Figure 5.7.

(MYCIN consultation excerpt)

```
Please enter CSF findings in the following table

     CSF        Percent      Protein        CSF          Peripheral
     WBC        PMN                         Glucose      Glucose
 27)**  180     99           266            5            UNKNOWN
```

(part of MYCIN data structure)

```
CSFFINDINGS  [PROP-BLOCK]

TABPARMS:  (CSFCELLCOUNT CSFPOLY PROTEIN CSFGLUC BLOODGLUC)
TRANS:  (the "results of tests performed on the CSF")
```

(added in GUIDON)

```
SCREENED-BY: LP
```

Figure 5.7. Consultation Excerpt and Data Structure for CSF Findings Block

It turns out that there is a screening parameter for each table, e.g., the csf findings are available if and only if a "lumbar puncture" has been performed on the

[1] Like other information added to the system, these pointers currently reside on a special disk file. We might modify the EMYCIN representation package so that all MYCIN-like systems routinely incorporate this information in the future.

patient. Thus, for GUIDON's purposes, we have added SCREENS and SCREENED-BY relations for entries in the table. (E.g., lumbar puncture screens csf glucose, one of the csf findings.) The protocol (page 54) illustrates GUIDON's use of data tables; further discussion of their tutorial significance appears in Section 7.1.1.

5.1.3 Kinds of Rules

There are three primary syntactic kinds of rules: iterative, tabular, and definitional. These are domain-independent classifications that GUIDON can discern by using templates to read the rule. We describe them here because: (1) *iterative* rules pose special problems for preparing the complete AND/OR tree, and require special tutorial rules for discussing them; (2) *tabular* rules conveniently structure the rule set, and it is easy to generate interesting questions from them; (3) *definitional* rules are given special treatment in tutorials because they seem to be easy to state and remember.

Iterative Rules

Of MYCIN's 450 rules, 25% have a precondition that evaluates a predicate over a set of objects. For example, "the cutaneous site rule" (rule 146) [1] shown below, requires consideration of all of the organisms isolated from positive cultures of the patient. The iteration predicate (clause 3) performs a subset operation: it returns the members of the specified object set that satisfy the internal premise (A, B, and C).[2]

---

[1] Named with irony, for the list of sites is not so described in the rule itself.

[2] In general, iterative predicates set a global variable which is passed on to other preconditions or to the action part of the rule. In this example it is COLLECTEDORGS.

Besides the subset predicate, there are two other kinds of iterative preconditions: operators to determine the existence of a set and operators to find the maximum (minimum) element of a set (for numeric-valued parameters). Appendix A describes how preconditions of this type complicate the process of configuring a complete AND/OR tree from the traces of a consultation. Section 6.2.1.1 discusses some of the problems of teaching rules of this type.

```
RULE146
phrase
  If:  1) The infection which requires therapy is meningitis, and
       2) Organisms were not seen on the stain of the culture, and
       3) You have examined the organisms isolated from positive cultures
          obtained from the patient, selecting those for which
          A: The site of the culture is one of: breast-discharge burn-site
             cutaneous-ulcer external-ear iv-site rectal-fissure surg-wound
             trauma-wound, and
          B: The identity of the organism is known with certainty, and
          C: There is significant disease associated with this occurrence of
             the organism
  Then:  There is weakly suggestive evidence (.2) that the identity of each
         of the organisms that you selected is the organisms (other than
         those seen on cultures or smears) which might be causing the
         infection

 PREMISE:  ($AND (SAME CNTXT TREATINF MENINGITIS)
                 (NOTSAME CNTXT SPECSTAIN)
                 (THEREARE (GETALL KNOWNORG)
                           ($AND (SAME FREEVAR SITE
                                       (ONEOF BREAST-DISCHARGE BURN-SITE
                                              CUTANEOUS-ULCER EXTERNAL-EAR
                                              IV-SITE RECTAL-FISSURE
                                              SURG-WOUND TRAUMA-WOUND))
                                 (DEFINITE FREEVAR IDENT)
                                 (SAME FREEVAR SIGNIFICANCE))
                           NIL COLLECTEDORGS T))
 ACTION:  (TRANSDIFPARM COLLECTEDORGS IDENT CNTXT COVERFOR 200)

 COMMENTS: This rule uses information about nonsterile cutaneous sites to
    determine what might be causing the meningitis.  (See also Rule 158.)
 Created: 1-JUL-77 12:29
 Last edited: 20-SEP-77 14:54
```

Figure 5.8. An Interative Rule: The Cutaneous Site Rule

Tabular Rules

About 10 percent of MYCIN's meningitis rules have a table for computing the conclusion of the action part. "The CSF Glucose Rule" (Rule 502) is typical (Figure 5.9). Notice that the table indexes the certainty factors for a list of conclusions, based on the range of the csf glucose value. Rules of this kind were placed in the system in order to collapse the rule set; the rule authors and system maintainers found this representation to be more convenient than using a separate rule for each range of the csf glucose value.[1] The use of tabular rules is an important means of structuring the knowledge base, and the organizational benefit is at least as important for tutoring the rules, as for the original acquisition process.

It is fairly easy to generate multiple choice and fill-in questions from tabular rules. Three examples are shown in Figure 5.10 . Preceding each example is the specification given to the question-generating routine. Appendix B describes the process of generating questions from tabular rules. One of the complications is that "incorrect conclusions" cannot simply be chosen randomly from the conclusion table--the program must be sensitive to what is a *meaningful* change in the correct certainty factor.

[1] Tabular rules were developed by Lawrence Fagan for the PUFF system. In PUFF, 15 percent of the rules are tabular.

RULE502

```
If:  1) The infection which requires therapy is meningitis,
     2) A lumbar puncture has been performed on the patient,
     3) The csf glucose value is less than or equal to 40, and
     4) A: The simultaneous blood glucose value is not known, or
        B: The csf glucose value divided by the simultaneous blood glucose
           value is less than or equal to .4
Then:  The type of the infection is as follows:
       If the CSF glucose value is:
       a) less than 5 then: bacterial (.4), not viral (.4), not fungal
          (.2),
       b) between 5 and 10 then: bacterial (.35), not viral (.35),
       c) between 10 and 15 then: bacterial (.3), not viral (.3), fungal
          (.3), tb (.3);
       d) between 15 and 20 then: bacterial (.25), fungal (.25), tb (.25);
       e) between 20 and 40 then: bacterial (.2), fungal (.2), tb (.2);

PREMISE:  ($AND (SAME CNTXT TREATINF MENINGITIS)
                (SAME CNTXT LP)
                (LESSEQ* (VAL1 CNTXT CSFGLUC)
                         40)
                ($OR (NOTKNOWN CNTXT BLOODGLUC)
                     (LESSEQ* (FQUOTIENT (VAL1 CNTXT CSFGLUC)
                                         (VAL1 CNTXT BLOODGLUC))
                              .4)))
ACTION:  (CONCLUDET CNTXT (VAL1 CNTXT CSFGLUC)
                    '((LT 5 400 -400 -200 0)
                      (BT 5 10 350 -350 0 0)
                      (BT 10 15 300 -300 300 300)
                      (BT 15 20 250 0 250 250)
                      (BT 20 40 200 0 200 200))
                    TALLY TYPE '(BACTERIAL VIRAL FUNGAL TB))
```

Figure 5.9. A Tabular Rule: The CSF Glucose Rule

```
PREMISE: multiple choice
ACTION: actual conclusion

Given the fact that the simultaneous blood glucose value is not known, what
value for the CSF glucose value would tell you that the type of the
infection is bacterial (.35) viral (-.35)?
Please choose one of the following (1 to 5):
       1) less than 5
       2) between 5 and 10
       3) between 10 and 15
       4) between 15 and 20
       5) between 20 and 40
   ** 4

No, because the CSF glucose value is between 5 and 10 we can conclude the
type of the infection is bacterial (.35) viral (-.35) [RULE502].
If it were true that the CSF glucose value were between 15 and 20 then one
could conclude that the type of the infection is bacterial (.25) fungal
(.25) tb (.25).
```

```
PREMISE: case facts
ACTION: multiple choice

From the CSF glucose value and the fact that the simultaneous blood glucose
value is not known one can conclude that the type of the infection is
(select from the following):
       1) fungal --  strongly suggestive evidence (.92)
       2) bacterial --  weakly suggestive evidence (.35)
       3) not viral --  weakly suggestive evidence (-.35)
       4) tb --  weakly suggestive evidence (.08)
   ** 2 4

The following value is correct: bacterial (.35).
The following value is not correct: tb (.08).
The following value is missing from your answer: viral (-.35).
(Refer to RULE502.)
```

```
PREMISE: case facts
ACTION: incorrect conclusion

Given the CSF glucose value and the fact that the simultaneous blood glucose
value is not known, does this tell you that the type of the infection is
bacterial (.4) viral (-.4) fungal (-.2)?
** YES
No, given these facts, we can conclude that the type of the infection is
bacterial (.35) viral (-.35) [RULE502].
The conclusions in the question hold when the CSF glucose value is less than
5.
```

Figure 5.10. Three Sample Questions Generated from a Tabular Rule (shown in Figure 5.9)

Definitional Rules

Some of MYCIN's rules state common sense relations that shouldn't need to be explained to a student. An example is the rule (paraphrased) that "the duration that a drug was prescribed is the time elapsed from when it was started to when it was discontinued." At first, it appeared that GUIDON could recognize all such rules by presence of a definite CF (1.0) in a rule's action. That is, these rules are *definitional*. However, it soon became clear that a rule is definitional to a student only if he has the necessary background knowledge to *support it* "with certainty." Moreover, there is such a thing as "definitional intent" that has little to do with how certain the rule author was about the strength of evidence--a somewhat unexpected distinction.

We concluded that there are three kinds of rules that determine values with certainty:

> 1) ***Common sense rules*** that the man-on-the-street should know, e.g., "if you didn't find organisms on the culture when you looked under the microscope, then there are no organisms to identify."
>
> 2) ***Basic medicine rules*** that we can expect a second year medical student to know (though we may find it useful to make finer distinctions) e.g., "if the route of administration for the drug was oral, and the drug is one that is poorly absorbed in the intestinal tract, then the dose of the drug received by the patient is inadequate." Most definitional rules are of this type.
>
> 3) ***Intent rules*** that advocate a definite therapeutic choice of action, e.g., "if the patient is less than 8 years old then don't prescribe tetracycline." (Rules like this frequently appear to make a great leap from premise to conclusion. The problem of explaining them to a student is considered in Section 5.2.)

Originally, we had hoped that the student model could be initialized by assuming that every student would know every definitional rule. Given the findings outlined above, it became necessary to examine each rule individually. (The process of setting up the *a priori component* of the student model is discussed further in Section 8.2.1.) Nevertheless, we found it useful to write special discourse procedures for dealing with definitional rules because there are so many of them and they seem to be easily remembered by new students. Section 6.2.1.2 describes these discourse procedures.

5.1.4 Rule Clusters

Rule Models

Rule models [Davis, 1976] are program-generated patterns that represent typical clusters of factors in rule preconditions for determining a given parameter. Rule models do not necessarily correspond to domain concepts, though in so far as correlation of factors is an indication of importance, they usually represent general principles that are part of the human expert's organization of his knowledge. An example is shown below; it indicates that the name of the infection (TREATINF) and the subclassification of the infection (TYPE) tend to appear together in rules for determining the organisms that should be treated for, based on clinical evidence alone.[1]

[1] These patterns are simply pairwise correlations between all parameters that appear in a rule group. The numbers are computed by summing the certainty factors that appear in rule conclusions, and thus reflect a measure of frequency of appearance and strength of evidence [Davis, 1976]. The entry designated by an arrow (-->) was used by GUIDON in the sample interaction when presenting *orientation* for pursuing this goal (page 47).

```
COVERFOR-MOD
P-ADVICE: ((TREATINF SAME 40000)
           (TYPE SAME 22500)
           (SPECSTAIN NOTSAME 13400)
            ((TREATINF SAME) (TYPE SAME) 40000)
            ((TYPE SAME) (TREATINF SAME) 22500)
           ((SPECSTAIN NOTSAME) (TREATINF SAME)
            (TYPE SAME) (EXAMSTAIN NOTSAME) 13400)
     -->    ((PENDCUL SUBJECT) (TREATINF SAME)
            (TYPE SAME) (SPECSTAIN NOTSAME)
            (EXAMSTAIN NOTSAME) 10900)
           ((NONCUL SUBJECT) (TREATINF SAME)
            (TYPE SAME) (SPECSTAIN NOTSAME)
            (EXAMSTAIN NOTSAME) 9700)
                              .
                              .
                              .
A-ADVICE: ((COVERFOR CONCLUDE CONCLUDE* 42000)
           ((COVERFOR) (PENDCUL) (NONCUL)
            (SMEARORG) (KNOWNORG) 42000)
           ((PENDCUL SUBJECT)
            (COVERFOR CONCLUDE CONCLUDE*) 10900)
                              .
                              .
                              .
```

Figure 5.11. Rule Model for the Parameter COVERFOR

Because rule models make explicit the factors that most commonly appear in rules for pursuing a goal, we are using them as a form of *orientation* for beginning students (Section 6.2.3.1). For GUIDON's purposes, rule models have been altered to include the context of application of the rule, namely the kind of object in the context tree to which the rule applies, e.g., "pending cultures."[1] For reasons that will become clear below, this provides a much more useful analysis of patterns in the rules.

[1] This is done by simply including a rule's *subject of application* as a precondition (which it is in practice), with the "predicate" SUBJECT. This extension to Randy Davis' rule model creation code was made in collaboration with Jim Bennett.

Rule Schemas

A rule schema is a description of a kind of rule: a pattern of preconditions that appears in the premise, the goal concluded about, and the context of its application. Unlike rule models, these schemas are given to the program maintainers by a physician expert. An example is shown in Figure 5.12; notice that provision is made for relating analogous schemas.

```
COVERFOR.CLINICAL  [PROP-RULE.SCHEMA]
-----------------
ANALOGOUS-SCHEMA:  (COVERFOR.LABDATA)
CF-COMPARISON:  GREATER
IMPORT:  ("When trying to find out the organisms that
          require therapy, give substantial weight to
          clinical evidence when there is no (hard) lab
          data available")
PARMS-SCHEMA:  (TREATINF EXAMSTAIN SPECSTAIN TYPE)
PREM-SCHEMA:  ($AND (SAME CNTXT TREATINF MENINGITIS)
                    ($OR (NOTSAME CNTXT EXAMSTAIN)
                         (NOTSAME CNTXT SPECSTAIN))
                (SAME CNTXT TYPE BACTERIAL))
SUBJ-SCHEMA:  NONPOSRULES
UPDATE-PARM:  COVERFOR
```

Figure 5.12. Rule Schema for the Parameter COVERFOR

Schemas were introduced to the system when it was observed that the expert had written large sets of rules by working from a template or pattern. Each rule in the set had a fixed set of introductory clauses and one or more factors usually specific to that rule. Using the template (rule schema), GUIDON can "subtract off" the rule preconditions common to all rules of the type, leaving behind the factors that are specific to the particular rule (this is done dynamically, during the tutorial). These are called the **key factors** of the rule. Thus, the key factor of d-rule 578 (see Figure

5.2), "the patient was seriously burned," was determined by removing the contextual information of the name of the infection, whether organisms were seen, and the type of the infection.

When computing the key factors of a rule, GUIDON also removes restrictions specified by screening relations (Section 5.1.2) and exception clauses which are generally present to screen for a condition that is dealt with in a special rule (e.g, if the patient has a head injury, default information based on the age of the patient alone is not useful (Figure 5.13)).

RULE507

```
If:  1) The infection which requires therapy is meningitis,
     2) A: A smear of the culture was not examined, or
        B: Organisms were not seen on the stain of the culture,
     3) The type of the infection is bacterial,
     4) The patient does not have a head injury defect, and
     5) The age of the patient is known
Then: The organisms (other than those seen on cultures or smears) which
        might be causing the infection is as follows:
      If the age of the patient is:
      a) less than 10 days then: e.coli (.75), klebsiella-pneumoniae
         (.5), streptococcus-group-b (.75),
      b) between 10 days and 1 month then: e.coli (.75), klebsiella-
         pneumoniae (.5), streptococcus-group-b (.75),
      c) between 1 month and 2 months then: e.coli (.75), streptococcus-
         group-b (.75), listeria (.4);
      d) between 2 months and 1 year then: diplococcus-pneumoniae (.41),
         streptococcus-group-b (.4), hemophilus-influenzae (.75),
         neisseria-meningitidis (.4),
      e) between 1 year and 5 years then: diplococcus-pneumoniae (.41),
         hemophilus-influenzae (.75), neisseria-meningitidis (.4),
      f) between 5 years and 15 years then: diplococcus-pneumoniae (.75),
         hemophilus-influenzae (.73), neisseria-meningitidis (.74),
      g) between 15 years and 55 years then: diplococcus-pneumoniae
         (.75), neisseria-meningitidis (.74),
      h) greater or equal to 55 years then: staphylococcus-coag-pos (.4),
         streptococcus-species (.4), diplococcus-pneumoniae (.75),

SCHEMA: COVERFOR CLINICAL
EXCEPTIONS: HEADINJDEF
DUAL:  RULE554
```

Figure 5.13. The Age Rule and its Schema-related Properties

Examples of the use of key factors occur throughout the sample protocol, e.g., on page 76, when GUIDON says, "...considering whether J.Smith has a head injury defect [RULE607]."

As shown in Figure 6.12, there are two COVERFOR rule schemas that are duals. Dual schemas describe two classes of rules which have the property that each rule in one class has an analogue (dual) in the other. Making the relationship between the classes in rules explicit (in the schema descriptions), would make it possible for the tutorial program to form analogies between specific rules. For example, after the tutor and student have discussed rule 607, the tutor could ask the student what conclusions he would make if he had laboratory data, as well as clinical information for making a diagnosis, thus asking for the student to state the specific rule for an analogous situation. Under this notion of analogy, two specific situations are analogous if the classes to which they belong (that is, their templates) are analogous.[1] Observe that this kind of knowledge, meta-knowledge that makes rule relationships explicit, enables a tutor to teach general concepts as well as specific inferences. We intend to implement strategies for using dual rule schemas in later versions of the system.

The duals discussed here are contextual; other duals, much more difficult to detect by inspection, involve exception clauses. For example, there is a rule that mentions cancer cases (i.e., Lymphoma and Leukemia) that is a dual to a rule that restricts itself to non-cancer cases. The second rule mentions steroids, but the first

[1] Using the same notion of analogy, Goldstein has recently added links between analogous rules in the skill network for the WUMPUS game [Goldstein, 1978].

does not because the rule author assumes that a cancer patient will be receiving steroids. Thus, this pair of rules implicitly considers the effect of steroids with and without cancer. (See "programming with rules" in Section 5.1.5.)

Subgoal Structure

Another way to organize the rules is to make explicit the structure of subgoals. This construct may provide a valuable foundation for conveying the rule base to students, for the general outline of MYCIN's AND/OR tree corresponds quite well to the approach that is often recommended to medical students (for example, see [Harrison's, 1974] [Yu, et al., 1978]). Figure 5.14 illustrates the subgoal structure of MYCIN's rules.[1]

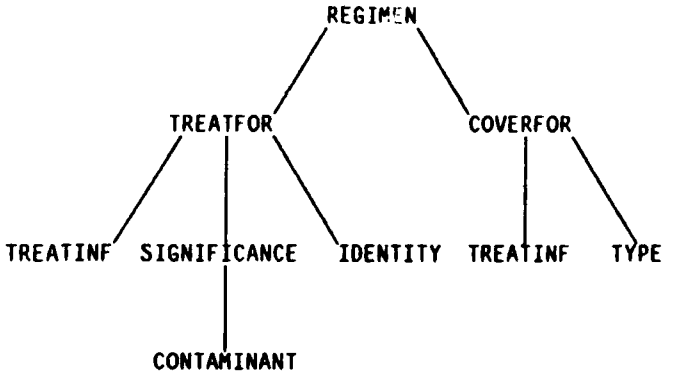


Figure 5.14. Subgoal Structure of MYCIN's Rule Set

GUIDON conveys this structure to a student in three ways during a tutorial:

1) A *sketch* of the subgoal tree (generated by a procedure from the AND/OR tree) lays out the "most important" steps for reaching a

[1] Bennett has called such a construct an *inference structure* and has described its importance in knowledge acquisition [Bennett & Engelmore, 1979].

diagnosis in a particular case (see Appendix D and Section 6.2.3.1).

2) Rule models reveal one level of the tree; they are presented when discussion of a topic begins and are independent of any particular case (see orientation strategies, Section 6.2.3.1).

3) Strategies for assisting the student single out subgoals that haven't been completed and suggest them to the student. (See page 76 and Section 6.2.3.2 for discussion.)

5.1.5 Rule Interactions

In this section we will consider the effects that rules have on one another. The main thrust will be to describe how the rule formalism has been used to encode knowledge to get a "performance effect," yet no meta-knowledge annotates these decisions. Consequently, the tutor is unaware of many subtle nuances that lie behind choices like the certainty factor of a rule's conclusion, whether or not a rule is "self-referencing" (concludes about a parameter mentioned in the premise), and how a rule's dual is encoded. We have dealt with some of these problems by adding meta-knowledge to the system, in other cases only a fixed-text justification annotates the rule.

The Cascading Certainty Factor Problem

To compute the certainty of the conclusions in a rule's action, MYCIN multiplies the minimum certainty of the set of preconditions (passed as the variable TALLY) by the certainty factor (CF) that appears in the rule's action. This CF is then combined with those derived from other rules that successfully update the parameter; see Figure 5.15.

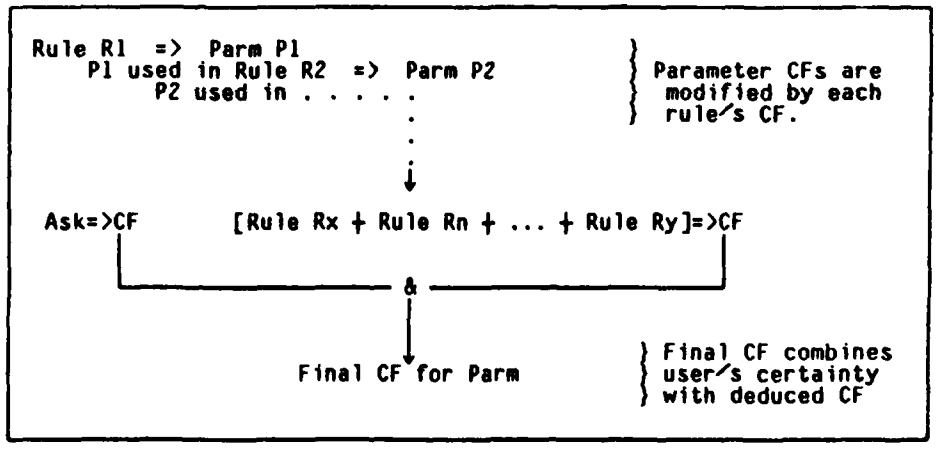


Figure 5.15. CF Combinations in the AND/OR Tree

For the resultant conclusion (the combination of all CFs in rules that succeed) to be considered as *evidence*, it must exceed .2. For example, if a rule precondition states "the morphology of the organism is rods," the CF associated with rods must exceed .2 for this precondition to succeed. The cascading problem arises when belief is handed down in a hierarchy of rules: As CFs are computed at each level, the CF being pushed to the top of the tree becomes smaller and smaller because each level multiplies the result by a number less than 1. See Figure 5.16.

level	Parameter	Minimum Rule CF	Minimum Conclusion CF	CF CUTOFF
1	TREATID	.4	.5 x .4 = .2	.2
2	COVERFOR	.7	.7 x .7 = .49	.5
3	TYPE	.7	.7	.7

Figure 5.16. Table of CF Thresholds in the AND/OR Tree

This table illustrates that, as we descend into the AND/OR tree, rules must use higher certainty factors in order for the result at the top of the tree to exceed the .2 cutoff. Specifically, a rule that concludes COVERFOR must use CFs greater than .7 ("minimum rule cf") so that when TALLY reflects the cumulative CF for TYPE (the "minimum conclusion cf," also .7), the result will be greater than .49. Then, when the cumulative CF for COVERFOR is at least .6, and it is used in TREATID rules having a rule CF greater than .4, the final outcome will be greater than .2, the cutoff for "significant evidence." Put another way, if you're writing a rule that concludes that Pseudomonas should be treated (COVERFOR pseudomonas), then you'll need to use a CF greater than .7 in the rule, to allow for the reduction by TALLY coming from TYPE and the expected change higher in the tree when COVERFOR is reduced by the TREATID rules.

The column on the far right, "CF CUTOFF," is the CF threshold for cumulative evidence on each level. It is used by GUIDON to provide commentary to the student to inform him about the changing interpretation of CFs throughout the tree. For example, if the tutor and student have just gathered evidence to support the hypothesis that Pseudomonas should be covered for, and the cumulative CF is below .6, GUIDON will tell the student that the cumulative evidence is not yet significant (analogous to what occurs on page 76).

Programming With Rules

Sometimes single rules in isolation may appear nonsensical even to another expert, but when their effect is combined with other rules, the consultation system performs correctly. Some examples are given here.

1) The ***fungal rules***. There are basically two fungi that cause fungal meningitis; the rule set is designed so that a single rule always posits that both fungi are present and other rules "discredit" one fungus when there is evidence for the other. For example, if the patient was in an area that is endemic to coccidiomycoses, then cryptococcus (the other fungi) is concluded with a negative CF.

2) The ***partial treatment rule***. When a patient with bacterial meningitis has been receiving antibiotics in the period before the consultant was called in, the CSF findings are shifted (e.g., the CSF glucose value increases), so that the evidence is not as strong that the infection is bacterial. Consequently, rules that use the CSF findings will be fooled into concluding "not bacterial," while in fact the patient is simply recovering from his bacterial infection.

Rather than using the fact that the meningitis is partially treated as contextual information to modify each rule that interprets CSF findings, the rule authors chose to write a single rule that says, "if the meningitis is partially treated, then increase your belief that the infection is bacterial." Thus, this rule adjusts the certainty of a conclusion to get the right effect. However, in isolation the rule makes no sense: A naive student might conclude that antibiotics somehow *cause* bacterial meningitis (!).

3) The ***x-Ray Rules***. Rules can be implicit duals of one another, without any indication in the knowledge base that complimentary situations are being considered. For example, there is a rule that says that when the chest x-ray is normal and the patient does not have a tb risk factor, then don't worry about a TB infection. Another rule says that an abnormal chest x-ray that suggests Active-tb is reason for considering TB.

These two rules are interesting because one is (almost) the inverse of the other. Here a simple idea has been split into two parts, with no trace of the connection to be found. When a tutorial program looks around for interesting questions to pose to a student, it needs special heuristics to find pairs of rules like this. (Suppose the x-ray is normal and there is no tb-risk, the tutor might want to ask, "If the x-ray were abnormal and indicated active tb, what organism might be causing the infection?") A similar relationship exists among the rules for concluding whether a culture was taken from a sterile source in the body. The rule authors designed the set of eight rules so that they are disjoint: exactly one rule succeeds in every case.

There appears to be a choice here: write heuristics for "discovering" these patterns by analyzing the rule base (preferably one set of modular functions or rules for use by the knowledge acquisition, explanation, and tutorial programs), or make the relationships explicit in the representation.

4) ***Self-referencing Rules.*** The only way to control the order that rules are applied in MYCIN is to make some rules "self-referencing." These are rules for determining a parameter that refer to the parameter being concluded about. For example, the rules for concluding the name of the infection (meningitis, bacteremia, etc.) are broken into two groups: (1) normal rules that seek to discredit the default conclusion, and (2) "self-referencing rules" that make a default conclusion in the absence of negative evidence from the first group (for example, one self-referencing rule states that "if there is no evidence that meningitis is not the infection and the site cultured is CSF, then assume that the infection is meningitis").

Interactions like this require the tutorial program to be sensitive to whether or not a rule is self-referencing. In particular, no self-referencing rule for a parameter can properly be considered for discussion until all of the normal rules for this parameter have been evaluated. The overlay model (Section 8.3.2) enforces this distinction, however the tutor sometimes finds it convenient to ignore normal rules that fail to apply to the particular case, and this generally causes no difficulties.

In summary, rules do not exist in isolation to one another; rule authors were often quite aware of interactions that ensured correct results. None of these interactions are explicit in the MYCIN knowledge base. We have given GUIDON some ad hoc meta-knowledge about this structure, but most of it can only be discovered and understood by reading the fixed-text comments and justifications written by rule authors. It is perhaps inevitable that every knowledge representation we use will have an accompanying folklore for its use. The challenge is to make this information explicit, and to keep its need to a minimum.

5.2 Support Knowledge -- for explaining rules

One MYCIN rule says, "If the patient is less than 8 years old, don't prescribe tetracycline." As a dictum, this is clear enough, and it indeed makes MYCIN perform properly when prescribing drugs for youngsters. However, unless you know about chelation, the process of drug deposition in developing bones, and the blackened permanent teeth that result, you won't be able to understand why this rule is medically (and socially) correct. One of the points we will develop here is that after you have understood a number of rules like this, you begin to form an intuition, or model for the kinds of arguments that lie behind rules of this type. So after learning a few of of MYCIN's drug contraindication rules, like the dictum about Tetracycline, you will find it natural to group them under some rubric like "undesirable body changes of antibiotics that depend on patient-specific factors." Accompanying this concept will be the notion of a *process*, in this case telling the story of drug administration by some method, migration of the drug in the body, chemical effects along the way, deposition, undesirable body changes, and so on. We claim that mechanistic stories like this form the foundation for satisfying explanations of MYCIN's rules: they support the leap from premise to conclusion.

The subsections below characterize MYCIN's rules as *optimized knowledge* (Section 5.2.1) and then categorize the forms of knowledge that are not expressed in the rules, namely principles of abstraction, inference, and mechanism (Section 5.2.2). Implications for tutoring are considered (Section 5.2.3). Mechanism descriptions are introduced as a form of meta-knowledge that GUIDON uses to explain rules (Section 5.2.4). It is argued that underlying process

models appear to be a form of expert knowledge that is orthogonal to the set of performance rules, posing some strict limitations on GUIDON's ability to discuss and relate individual rules when these models are not available.

5.2.1 What's in a Rule?

The most serious hindrance to using MYCIN's rules for teaching is that significant reasoning steps are "missing" from nearly every rule. Put simply, MYCIN's rules represent *optimized knowledge*. Domain relations that clinch the inference argument of the rule have been "compiled out"; they are not explicit anywhere in the knowledge base. If a student does not know these relationships or has forgotten them, then he will not be able to make sense of the rule. However, the expert who wrote the rule did not find it necessary to include these steps because evidently their applicability does not vary from case to case. That is, there is no need to parametrize this knowledge--to make it dependent on other rules or to ask the program's user to verify that they hold.

For example, observe that chelation of drugs in bones depends on the age of the patient. The Tetracycline rule could have been written so that instead of asking for the patient's age, the program would have asked, "Are the patient's bones subject to chelation?" But MYCIN is designed for use by non-experts who are only expected to supply primary data about the patient. To avoid asking about chelation, we could have the program infer this condition from the patient's age.

Thus, we could have written MYCIN's Tetracycline rule in two parts:

(1) IF the patient's bones are subject to drug chelation,
THEN don't prescribe Tetracycline.

(2) IF the patient is less than 8 years old,
THEN the patient's bones are subject to drug chelation.

However, we only have one rule to determine whether chelation might occur (2), so it was convenient to collapse this argument into one step: if the patient is less than 8, don't prescribe Tetracycline. Imagine the presence of a third rule, say:

(3) IF the patient has received steroids in the past 3 days,
THEN the patient's bones are subject to chelation.

Now we might write

(4) IF the patient is less than 8 or
has received steroids in the past 3 days,
THEN don't prescribe Tetracycline.

And we have again "compiled out" the intermediate concept of chelation.

The point of this exercise is to illustrate that there are many ways to write rules so that the "right answer" (correct medical diagnosis and therapy) comes out. In fact, when the 400 rules were written for MYCIN, we found it convenient to keep the number of parameters in the system to a minimum, making it a *practice* to leave out intermediate concepts (like "chelation") wherever possible. For this reason, we claim that MYCIN's rules really only encode the input/output behavior of medical reasoning. Raw data feeds into the rules, raw conclusions come out (see Section 5.2.2.2), with only a small number of high-level intermediate concepts tying it all together. Consequently, in order to use MYCIN's rules for teaching, we must either

add more knowledge to the system, or rely on the student to fill in the missing steps, based on his limited classroom and clinical experience.

Since it seems implausible to expect a second year medical student to be able to figure out every MYCIN rule on his own, we have annotated the rules with fixed-text (not machine readable) justifications. Moreover, literature references cite published articles that provide further information (including original clinical studies) for illustrating the reasoning behind the rule.[1] Finally, to unveil difficult implementation problems, **comments** are also available, though they are usually only useful to someone with a thorough knowledge of MYCIN. At best they bring out misgivings that the rule author has about the adequacy of the rule for covering the concept he is trying to incorporate in the program.[2] Examples of these appear in Figure 5.2.

Finally, we should emphasize that the problem of missing reasoning steps is not necessarily solved by writing intermediate concepts and rules: We simply don't know offhand which concepts to include. Imagine expanding the Tetracycline rule by introducing three intermediate concepts: chelation, teeth discoloration, and undesirable body change:

[1] This information is stored in a hash file on disk, along with other rule-specific annotations.

[2] For example, the rule may be a gross oversimplification, using one or two crude measurements to check for a patient condition that would require dozens of finer considerations for precision. Thus, MYCIN concludes that a current therapy is not effective if the patient has not "responded" in 2 days. To do this, MYCIN asks the physician whether the patient has responded--though sometimes it takes an expert to make this judgement, involving examination of temperature charts, the typical course of the illness, and consideration of other drugs the patient is receiving. (See also the comments for the "clostridia rule" in Section 5.1.1.)

```
youngster  =>  propensity for chelation in growing bones
Tetracycline administration & propensity for chelation
      =>  teeth discoloration
teeth discoloration  =>  undesirable body change
undesirable body changes  =>  contraindication
```

The choice of steps or intermediate concepts is arbitrary, of course. For example, there is no mention of how the chelation occurs. What are the conditions? What molecules or ions are involved? Even if we tried to fill in all of the missing steps in MYCIN's rules, how would we choose the right level of abstraction to formalize the reasoning steps involved? If our purpose is good *performance*, the tetracycline rule as first stated above is quite adequate. If we wish to explain the rule to a student, we'll need some mention of bone and teeth involvement.[1] But if the purpose is to develop a system that can truly reason about its knowledge (e.g., to find analogies during a knowledge acquisition or learning process), all the links that a physician knows might conceivably be useful.

This suggests that we need two representations: one that represents compiled (or optimized) knowledge (as in current MYCIN rules), for performance, and another representation that details step-by-step reasoning from empirical data to final conclusions. Ideally, the compiled form would be generated from the detailed version, and for exotic cases the program would be able to reason on the detailed level (*careful mode* in the style of Sussman's program writer), looking for possible bugs in its cookbook (compiled) knowledge. Knowledge acquisition and tutorial programs would very likely benefit from both levels of representation.

---

[1] The careful reader might agree that teeth discoloration is certainly a critical thing to know, but he might wonder why this is satisfying. Section 5.2.4 characterizes satisfying explanations in terms of shared underlying models.

Does a MYCIN rule merely leave out invariable steps in the reasoning chain? Can/should we view rules as "looking down on" a network of some kind [Hart, 1975]? We are in no position here to suggest a possible representation for all of the physiological, empirical, and social knowledge that lies behind MYCIN's rules. It is extensive, that much is certain, and for the moment we will have to rely to a large degree on the justifications and literature references. Thus, GUIDON carries on a dialogue about the USE of MYCIN's rules, and cannot clarify the meaning of the inferences themselves. To see how the situation might be improved, we first study in more detail the kinds of knowledge that MYCIN's rules leave out.

5.2.2 Principles Missing from Optimized Knowledge

Examples below illustrate three kinds of principles that are not represented in the rules, but are nevertheless an intricate part of the support knowledge for understanding them. **Data abstraction principles** are concepts for interpreting case data: Rules tend to state the raw data directly. **Inference principles** are concepts for interpreting rule conclusions: Rules tend to state detailed values, without discriminating trends. **Mechanistic principles** are concepts that underlie the rules and serve to tie premise to conclusion (like "chelation").

5.2.2.1 Data Abstraction Principles

The first type of optimization we will consider is where an abstract concept for interpreting case data has been replaced in a rule premise by a specific reference to empiric data. Consider "the Leukopenia rule" (Rule 557).

RULE557

```
If:  1) The infection which requires therapy is meningitis,
     2) A: A smear of the culture was not examined, or
        B: Organisms were not seen on the stain of the culture,
     3) The type of the infection is bacterial,
     4) The results of a recent CBC of the patient are available, and
     5) The white count from the patient's peripheral CBC (in thousands)
        is less than 2.5
Then:  There is evidence that the organisms (other than those seen on
         cultures or smears) which might be causing the infection is
         pseudomonas-aeruginosa (.5) e.coli (.75) klebsiella-pneumoniae
         (.5)
```

Figure 5.17. Leaving Out Data Abstraction Principles: The Leukopenia Rule

In terms of principles, a white blood count of less than 2.5 would be considered by a physician to be clear evidence of "leukopenia." Rather than adding a new parameter to the MYCIN system and another rule of the form "if the white blood count is less than 2.5 then there is evidence of leukopenia," the rule author "compiled" this knowledge into the rule.

Many other examples could be given of rules in MYCIN that skip directly from observation (case data) to long lists of conclusions without explicit representation of intermediate concepts for interpretation of the data. Part of the problem was that intermediate concepts may be useful for organizational constructs, but specific rules demand precise consideration of empiric details. Catch-all terminology might lead to broad, weak rules of little diagnostic value.

It turns out that in general a white blood count below 4.5 is called "leukopenia." [Harrison's, 1974] Thus, the author of the leukopenia rule evidently wanted evidence of "pronounced leukopenia." It is somewhat awkward to make this a parameter since it

would probably have no use elsewhere in the system, and if it did, we wouldn't be surprised if our notion of "pronounced leukopenia" changed in different contexts.

Concept broadening is a more general problem, however, and there is a fairly straightforward remedy. Once a physician included the pre-existing MYCIN concept "compromised host" in a new rule that he was writing. The problem is that he meant something more specific ("acute leukemia with granulocytopenia induced by chemotherapy") than the grounds MYCIN would use for deducing this condition, e.g., simply the fact that the patient is undergoing chemotherapy. The remedy is to include both the general concept and its empiric refinement in the rule, and perhaps annotate the clauses to indicate this.

Applying this technique to the Leukopenia rule, clause 4 could then be translated, "the WBC is less than 2.5, i.e., leukopenia." However, now the rule set would no longer be optimized, for the precondition "leukopenia" and the rule that deduces it are of explanatory benefit only.[1] In Section 5.2.4 we show how the explanatory power of intermediate concepts like "compromised host" and "leukopenia" lies in their relation to an underlying model of disease. There we suggest that it may be impractical to structure the performance knowledge so that it expresses this model; instead we may need to maintain separate representations.

[1] Fagan believes that rules that use numerical bounds are easier to understand if they refer to measurements with respect to the norm, e.g., "if the white blood count is HIGH." He uses intermediate rules to make these distinctions in different patient situations in the Ventilator Management program. **[Fagan, et al., 1979]**

5.2.2.2 Inference Principles

The second characteristic of optimized knowledge is where inference principles have been replaced by a list of specific conclusions. To illustrate this, we return to the tabular rule shown in Figure 5.9. While tabular rules are tremendously useful for acquiring sets of implicitly related rules from experts, the representation falls considerably short of what GUIDON needs for conveying the principle of the rule. Imagine that you are a student, faced with the table of parameter ranges and certainty factors shown in Figure 5.9. How are you going to remember all of those numbers? Wouldn't it be better to know the principle behind that table instead? At the very least, we'd be better off if we had a handle on the trends or most significant conclusions for each range. Considerations like these suggest that we graph the table of conclusions to see if there is a pattern we can easily grasp and remember. The table for the CSF glucose rule is shown in Figure 5.18.

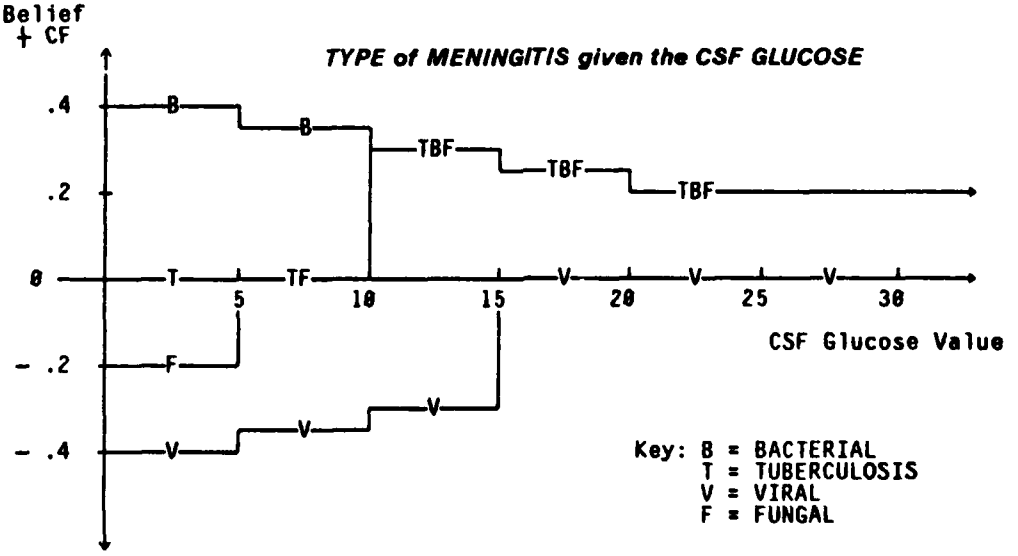


Figure 5.18. Graph of the Conclusions Made by Rule 502 (Figure 5.9)

The graphical representation brings out the structure of the table. For example, the evidence for bacterial meningitis is inversely proportional to the csf glucose value; on the other hand, a glucose value below 15 tends to rule out viral meningitis. Furthermore, somewhere between 5 and 10 an important discrimination can be made: above 10 there is no way to differentiate between tb, bacterial, and fungal meningitis using the csf glucose value, while a value less than 10 yields no evidence for tb or fungal but strongly supports bacterial.

To recap, using this chart alone, we derived principles that are precisely what we want to teach a medical student.[1] But notice that MYCIN has no grasp on the concepts "low value" or "discrimination point." So how is GUIDON to be sure that the student understands what is essential (worth remembering) about this rule? Quite simply, the program has no way to do this because it has no handle on the principles involved.

We are led again to observe that MYCIN's rules constitute "compiled" performance knowledge. (Think of the table as the "assembly language" form of the abstractions we have stated above.) However, our observations go beyond the problem of using performance knowledge for teaching. Imagine the difficulty that a trained physician would have in trying to convince himself that the csf glucose rule is correct: he will have to go through the very same analysis we produced above! When rules implicitly encode principles it will take a significant amount of intelligence to make them explicit again. Moreover, there is the danger that GUIDON will be too picky

[1] Of course students will use different techniques for committing this to memory. For example, the graph has a strong visual effect with tb, bacterial, and fungal marching down together.

about numeric ranges and long lists of certainty factors, and have no grasp on the higher level principles it should be trying to convey.

We have considered the possibility of writing a program that could analyze MYCIN's tabular rules, using general heuristics for asking questions to make sense of graphs. However, since there is every reason to believe that the original rule author knew the principles, we would be better off having him represent the principles explicitly in the first place.

5.2.2.3 Mechanistic Principles

We have already considered how MYCIN's rules lack a generalization level of data abstraction (premise) and inference (conclusions). We will now examine an underlying level of reasoning that ties premise and conclusion together. This underlying knowledge is often called a *mechanistic model*.

It is possible that the medical knowledge we are trying to codify is really on two levels of detail: 1) principles or generalizations, and 2) empiric details or specializations. Following the ideas of web tutoring [Norman, Gentner, & Stevens, 1976], we might claim that a student should be taught principles at first to set up a foundation or outline. Then when the student begins his clinical experience as an intern, he will learn to cope with many detailed variations of these principles. The conflict here is that principles are useful for getting an initial grasp on the material, as an orientation to understand kinds of data and inferences,[1] but they may be inappropriate for practical problem-solving.

[1] Assuming that the principles and concrete examples are presented together--the relationship is posited to be what provides leverage for memory.

For example, "consider the portal of entry of the organism into the body," is a principle based on a mechanistic model of infection that is taught to medical students,[1] but perhaps it is a poor concept to use in a performance knowledge base: rather than providing diagnostic clues, "portal" is very often deduced from the diagnosis itself. The following excerpt from a dialogue with a physician [2] illustrates this point:

****

Interviewer: Okay.... What do you think about "portal of entry"? Is that of any interest...?

Physician: ...if this is an uncomplicated, if this case is what it appears to be, which is a typical case of pneumococcal pneumonia, there wouldn't be any question in my mind as to the portal of entry... It's obvious: the patient has pneumonia, he has a pocket of bacteria in his lungs and that's obviously the way the bacteria got into the blood....

Interviewer: ...lung tissue...

Physician: Uh-huh, this question can be moot if the patient has MORE than one site of infection.

Interviewer: What COULD the portal of entry tell you? Dr. Y says that it's retrospective, it's something you know after you've already decided what the infection and the organisms are, and that doesn't help you conclude the identity of the organism or the name of the infection.

Physician: In this particular case it doesn't add anything. If you were dealing with a a case of, well, consider this case, young 20 year old patient has his teeth... Say he has rotten teeth, and he goes to a dentist and then 2 or 3 weeks later develops bacterial endocarditis, temperature of 105 degrees, very ill, and you have a coccus growing out of the blood. Then, portal of entry adds something. Because the typical organism that causes endocarditis under those circumstances is strep-viridans. Strep-viridans is normal flora in the mouth and a typical history of strep-viridans bacterial endocarditis is that of having dental work done. And so the fact that the portal of entry was probably the mouth DOES influence your decision about what the organism is.

Interviewer: So it's the OTHER history that told you the probable portal of entry...?

---

[1] "With relatively minor variations, the development of an infectious disease follows a consistent pattern. The parasites enter the body through the skin, nasopharynx,... or other portal, and a regular sequence ensues...." **[Harrison's, 1974]**, p. 724. Compare with the model we formalize in Section 5.2.4.3.

[2] The interviewer was simulating GUIDON and the physician was playing the role of a student. Thus, the physician was given only limited information about the case and needed to request more as necessary. One purpose of the interview was to study how the accumulating evidence changed the physician's understanding of the case.

Physician: Uh-huh, and that [hypothetical] case is different from this one in that you don't have an infection in the mouth [corresponding to the infection in the lungs]. [In the hypothetical case]...all you have is a history of work having been done in the mouth...
Interviewer: Right. I see. Things having to do with wound infection or something like this, is that useful sometimes...?
Physician: Ah... There are, I'm sure there...
Interviewer: Well, I'm probably getting into things I don't want to get into right now...
Physician: I'm sure that it might be useful...
Interviewer: ...trauma wounds, it seems like...
Physician: ...sure, sure, that might be useful.
Interviewer: ...it's used.... Okay, so let's start to wrap this up here.
Physician: You may not know what the portal of entry is I think typically, if you have several infectious sites in the body, you have a trauma wound on the hand, you have meningitis, a kidney infection, and those are the kind of cases that MYCIN will be asked to consult on. I don't see that the physician has any way of knowing. And I think that what Dr. Y means by retrospective decision is that if there's agreement between the bacteria growing in the blood and bacteria from one of those sites of infection, then you assume that the portal of entry was from that site of infection.

****

From this perspective, "portal" is a concept that a physician most often uses *after the fact* to form a complete picture that satisfies his model of the infectious disease process. However, consideration of "portal" only infrequently provides leverage for problem solving itself. This suggests that it may be difficult or impossible to expect a single set of MYCIN diagnostic rules to serve both as concise, "clincher" methods for efficiently getting to the right data, and still represent a model of disease. Put another way, a student may need the model if he is to understand new associations between disease and manifestations, but he will be an inefficient problem solver if he always attempts to directly convert that model to a subgoal structure for solving ordinary problems. More generally, the knowledge we use to justify what we know may be used only tacitly to solve the problem by providing feedback (meta-knowledge) that we are *getting it right*, that everything fits [Barr, 1979].

If we attempt to make our model of disease explicit in MYCIN's rules, we may find that the organizing structure gets in the way, as one physician has claimed about the parameter "portal of entry." It shouldn't be expected that general principles will be a good mold for constructing interpretations of every case. According to this view, principles are good for summarizing arguments, and good to fall back on when you've lost grasp on the problem, but they don't drive the process of expert reasoning.

Perhaps one good way to get a handle on this problem is to consider how MYCIN's purely goal-directed approach to diagnosis is entirely inconsistent with data-directed thinking. Indeed, when we analyze the different approaches used in MYCIN's rule set, we notice that there are kinds of subtrees: (1) deep, clear subgoal structure that reflects an approach for doing diagnosis, and (2) a shallow, broad layer of rules for triggering on specific observations. To a large degree, this difference reflects the differing styles of rule authors. One group tended to make concepts explicit, as can be seen in Figure 5.14, illustrating the basic subgoal structure for tackling a problem. The other group generally preferred to express their knowledge in "cookbook" form, going directly from observation to diagnosis instead of lumping observations into intermediate principles. This second group argued that they didn't reason from principles and it was difficult to approach a problem in this way. They even claimed that they would have "failed" if they attempted to break every problem down in a strict subgoal fashion that used explanatory principles like "portal of entry" and "compromised host."

However, it is possible that the two groups were dealing with different kinds of problems. The group that used intermediate concepts and formalized a subgoal

structure of some depth were dealing with hard data (for identifying an unknown organism given observed culture data). Here a few basic factors enable one to write solid rules that quickly latch onto the answer. (Indeed the rules for these intermediate concepts tend to be mutually exclusive.) When soft, clinical concepts like "compromised host" are considered, they don't play a central role in decision making.

On the other hand, the other group of rule writers, those that designed a shallow subgoal structure with many rules for a general goal, may have been confronted with a more nebulous kind of pattern-matching problem: determining an unknown organism from clinical data alone. There are many pieces of data available, but few hard and fast rules to relate them. The underlying theory recedes to the background, and the expert tends to approach his problems simply in terms of associations between observed data and bottom-line conclusions. This may have promoted a rule writing style that discouraged introducing intermediate concepts, even where they might have been appropriate, e.g., the concept of "leukopenia" described above.

5.2.3 Implications for Tutoring

The analysis above is partly speculation to account for the missing principles we observe in the MYCIN rule set. However, our immediate concern is to consider the implications of tutoring from this rule set. To the extent that MYCIN's rules are "empiric," that is, capture every practical consideration, they will be too detailed for conveying principles to students. As described above, MYCIN's rules are too detailed in three ways: 1) they check for many specialized cases without lumping the observations into general concepts (such as "high" and "low" values); 2) they

conclude a uniform list of possibilities that fails to distinguish between rare and expected outcomes (indicating inference discrimination points); and 3) they omit underlying abstract (mechanistic) concepts (like "impairment of host-defense mechanism"). Besides requiring the student to look for patterns in the rules to find the principles, detailed knowledge of this kind will inhibit GUIDON's ability to convey principles by appropriately selecting cases and weighing presentations.

A likely result is that GUIDON will attempt to teach the wrong level of knowledge. It might demand support for hypotheses on the level of detailed data, when the student may most need to know and remember underlying concepts and relations that medical science has abstracted from the observed data. Using MYCIN's rules alone, GUIDON might possibly teach how to conceptually subdivide a problem, but it could never provide explanatory principles that justify that subdivision.

5.2.4 The Use of Models for Explanation

The point of this subsection is very simple, though the exposition is long. We claim that satisfying explanations of MYCIN-like rules are couched in terms of models that experts use to convince themselves that there is sufficient evidence for some condition. Specifically, explanations of rules enable one to conclude that the critical events of a process have taken place, and so an end condition claimed by the rule is true. Not all critical events of the script are mentioned in the explanation or supported directly by the rule. Many are implicit and derived from default knowledge about the world.

The basic observation is that to write a set of judgmental rules to determine some parameter (a proposition that postulates evidence for some condition), you must have some notion of ***what constitutes sufficient evidence for that condition.*** Moreover, each time that you write such a rule, you instantiate that notion of sufficient evidence by your choice of premise clauses in the rule and the hypothesizes that you claim these clauses support. Evidence for a process is given in terms of evidence for critical events or states of the process. Rules that conclude that a process has taken place (or that a condition exists) mention factors that allow one to infer that the critical events have taken place or that the critical conditions hold. If you know the sequence of critical events, that is, the process model, and can infer that each has been enabled in a particular case, then you have evidence that the end condition has taken place.

Each rule is an argument that the end condition has taken place (with a particular certainty). If the process model is known to the student, then explaining a rule involves arguing that the premise provides evidence that the end condition holds. Default knowledge may be necessary, for MYCIN rules tend to refer to only one condition in the process.[1] In accord with this view of what a rule is, the rule set is a kind of mapping that relates observable phenomena in the world to the model of a process. New rules or rule modifications are designed to complete the coverage of observable factors to critical events that enable the process. For example, following the infectious disease model we present in this section, we could imagine that a

[1] Probably because of the assumption that case factors are independent, so they can be considered in separate rules--leading to weak rules with small certainty factors.

research physician will look for new rules to relate patient occupations to infection, knowing that exposure to an organism is a critical event in the infection process. It is this connection between an occupation and the notion of exposure that is not represented in MYCIN's rules.

To make our point about the use of models for explaining rules, we have chosen to justify MYCIN's 40 rules for diagnosing the cause of meningitis (roughly 75% of the system's rules that are specific to meningitis). The classifications presented below were derived by the author by examining these 40 rules; preliminary results were improved with the help of physicians. Development of the model proceeded in three phases: 1) determining the **key factors** of rules; 2) classifying the key factors by kind of evidence; and 3) abstracting, from the kinds of evidence, a model of what constituted *sufficient* evidence of infection.

5.2.4.1 The Structure of Rules: Determining Key Factors

Analysis of the 40 rules showed that many of them had certain clauses in common; from this we defined the rule schemas that are described in Section 5.1.4. These clauses set the context of the rule, e.g., "bacterial meningitis with no lab data available." Of the remaining clauses, some we classified most as *restrictions*, and the one or two that remained constituted the key factor(s) of the rule. The "petechial lesions" rule (rule 541) exemplifies this classification.

RULE541

CONTEXT:
1) The infection which requires therapy is meningitis,
2) A: A smear of the culture was not examined, or
   B: Organisms were not seen on the stain of the culture,
3) The type of the infection is bacterial,

KEY-FACTOR:
4) Petechial is one of the types of rash which the patient has,

RESTRICTIONS:
5) Purpuric is not one of the types of rash which the patient has,
6) Leukemia is not one of the diagnoses of the patient,
7) Lymphoma is not one of the diagnoses of the patient, and
8) The patient has not received cytotoxic drugs just prior to or during this infection

CONCLUSION:
There is weakly suggestive evidence (.3) that Neisseria-meningitidis is one of the organisms (other than those seen on cultures or smears) which might be causing the infection

Figure 5.19. Classification of Clauses in the "petechial lesions" Rule

A heuristic for detecting restriction clauses is that they are stated negatively, e.g., clauses 5-8 above. We have identified seven kinds of restriction clauses in MYCIN's rules:

1) **Data Screen** (a parameter that indicates whether a class of data is available, e.g., CSF results are available iff a lumbar puncture was performed)

2) **World Knowledge Relation** (relationships between age or sex and other factors, e.g., alcoholism is not considered in children)

3) **Refinement** (further specification of a factor, e.g., "neurosurgery" further specifies "surgery")

4) **Other Cause** (evidence that may account for the key factor, and so invalidate this particular rule, e.g., leukemia,

lymphoma, and cytotoxic may also cause a petechial rash, making it wrong to conclude that Neisseria is necessarily the cause (Figure 5.19))

5) **Default-overruling** (evidence that invalidates default conclusions, e.g., a head injury overrules the relevance of the patient's age)

6) **Distinguished Subset** (condition that, when it occurs in combination with the key factor, must be treated as a special case, e.g., hospital-acquired infections for new borns, purpuric rash in combination with petechial rash)

7) **Evidence Tightening** (additional, optional factor that shores up the key factor argument, e.g., Neisseria is seldom the cause of hospital-acquired meningitis, and the absence of an epidemic is double evidence that clinches the argument)

These restriction clauses are lumped together into three kinds of annotations in the current version of GUIDON: screening clauses, exceptions, and refinements. Using this knowledge, GUIDON can subtract off the context and restrictions to get at the key factor. Once the pivotal or key factor has been identified, it remains to determine how this is evidence for the conclusion, e.g., what does a petechial lesion have to do with Neisseria?

5.2.4.2 Classifying the Parameters by Evidence Source

Without prior medical knowledge, it was impossible to determine how the key factor was specifically related to the conclusion. However, it was possible to find regularity by classifying the parameters by the source of evidence. We discovered that the 35 clinical parameters in the 40 rules could be broken into three categories,

corresponding to (1) direct observations of disease, (2) history of the present illness, and (3) predisposing factors for disease. The final category is further broken into (a) exogenous factors that affect the body and (b) endogenous characteristics of the body that make it prone to disease. This classification is shown in detail in Figure 5.20. (Parameters appear in all caps; definitions are given in Appendix C for the interested reader.)

By examining the kinds of evidence used in the rules, we were a step closer to understanding the concept "clinical evidence of disease" that these rules deduce. However, this classification was insufficient, for it seemed to be orthogonal to the underlying understanding we were groping for. For example, a congenital malformation, an endogenous factor (III.B.1.b), obviously must be a predisposition of some kind--the same reason for considering alcoholism, an exogenous factor (III.A.2.b). What *role* do these conditions play that makes their effect analogous? What is the mechanism? The remarks labelled "intuition" represent our first attempt to relate the source of evidence to a description of the disease process.

I. **Indirect Observations of Present Causative Organism/Disease**
[*Intuition: existing disease may have a visible effect*]

A. Laboratory Tests
CBC WBC TBRISK (biopsy, ppd, chest-xray scar) CXRAB
B. Signs
OCNERVE RASHES SKININFECT
C. Symptoms
MUMPSYM

II. **History of Present Illness**
[*Intuition: recent diagnoses are evidence for current problems*]
CLINEV-TUBERCULOSIS OTITIS-MEDIA EPIGLOTTITIS

III. **Predisposing Factors**

A. Exogenous
[*Intuition: factors that have changed/challenged the body*]

1. Environment
[*Intuition: exposure is a prerequisite for some diseases*]
ANIMALEXP CROWD EPIDEMIC NOSOCOMIAL TBRISK (contact)

2. Body Infractions
[*Intuition: entry of organism or weakening of defenses leads to disease*]
a) Procedures
CNSRADIATE CARDIACSURGERY GU-MANIP
NEUROSURGERY SPLENECTOMY
b) Drugs
ALCOHOLIC CYTOTOXIC JUNKIE STEROIDS
c) Traumatic Wounds
BURNED HEADTRAUMA HEADINJDEF

B. Endogenous
[*Intuition: what the patient "brought" to this situation*]

1. Physical Characteristics
[*Intuition: some diseases are correlated with body type*]
a) Demographic
AGE RACE
b) Congenital
CNSMALFORM
c) Acquired

2. Past Medical History
[*Intuition: previous diseases change the body and/or persist*]
DIABETIC LYMPHOMA LEUKEMIA SICKLECELL

Figure 6.20. Preliminary Classification of COVERFOR Factors

5.2.4.3 Critical Events: Constructing a Model of Sufficient Evidence

Proceeding from the classification of evidence source, we formed a model of *sufficient evidence for disease*--what you need to know before you believe that some organism is causing disease. It is a description of the mechanism of infection, including mention of an agent and critical events that must take place for infection to occur. Thus, we have a representation of the COVERFOR concept--"organism (other than those seen on cultures or smears) that might be causing the infection." The critical events [1] of the description are derived directly from our shared scientific model of the infectious disease process, which generally describes infection in terms of war: an invading organism, port(al) of entry, protective white cells that fight infection, and so on.

The mechanism description sets constraints on the rules that will conclude COVERFOR by characterizing the kind of evidence that they must mention in their preconditions. The frame-like representation of this description is shown below. Particular parameters can refer to one or more slots; we say that they *satisfy* the slots. Explaining a rule consists of fitting the rule premise and action to the frame of the parameter it determines. That is, demonstrating that the rule satisfies all slots of the description.

[1] Our use of the term "critical events" comes from Stevens [Stevens & Collins, 1978] who used to term to relate scripts used in WHY to models of meteorological processes.

```
COVERFOR-FRAME

BUG-AGENT:  (PARM "implies infection by" ORGS)
BUG-ORIGIN:  (PARM "implies that the patient has been
              exposed to" ORGS)
PORTAL-PATHWAY:  (PARM "allows access of organisms from"
                  SITE1 "to" SITE2)
IMMUNOLOGY:  (PARM "implies impairment of the host-defense
              mechanism")
DIAGNOSIS-CAUSE:  (ORG "causes" PARM)
```

Figure 5.21. Mechanism Description for the Parameter COVERFOR

The COVERFOR mechanism description tells the story of an organism (BUG-AGENT) that is normally found at BUG-ORIGIN; it enters the body and/or proceeds to the infection site via PORTAL-PATHWAY, where conditions favor its growth (IMMUNOLOGY); thus, an infection is visible by certain symptoms and signs (DIAGNOSIS-CAUSE).[1] The factors given in Figure 5.20 are related to the frame by associating a property with each factor that instantiates a slot of the frame. The general reorganization is shown below:

slot	factor classifications
BUG-ORIGIN:	ENVIRONMENT
PORTAL-PATHWAY:	INVASIVE-PROCS, TRAUMATIC-WOUNDS
IMMUNOLOGY:	DRUGS, PAST-MED-HISTORY, PHYSICAL-CHARACTERISTICS
DIAGNOSIS-CAUSE:	OBSERVATIONS, PRESENT-ILLNESS

Figure 5.22. Association of COVERFOR Factors with Frame Slots

[1] This story suggests that we call these frames scripts [Schank & Abelson, 1975] [Stevens & Collins, 1978]. However, it is too soon to say whether frames for other parameters (e.g., CONTAMINANT, SIGNIFICANCE etc.) will have the same story-like character. For now, we will simply observe that the slots are a more-or-less ordered set of conditions.

Figure 5.23 illustrates how the COVERFOR frame can be used to "explain" a rule. For comparison, a physician's explanation for the same rule appears in Figure 5.24.

RULE535

The fact that the patient is an alcoholic allows access of organisms from normal flora of throat and mouth to lungs (by reaspiration of secretions).

Figure 5.23. GUIDON's Explanation of the "alcoholic rule" (Figure 5.6)

Physician: I'd be interested knowing if... if this patient were an alcoholic would be one of the things.
Interviewer: Why would that be useful to you?
Physician: Alcoholics, particularly skid-row bums, typically pneumonia is an occupational hazard of being a skid-row bum. They drink themselves into a stupor and then aspirate their own oral secretions and develop pneumonia. Also, there have been some studies to show that alcohol inhibits the cilliary motion of the respiratory tree and that would prevent clearing of aspirated bacteria, which would predispose to pneumonia.
Interviewer: In particular, it's the pneumonia that makes you curious about that?
Physician: A 59 year old male with pneumonia would make me wonder what the predisposing cause was. Pneumonia is not a terribly uncommon disease, but when it appears you wonder about the etiology--the conditions that set this particular 59 year old man up to get pneumonia. In fact, one of the things you'd do in following this patient after the patient is discharged, you will seek in 3 weeks to determine whether this infiltrate in the chest x-ray has cleared. And if it's not, it'll raise the question of some underlying disease process causing the pneumonia.

Figure 5.24. Physician's Search for Predisposing Factors

5.2.4.4 Recap: The Process of Explaining a Rule

To back up a bit to see what we have accomplished and what remains to be done, recall that the original problem of this section was to explain additional medical knowledge would enable GUIDON to explain MYCIN's rules, that is, to convince a

student that a rule makes sense. Even if we had a huge semantic net about physiological processes, how would we select from it to explain a rule? What relationships, what level of detail, would provide satisfying explanations?

The breakthrough in this problem comes from viewing the rule as statement that has to be proved (albeit informally), so that the problem of explaining a rule is tantamount to providing sufficient evidence that the conclusion holds. What constitutes sufficient evidence is obviously dependent on the particular conclusions of the rule (e.g., diplococcus-pneumoniae with strong certainty), but more importantly, our notion of sufficient evidence for the rule is strongly grounded in the *concept* we are trying to construct. Thus, we are led to the observation that a rule's preconditions must satisfy the implicit, abstract "preconditions" of the concept mentioned in the rule's action--a rule is something like a sketch of an existence proof.

An explanation for a rule has two constraints to satisfy: 1) it must account for all preconditions and rule conclusions, and 2) it must demonstrate, through plausible reasoning (involving a model of the listener's knowledge), that there is sufficient evidence for the process condition claimed by the rule's conclusion (e.g., "X is an organism thought (on the basis of clinical evidence alone) to be causing an infection"). The algorithm used by GUIDON to explain a rule is summarized below.

```
Input:  Rule premise and action
        Annotations: rule schema, screening relations, exception
                     and refinement clauses
        Frame for the action parameter
        Description of precondition parameters in terms of the frame's slots

Step 1: Remove contextual clauses using the rule schema template
     2: Remove screening, exception, and refinement clauses
     3: Output: Translations of all frame slots instantiated by remaining
                parameters
     4: Output: List of clauses that have not been accounted for
```

Figure 5.25. GUIDON's Algorithm for Explaining a Rule

Thus, to produce the explanation for the alcoholic rule, GUIDON (1) removed the (contextual) clauses that are implicitly part of the COVERFOR frame (bacterial meningitis diagnosed on the basis of clinical evidence); (2) removed the (screening) clause about the age of the patient; and (3) fetched the PORTAL-PATHWAY property of ALCOHOLIC and used it to fill in the corresponding slot in the COVERFOR frame. The algorithm currently used in GUIDON does not attempt to fill in every slot of the frame, for this would require additional knowledge about the organisms and considerably more sophisticated reasoning. For example, to satisfy the BUG-ORIGIN slot for the alcoholic rule, the program would have to use default knowledge about the BUG-AGENT Diplococcus-pneumoniae (mentioned in the rule conclusion) to account for its presence in the mouth.[1] GUIDON's explanations therefore only partly satisfy requirement "1" mentioned above: They account for every clause of the rule, but leave it up to the student to fill in the BUG-AGENT slot with the organisms mentioned in the rule conclusion, and to fill in other slots in the frame using default knowledge.

[1] In fact, Diplococcus normally exists there, even in healthy patients.

Producing explanations from a model of sufficient evidence is interesting because a satisfying explanation for any one of 40 rules is couched in terms of a concise description of a single underlying process. For GUIDON's explanations to work, the understander must know the mechanism description, and proceed from it, using the slots as expectations or necessary components of the argument. Given fitting instantiations for each of the slots, he will find that the inference is *satisfying*, that is, it fits his model of what constitutes evidence for the given concept. Of course, in developing the program further, we will want to perform experiments to determine the validity of this theory of explanation.

5.2.4.5 Completeness of the Explanation

As might be expected, GUIDON's algorithm for explaining a rule is strongly limited by the domain knowledge available to the program. Indeed, the major reason for implementing this primitive form of explanation routine was not to produce a practical system that explained all rules, but rather to illustrate some of the problems involved in making sense of a rule. GUIDON cannot really fit a rule to the COVERFOR frame because it lacks default knowledge about organisms and body sites.[1] For example, if a key factor refers to very contagious organisms, GUIDON has no way of checking to see if the organisms in the rule's conclusion actually are very contagious.

Nevertheless, GUIDON manages quite well to get at the essence of the explanation because, first, it has the necessary knowledge to strip away contextual and context-restricting information, and second, rules are optimized so that the

[1] See [Reither, 1978] for a study of reasoning with default knowledge.

parameters that remain in the rule's premise are those that really clinch the argument: they enable the frame, and default knowledge makes everything else fit into place. By characterizing the *enabling parameter(s)* in terms of the underlying model, GUIDON is supplying the essential step that makes everything else go forward. To see this again, consider the "crowd rule" (rule533) and the program's explanation.

```
RULE533
-------

If:  1) The infection which requires therapy is meningitis,
     2) A: A smear of the culture was not examined, or
        B: Organisms were not seen on the stain of the culture,
     3) The type of the infection is bacterial, and
     4) The patient does live in a crowded environment
Then:  There is weakly suggestive evidence (.2) that Neisseria-
       meningitidis is one of the organisms (other than those seen on
       cultures or smears) which might be causing the infection

RULE533
-------

     The fact that the patient does live in a crowded environment implies
     that the patient has been exposed to very contagious organisms.
```

Figure 5.26. The "Crowd Rule" and its Explanation

GUIDON's explanation is again superficial and it involves no real reasoning, but it makes a connection on the level that counts: Living in a crowded environment is relevant because of the *exposure to organisms*. That Neisseria is contagious is default knowledge that completes the argument that therapy for the infected patient should cover for this organism.

Some of the slots in the COVERFOR frame appear to be very general. What reason do we have to believe that GUIDON won't be called on to explain them further? The explanation for the "CNS radiation therapy rule" (rule515) states, "The fact that the patient has undergone recent radiation therapy to the CNS implies impairment of

the host-defense mechanism." Well, what is the impairment? The conclusion of the rule mentions the default list of "unusual bacteria" for meningitis (staphylococcus, e.coli, klebsiella, and pseudomonas) that we see in rules that mention Leukemia, Lymphoma, etc., so the impairment seems to be generally non-specific (see the sickle cell rule below for comparison). Indeed, when we consult an expert we find that the mechanism is not well-understood. Ironically, the best reason to believe that GUiDON's explanations (of rules like the CNS radiation rule) will be acceptable is that this is the best medicine can do. Connection to the underlying model must be enough, for it has been sufficient to teach rules like this to many doctors over many years.

The explanation of the "sickle cell disease" rule is similar to that for the "CNS radiation therapy" rule, but here medical science enables us to delve into physiological details. The sickle cell disease rule states that "if the race of the patient is black and the patient has had sickle cell disease, then this is weak evidence that diplococcus-pneumoniae is one of the organisms that might be causing the infection."

Consider how a student who knew the COVERFOR frame might proceed to explain this rule. Assuming that he knew that only Blacks get sickle cell disease (thus consideration of the race of patient is a screening clause), how would he make sense of this rule? There appear to be two good possibilities: either sickle cell disease causes diplococcus-pneumoniae infection (by enabling PORTAL-PATHWAY or IMMUNOLOGY) or sickle cell disease is caused by diplococcus (DIAGNOSIS-CAUSE). But he probably knows that sickle cell disease is a genetic condition and so it develops "by itself." Besides, he will have trouble explaining why diplococcus doesn't

cause sickle cell disease in other races, requiring yet another causal factor. The reasoning involved here bears some resemblance to that investigated by Stevens and Collins in the domain of meteorology (in particular, for understanding the cause of heavy rainfall) **[Stevens, Collins, & Goldin, 1978]**.

GUIDON's explanation of this rule is: "The fact that the patient has sickle cell disease implies impairment of the host-defense mechanism." As before, GUIDON offers no details about the nature of host-defense impairment. However, in this case more is known. The impairment is caused by angular blood cells (sickle cells) that clog the vessels of the spleen and kill it. The (empirically observed) function of the spleen is to act as a filter to trap diplococcus. This function is impaired, so the IMMUNOLOGY condition is enabled. These details are interesting and insightful, and they place medicine on a stronger scientific footing, perhaps providing flexibility for [illegible] about difficult cases where expectations are violated. We will not consider further how these details could be represented in GUIDON; though we observe that the explanation given above (using general notions like filtration and function-impairment) suggests that the idea of using an explanation model might be extended to characterize sufficient evidence for a condition in the COVERFOR frame, just as the frame we have given characterizes evidence of the COVERFOR concept.[1]

5.2.4.6 More Sophisticated Reasoning

Finally, it is important to point out that the process of fitting the COVERFOR

[1] Work reported in **[Rieger & Grinberg, 1977]** and **[de Kleer, 1979]** might be relevant here.

frame may involve much more sophisticated reasoning than simply filling in slots. In particular, when an argument is clinched by intermediate reasoning steps based on two or more of the rule's preconditions, it will be necessary to relate the preconditions to show how they together "fill in" a slot. This problem is illustrated by the "nosocomial rule" (rule 561).

RULE561

```
If:  1) The infection which requires therapy is meningitis, and
     2) The infection was acquired while the patient was hospitalized, and
     3) Meningococcal-disease is not one of the epidemic diseases to which
        the patient has been exposed recently
Then: There is suggestive evidence (.5) that neisseria-meningitidis is
        not one of the organisms (other than those seen on cultures or
        smears) which might be causing the infection
```

Figure 5.27. The Nosocomial Rule

Notice that this rule is saying that the COVERFOR frame is *not* satisfied. It does this by showing that for a given BUG-AGENT (Neisseria), the BUG-ORIGIN slot is not satisfied. We know from our general medical facts (what we have been calling default knowledge here) that Neisseria is seldom the cause of bacterial meningitis that originates during a hospital stay. This alone is evidence against Neisseria, however the rule author included another clause (3) to shore up the argument.

Suppose that the patient was admitted *with* Neisseria meningitis, but it was undiagnosed at the time. When the bacterial meningitis is detected during his hospital stay, it will be termed "nosocomial." Concluding that it could not be Neisseria (because it is nosocomial) would be wrong. To minimize misinterpretations like this, we check to see if there is a Neisseria epidemic, if not, then this is reasonable evidence that the patient did not *enter* the hospital with an undiagnosed case of Neisseria.

Thus, the rule author has hit BUG-ORIGIN from both sides: If the infection originated in the hospital, it is not Neisseria, because that is unusual, and if it originated outside the hospital, that too is unlikely to be Neisseria-caused because there is no epidemic (Neisseria is very contagious, another general fact).

Since we have evidence that Neisseria cannot satisfy BUG-ORIGIN, we conclude that it does not satisfy the COVERFOR frame (-.5). This illustrates that a rule explanation program must have the capability to form a chain of reasoning from the preconditions. However, note that the existence of the frame will be a significant advantage for focusing search. That is, the program needs to prove (or disprove, if evidence is negative) that the conditions of the frame are satisfied. Thus the model of sufficient evidence serves to reduce the combinatorics of the problem.

Finally, some COVERFOR rules can be characterized as being statistical, grounded on purely empiric observations, e.g., the age rule (Figure 5.13). They are default inferences that say, "given no other information about the patient, this is what typically causes an infection." The explanation model is not satisfied by rules like this, and it appears that whatever representation we use for the COVERFOR parameter, it will need a special slot to account for default evidence. This is not surprising, for several recent knowledge representation languages, e.g., KRL **[Bobrow & Winograd, 1977]** and UNITS **[Stefik, 1979]**, have chosen to label or distinguish inferences in this way.

5.2.4.7 What We Accomplished

To recap, we began by asking questions about MYCIN's rules, trying to account for all of the clauses. "Does diplococcus CAUSE sickle cell disease? Or does sickle cell disease leave one SUSCEPTIBLE to infection with diplococcus-pneumoniae?" Off hand, we couldn't tell which way the causality went in these rules. Though, when we step back, we see that these questions about causality and susceptibility are asked by an understander with an underlying model of the kinds of relations he expects to see mentioned in the rules e.g., organisms cause a diagnosed disease, previous illnesses weaken the body. In seeking an explanation, the understander tries to fit this rule to his model of disease. Put another way, if a human teacher (or program) is to *provide* satisfying explanations, he will have to respect this model as a constraint that the understander is trying to satisfy. And if the student is a complete novice, it will be important to sketch out the model itself so that the student will be able to learn (accept as meaningful) specific relations like this.

To illustrate how such a model could be used by a program to explain the rules, we formulated a frame to account for about 40 of MYCIN's rules. We believe that the algorithm provides satisfying explanations for about half of these rules. It falls short where deeper explanations are possible (as for sickle cell disease) or where clauses must be related to one another in order to fill in a slot of the frame (as for the nosocomial rule). We began by classifying the parameters because it appeared that there were different principles or means for deducing a parameter. We then discovered that each parameter (and rule) was referring (explicitly) to just part of an aggregate of conditions, and that explaining the rule involved filling in unmentioned

principles from default knowledge. (Thus, one rule mentioned evidence for entry of an organism into the body, another mentioned an immunological weakness, etc.) Explanation of a particular rule involves satisfying a single concept that is shared by all rules of the set, namely the concept of what constitutes sufficient evidence for the parameter being concluded about. This underlying single concept is represented here as a frame and is called the explanation model for the parameter.

5.2.4.8 Use of Explanation Models in Knowledge Acquisition

Restatement of rules via explanation models provides a simple, rudimentary facility for making semantic checks of the knowledge base. This is important because it ensures correctness of newly entered rules. Moreover, by having the tutor give its own explanation of a rule, it gives the rule author an opportunity to enter more annotations for students and other experts who will need help to understand the rule. Figure 5.28 illustrates both uses.

```
RULE511
-------

     The fact that the patient has had neurosurgery allows access of
     organisms from air and skin normal flora to cerebral-spinal fluid.

Can't explain the clause:
     the time since the neurosurgical operation was performed is less than 2
     months


RULE513
-------

     The fact that the patient has had neurosurgery allows access of
     organisms from air and skin normal flora to cerebral-spinal fluid.

Can't explain the clause:
     1) A smear of the culture was examined, or
2) Organisms were seen on the stain of the culture
```

Figure 5.28. Two Rules GUIDON Cannot Explain

Rule 511 illustrates how GUIDON is unable to account for the importance of knowing how long ago neurosurgery occurred. In fact, if the program knew this, it would be able to justify the table of conclusions that appear in the rule's action (see *Section 5.1.3 for related discussion*). *By showing this flawed explanation to the rule* author, a GUIDON-augmented knowledge acquisition program would be able to focus his attention on the clause that needs to be annotated in the rule justification.

Rule 513 illustrates an actual case where explanation models were used to detect an error in the knowledge base. In the process of testing GUIDON, the author used the explanation-generation function on all of the COVERFOR rules (40 pairs). Rule 513 is properly labelled as being an example of the schema COVERFOR.CLINICAL, but the OR clause is wrong (both subclauses should be negated). *This error had been* undetected for nearly 2 years. MYCIN's current capability to do syntactic type-checking is helpful for catching rule errors, but this example illustrates that using even a simple analytic approach (can we account for all of the clauses?) might be more effective.

5.2.5 Conclusions about Support Knowledge

We conclude here that knowledge representation for tutoring should distinguish among levels of abstraction. Problem-solving knowledge that uses a minimum number of intermediate reasoning steps (involving multiple levels of abstractions of case data, hypotheses and relations among them) might be perfectly adequate for high performance, but will be insufficient for explanation to users and students who lack the concepts for gluing it all together. Understanding the "tetracycline rule" involves

knowing about chelation, drug migration in the body, and even social criteria for what constitutes undesirable facial appearance. An important observation to make here is that the meaning of a rule can never be *completely* specified [Wittgenstein, 1958]. We need only prepare to give *reasonable* explanations. At some point, even a human tutor would give up, concluding that the student knew so little about the world that there was no hope to reason further on the level of diagnosing and treating infectious diseases.

Many of our other observations about the structure of MYCIN's knowledge led to the same conclusion, and many of our observations may be useful for designers of new representations. However, it is not the purpose of this thesis to suggest a new representation, but only to characterize knowledge for tutoring in terms of a critique of the existing MYCIN rule base. We want to emphasize that our critique is part of an ongoing struggle in the MYCIN team to decide how to best use rules for representing knowledge; it is by no means a sweeping condemnation of the formalism itself.

5.3 Strategical Knowledge -- for planning solutions

In previous sections, we have examined the use of structural knowledge to organize rules and parameters, and support knowledge to justify rules. In this section we relate the AND/OR tree of MYCIN-like rules and subgoals to a **deep structure** for working out a problem [Brown, Collins, & Harris, 1977] (hereafter this paper will be referred to as AILS). We characterize the observed behavior of questions asked by a consultant and an "output" diagnosis as a **surface structure** of the problem solution. We argue that current EMYCIN knowledge bases lack **planning knowledge** for ordering methods (rules) into a line of reasoning, and that this knowledge is conceptually equivalent to the meta-rule formalism implemented by Davis [Davis, 1976]. Finally, we demonstrate that addition of this planning knowledge in the form of strategies for controlling the search (construction) of the AND/OR tree would provide a basis for 1) providing assistance to complete partial student solutions; 2) planning which rules and topics should be included in the lesson plan; 3) summarizing the use of rules to attain a goal (motivate why the rules were considered at a given point in the problem solution); and 4) modeling approaches taken by the student to solve the problem and interpreting his hypotheses.

This analysis is based on the prescriptive statement made in AILS:

> Without explicit awareness of the largely tacit planning *and strategic knowledge inherent in each domain*, it is difficult for a person to "make sense of" many sequences of behavior as described by a story, a set of instructions, a problem solution, a complex system, etc.... The teacher should articulate for that domain the higher-order planning knowledge and strategic knowledge for formulating and revising hypotheses about what something means.

The hypothesis here is that providing a person with the means to understand the steps of a problem solution, is providing him with the basis for applying his problem-solving methods to other, similar problems.

5.3.1 Surface and Deep Structure of Rule Systems

As mentioned above, a person trying to understand a MYCIN-like consultation observes that pieces of case data are requested by the program (Figure 5.29). These questions are conceptually terminals hanging below an AND (rule) node in the subgoal tree (one portion corresponding to Figure 5.29 is shown in Figure 5.30.[1] In the terminology of the AILS paper, a rule node is a *method* for achieving a goal (e.g., "organisms that might be causing the infection") by setting up "deltacts" or some set of subgoals to achieve (the premise of a particular rule). Thus the tree of rules and subgoals constitutes a deep structure trace that Brown, et al. postulate is constructed when the understander makes sense of the surface problem solution.[2] (See also [Greeno, 1976].)

[1] Generally AND/OR trees for the infectious disease rule base are 3-5 levels deep; this is one reason that we believe a goal-directed dialogue does not become unmanageably hard to follow (the stack of goals is easy to keep in mind).

[2] Note that in AILS, subgoals follow from subgoals in a way that is not possible in MYCIN--intervening methods appear to be missing. For example, in the story about how a beggar obtained soup from a rich household, the subgoal of "getting inside the house" is followed directly by the subgoal of "conning the maid for permission to go inside." From our perspective, the beggar is applying the method by which one gets inside by getting a servant to let him in. (An alternative method would be to break in unseen.) The MYCIN rule formalism provides an interesting framework for viewing the understanding process outlined in AILS.

```
31) Has Pt538 ever undergone any type of surgery?
** YES
32) Does Pt538 have a history of NEUROSURGERY?
** NO
33) Does Pt538 live in a crowded environment?
** NO
34) Do you suspect recent alcoholic history in  Pt538?
** NO
35) Is meningitis a hospital-acquired infection?
** YES
36) Is Pt538's clinical history consistent with EPIGLOTTITIS?
** NO
37) Is Pt538's clinical history consistent with OTITIS-MEDIA?
** NO
38) Has Pt538 ever undergone splenectomy?
** NO
39) Is Pt538 a burn patient?
** YES
```

Figure 5.29. MYCIN Consultation Excerpt Corresponding to the Protocol

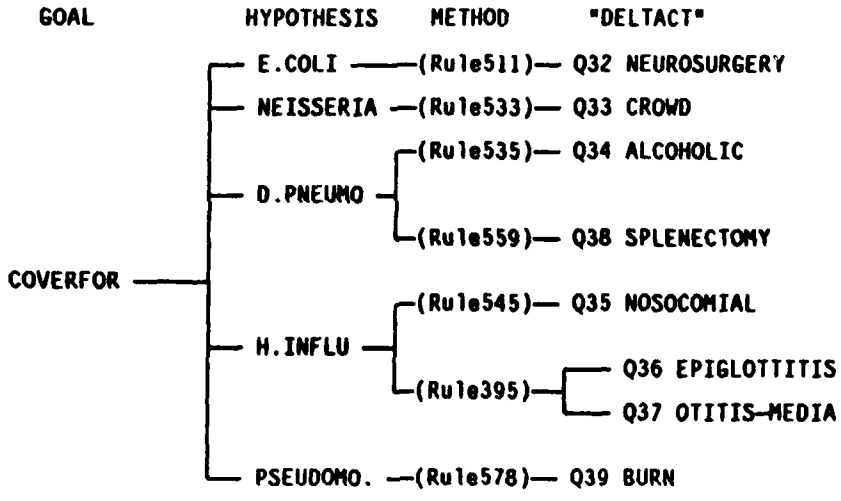


Figure 5.30. Portion of the AND/OR tree Corresponding to the Questions Shown in Figure 5.29 (reorganized according to the hypothesis each rule supports).

As discussed in AILS, it is not sufficient for a student to know all of the possible methods he can bring to bear on a problem. For example, a student who knows what kinds of algebraic transformations can be used to solve for "x" in "x**2 - 8 = 1" could only proceed to apply the methods randomly if he didn't have a *plan* for solving the problem (i.e., have schemas for kinds of problems that can be tackled using different *approaches* or lines of reasoning.) A plan sets up a sequence of applications of methods that might get you closer to the solution (though this is not guaranteed). Plans can be thought of as rules of thumb or strategies for dealing with recognizable situations. It is this planning knowledge that AILS suggests we might convey directly to the student.

5.3.2 Representing Planning Knowledge in Meta-rules

Davis showed how domain rules for achieving a goal might be ordered and pruned by meta-rules. These meta-rules are invoked just before the object-level rules are applied to achieve a goal. An example of an infectious disease meta-rule is shown below. Observe that this is a strategy for pursuing a goal.[1] In a simple way, it sets up a line of reasoning; in particular, it provides motivation for pursuing some subgoals before others: it will order the questions asked by the system, and hence change the surface structure of the consultation.

[1] In particular, this meta-rule might be associated with the goal "identity of the organism." It will be invoked to order the rules for every subgoal in the search tree below this goal. (Meta-rule001 is based on the observation that one of the group of organisms known as the enterobacteriaceae is more likely to be causing pelvic-abscess infection than a grampos-rod organism.)

```
META-RULE001

  If:   1) the infection is pelvic-abscess, and
        2) there are rules which mention in their premise
        enterobacteriaceae, and
        3) there are rules which mention in their premise
        grampos-rods,
 Then: There is suggestive evidence (.4) that the former
        should be done before the latter.
```

Figure 5.31. A MYCIN Planning Rule

In Figure 5.29 no meta-rule has ordered questions 31-39. Each question roughly corresponds to application of a single (deep structure) method, and these methods are applied *randomly* for each goal. Any strategical explanation of why MYCIN asked question 32 before question 33 will be spurious: the problem-solving system is simply applying rules (methods) exhaustively.

From our perspective of tutoring, meta-rules are an important part of an expert's explanation for applying a particular solution method at a particular time. This is consistent with Davis' computationally-motivated use of meta-rules to reduce the combinatorics of searching a large knowledge base.

Consider the problem of teaching rules to a student. We postulate that he will find it easier to remember the rules for a given goal if he has some strategy for applying them, i.e., some indexing scheme for ordering their retrieval. In particular, he might recognize the situation described by meta-rule001 above and follow that strategy for recalling particular rules to consider next. Similarly, there might be domain-independent strategies for examining the evidence gathered so far and seeking to confirm it, prove it inadequate, or further split the hypothesis space.

The GUIDON system does not currently use domain meta-rules in its tutorials. However, as discussed in Section 6.2.3.2, there are tutorial rules that are used to convey domain-independent diagnostic strategies to the student at the time he requests assistance for working out a subgoal.

5.3.3 Meta-strategies

It has been observed that teaching strategies explicitly is not always effective [Wescourt, et al., 1978]. For example, it is not always evident how Polya's problem-solving strategies should be applied to particular problems. During diagnosis, it is quite possible that one strategy will be useful for collecting the initial data and focusing the search, then another strategy will come into play to do a depth-first confirmation of the most likely hypothesis, and other strategies will determine when to consider revising a hypothesis. Burton & Brown have done pioneering work in incorporating knowledge of a student's strategies in the student model, and have considered some of the problems of detecting changes in strategy [Burton, 1979]. The use of meta-strategies for diagnosis has not been explored in GUIDON.

5.4 Epilogue: What is Meta-knowledge?

References to "meta-knowledge" have become increasingly frequent in this decade by educators, psychologists and AI researchers [Davis & Buchanan, 1977] [Barr, 1979] [Collins, et al., 1975] [Collins, 1978] [Flavell, 1977]. However it is not clear that there is a single definition that encompasses all of the ways in which the word has been used. In fact, there appear to be three distinct kinds of knowledge that we want to label meta-knowledge (Figure 5.32).

Type 1: Knowledge about the extent of what you know

Type 2: Meta-levels of knowledge: structure, support & strategy
(domain-specific knowledge about other knowledge)

Type 3: Knowledge about the thinking process

Figure 5.32. Kinds of Meta-knowledge

One meaning of meta-knowledge, used by cognitive psychologists, is knowledge about the extent of other knowledge. For example, your ability to indicate (more or less accurately) whether you could say the 16 points of the compass without being reminded of them first is knowledge about the extent of what you know. MYCIN's ability to "read the rules" gives it the capability to say what it knows. Rule models or program-generated patterns abstracted from the rules are another example. Thus meta-knowledge of this kind is knowledge that can be determined by introspection. It has the sense of describing what is known in general terms.

On the other hand, the meta-knowledge that we considered in this chapter--knowledge that structures, supports, and controls the use of performance rules--is *additional* domain-specific knowledge that happens to be about other knowledge. It is tempting to call these other "levels" of knowledge in the sense that strategy rules, for example, look down upon and control the use of problem-solving methods, and support knowledge provides detailed mechanisms and facts that underlie and hence justify methods. Thus, one level of knowledge either depends on or refers to another, leading to the term "meta-*level* knowledge" or simply "meta-knowledge." The rule schemas, explanation models, and meta-rules described in this dissertation are of this

type. Annotations in recent frame-oriented languages (FRL **[Goldstein & Roberts, 1977]**; KRL **[Bobrow & Winograd, 1977]**; UNITS **[Stefik, 1979]**) that explicitly characterize the way properties are inherited in a semantic structure are also examples of meta-level knowledge **[Barr, 1979]**.

The third use of the term "meta-knowledge" is to characterize knowledge about the thinking process itself. Flavell's "metamemory" research illustrates this. As one aspect of his research, Flavell has found that people learn to make judgments about the difficulty of retrieving certain information from memory, e.g., it is easier to recall a list of 16 words in the series "north," "north-northeast," . . . north-northwest" than a set of 10 randomly selected words **[Flavell & Wellman, 1977]**. This is knowledge about the capabilities of memory.[1] This form of meta-knowledge differs from those described above in that it does not refer to other things that we know about, but is instead knowledge about our ability to know or learn.

Finally, we should point out that complex judgments can use a mixture of these kinds of meta-knowledge. Indeed, we can see all three forms in a protocol analyzed by Collins **[Collins, 1978]**. A subject was asked, "Is the Nile longer than the Mekong River?" Figure 5.33 relates Collins' analysis of the subject's underlying inferences to the forms of meta-knowledge described above.[2]

---

[1] Interestingly enough, Flavell has evidence that this knowledge is learned and might be important for problem-solving effectiveness.

[2] Collins attributes inference 2 to the subject's knowledge of other long rivers ("meta-induction from cases"). However, note that this inference also depends on the subject's perception of his memory process: At this moment the subject does not recall that the Mekong is long (1) and he knows that waiting will not change that belief--hence, he "would know it *right now*" if it were so (2).

Subject's inference	Form of Meta-knowledge
1. I don't know that the Mekong is extremely long.	Type 1: *extent of knowledge*
2. I would know that the Mekong is extremely long if it were.	Type 3: *knowledge about memory process*
(Therefore, probably the Mekong is not extremely long.)	
3. The importance of a river depends in part on length.	Type 2: *abstraction (meta-level description of facts)*
4. The Mekong is not very important	Type 2: *importance of knowledge (meta-level relation)*
(Therefore, probably the Mekong is not extremely long.)	

Figure 5.33. Meta-knowledge Used in Plausible Reasoning (after Collins)

Our experience with the MYCIN program suggests that these kinds of meta-knowledge might in fact be intricately interwoven. As stated above, MYCIN knows what it knows about infectious diseases (meta-knowledge type 1) because it can read its rules. However, MYCIN's knowledge of rule syntax that enables it to read rules (templates of rule predicates) is the basis for this capability. From the human programmer's viewpoint, templates are part of the grammar of rules, and so in a sense constitute the underlying knowledge for decoding them. With respect to the *representation* of rules, templates are structural meta-level knowledge--clearly meta-knowledge of the second type we have described above. Thus, in MYCIN at least, the meta-cognitive ability to say what is known depends on meta-knowledge of the representation of what is known.[1]

[1] It is important to emphasize that this representational meta-knowledge is in the domain of programming languages, though it is is suggestive of the third kind of meta-knowledge--knowing about the thinking process itself.

In summary, we use the term meta-knowledge in this dissertation to characterize knowledge of distinctly different content--knowledge about: (1) what is known, (2) levels of abstraction, and (3) information processing capability. At the heart of the problem of distinguishing among types of meta-knowledge is that all reasoning inevitably involves other facts; recent work in cognitive psychology suggests that knowledge *about* facts is as important in problem solving as the facts themselves. If this is so, we may find that we are attaching "meta-" to the name of every cognitive operation, falsely suggesting that there are special kinds of induction, memory, etc., that operate with "meta-knowledge." Rather, the rise of terms like meta-induction, meta-memory, and meta-cognition reflects the importance of characterizing the content of problem-solving knowledge, rather than just the mechanism or process itself.

In fact, this chapter has illustrated that knowledge that is useful for tutoring goes well beyond the set of performance rules and control scheme that make MYCIN an effective problem-solver. In order to teach effectively, GUIDON has to *reason about* MYCIN's rules. Besides simply being able to access and dissect the rules, this reasoning involves knowing about their complexity, how they are related, what knowledge is necessary to understand them, and how they should be applied. Thus GUIDON makes use especially of meta-***levels*** of knowledge (type 2 meta-knowledge) in tutorials.

Chapter 6

Dialogue Management Heuristics for Rule-Based Tutorials

CONTENTS

6.1 Dialogue Management Overview:
- 6.1.1 Scope of the problem
- 6.1.2 Management Constraints
- 6.1.3 Knowledge of Discourse Structure
  - 6.1.3.1 Discourse Patterns
  - 6.1.3.2 Transitions for Alternative Dialogues

6.2 Rhetorical Structure of the Dialogue
- 6.2.1 Transitions I: ECONOMY
  - 6.2.1.1 Economically discussing a d-rule
    - -- Clause-by-clause:
    - -- Rules that succeed
    - -- Rules that fail
    - -- Recap: Complexity and Economy
  - 6.2.1.2 Economically discussing a topic
    - -- sufficient evidence
    - -- a single rule
  - 6.2.1.3 Completing discussion of a topic

- 6.2.2 Transitions II: DOMAIN LOGIC
  - 6.2.2.1 Coping with 'Improper' Case Data Requests
  - 6.2.2.2 Detecting Completion of a Topic

- 6.2.3 Transitions III: TUTORING GOALS
  - 6.2.3.1 Tutoring: orientation
    - -- presenting the goal rule and subgoal tree
    - -- "important" goals
    - -- input data and rules

  - 6.2.3.2 Tutoring: providing help
    - -- specific help
    - -- general help
      - > factors
      - > snags
      - > select and present a d-rule
    - -- help to make an hypothesis
    - -- a few complications

6.2.3.3 Opportunistic Tutoring
-- "Recent successes quiz"
> presenting an unknown rule
> dealing with unexpected expertise
-- "Recent failures quiz"
-- "Related rules quiz"
-- Experiments
6.2.3.4 Tutoring: Responding to an Hypothesis

6.2.4 Situation Transitions

6.3 Summary of Tutoring Principles

6.4 Dialogue Management Conclusion

6.1 Dialogue Management Overview

A tutorial program that engages a student in mixed-initiative discussion must have the means to manage its share of the dialogue. Managing a dialogue ranges from record-keeping and context-focusing to heuristics for directing the dialogue economically according to the needs of the student. GUIDON's expertise for carrying on a case method dialogue is represented as a network of procedures. The links between procedures are management heuristics, in the form of t-rules, that control transitions between dialogue situations.

This chapter surveys GUIDON's discourse procedures (sequences of t-rules). A dialogue transition diagram represents the procedural calling structure, revealing the structure of GUIDON's initiative during the tutorial. The penultimate section (Section 6.3) summarizes the principles that the program follows when guiding the dialogue. We conclude that a network of procedures is a useful representation for organizing heuristics for carrying on a structured dialogue.

6.1.1 Scope of the Problem

The complex and time-consuming task of diagnosing an infectious disease requires expertise for managing a dialogue that goes well beyond the local response strategies that have been considered in other tutorial programs. Sequences of student/tutor remarks can be structured into ***discourse situations*** or recurrent patterns in the discussion, making the mixed-initiative nature of this tutorial more complex than replying to a single student question and generating another problem.

For example, with the purpose of presenting a domain rule as a form of assistance (*the student doesn't know what to consider next*), the program not only chooses the rule to present, it also selects a method of presentation that accounts for the complexity of the domain rule, what the student knows, and what he will have to do before he can apply the rule. One action sequence is: (1) suggest a subgoal to the student; (2) discuss this subgoal with student (involving perhaps a protracted consideration of several other rules); (3) prompt the student for the conclusion that can be drawn using this new information; and (4) discuss related rules with the student. Discourse knowledge is also important for coping with the student's initiative. For example, the program has methods for deciding what to say when the student says that he knows the solution to a subtask (and doesn't want to discuss it in detail). Given these dialogue components and conditions for following them, we find that a basic characteristic of a mixed-initiative tutorial is the possibility of ***alternative dialogues***.

The WHY program **[Stevens & Collins, 1977]** has a *goal structure* that ultimately provides organization to the tutor's choice of subject material, but this is not the kind of knowledge about situations that we have considered in GUIDON. In GUIDON we have characterized alternative tutorial actions, such as wrapping up discussion of a rule, in terms of sequential considerations that comprise them (as illustrated by the example above). On the other hand, while we focus on the problem of making appropriate, prolonged presentations that go beyond interruption and repetitive question/answering (as in WHY), we use them as components in a larger scheme. Many of our t-rules bear resemblance in kind to tutorial strategies that appear in

WEST [Burton, 1979] (e.g., a rule that decides whether to interrupt the student), WHY (e.g., a rule that asks the student for relevant factors to support his wrong hypothesis for a goal), and WUMPUS [Goldstein, 1977] (e.g., rules that summarize evidence for a goal). The main difference between GUIDON and these other programs is that GUIDON's tutoring rules are structured by situations that occur in a comprehensive framework for discussing a case.

6.1.2 Management Constraints

Dialogue management involves coordinating tutorial goals with constraints imposed by:

1) the time available for the session

2) student initiative and conversational (social) postulates [Gordon, 1971] [Grice, 1975]

3) the communication channel

4) human memory and learning capability.

In early development of GUIDON, the limitation of time and verbosity have been the chief forcing functions: it is not possible to explicitly discuss every inference that the expert program attempts. By the use of heuristics for economical presentation and student initiative options presented below, the crux of dialogue management in GUIDON, session time for relatively simple cases has been reduced from about five hours to less than one (for students who know how to use the program).

The theoretical constraints listed above cut across practical issues of tutoring. These are the issues that are considered in this chapter and the next:

-- On what basis does a tutor select alternate presentation techniques?
-- How can we ensure dialogue connectedness and comprehensibility?
-- How does a tutorial program maintain and share dialogue context?
-- How can we provide for and cope with student initiative?

The first two issues, alternate presentation techniques and comprehensibility, are considered in this chapter. Chapter 7 deals with sharing context and student initiative.

6.1.3 Knowledge of Discourse Structure

Guidon's discourse procedures formalize knowledge of discourse situations and strategies for controlling transitions between situations. This section relates our work to other Natural Language studies, describes the t-rule formalism, and in broad terms characterizes the situations and strategies we have formalized. Subsequent sections will detail the implementation of the rules and survey their specific content.

6.1.3.1 Discourse Patterns

Recent Natural Language research has recognized that human dialogues exhibit regularity in the form of patterns that encompass several turns of initiative and that recur regularly [Levin & Moore, 1977] [Bobrow, et al., 1977] [Bruce, 1975] [Deutsch, 1974] [Winograd, 1977]. To quote Bruce [emphasis added]:

> (It is) ... useful to have a model of how social interactions typically fit together, and thus a model of discourse structure. Such a model can be viewed as a heuristic which suggests likely *action sequences*.... There are places in a discourse where questions make sense, others where explanations are expected. (These paradigms) ... facilitate generation and subsequent understanding. **[Bruce, 1975]**

GUIDON's knowledge for carrying on the case method dialogue is structured according to the recurring patterns that one finds in rule-based case method dialogues. Each pattern is represented as a *discourse procedure* that is made up of a series of steps; thus, these procedures are the action sequences mentioned by Bruce. Their applicability is determined by t-rules.

Faught **[Faught, 1977]** mentions two types of dialogue patterns: interpretation patterns (to understand a speaker) and action patterns (to generate utterances). GUIDON does not currently use interpretation patterns because it requires the student to change topics explicitly using command-oriented options. Discourse procedures are action patterns: they serve as an ordered list of options--types of remarks for the program to consider making at a particular time. For example, the procedure for discussing a domain rule (d-rule) includes a step that indicates to "consider mentioning d-rules related to the one just discussed." In this way, a discourse procedure step specifies in a schematic form **WHEN** a type of remark might be appropriate. **WHETHER** to take the option (e.g., is there an "interesting" d-rule to mention?) and **WHAT** to say exactly (the discourse pattern for mentioning the d-rule)

will be dynamically determined by tutoring rules whose preconditions refer to attributes of the knowledge being discussed and the communication model (student model, case lesson plan, and focus record). See Figure 6.1.

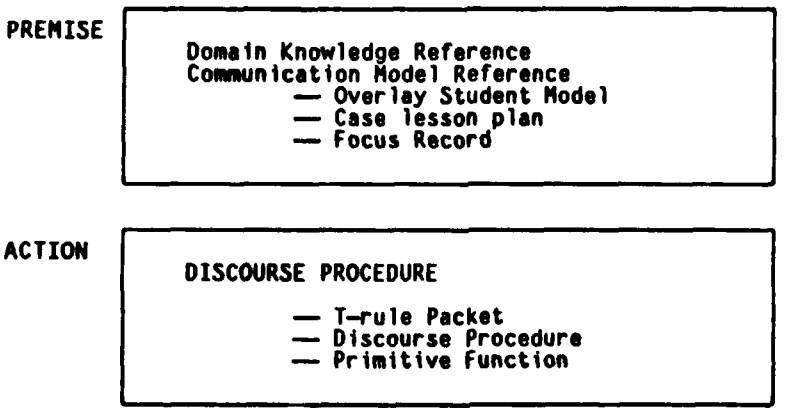


Figure 6.1. Form of a Tutorial Rule

T-rules are actually grouped into packets, each specific to a tutorial goal.[1] There are two types of t-rule packets: t-rules for selecting a discourse procedure and t-rules for modeling belief. A packet of t-rules for *selecting a discourse procedure* to follow generally applies its t-rules in order and stops when the first one succeeds. The form of t-rules of this type is shown in Figure 6.1 above. The action part consists of stylized code, just like the steps of a discourse procedure.[2] A step may invoke:

1) A packet of t-rules, e.g., to select an introductory remark for discussing a given d-rule.

2) A discourse procedure, e.g., to sequentially discuss each precondition of a d-rule.

---

[1] The INTERLISP representation is described in Appendix E.

[2] Discourse procedure steps also contain control information (e.g., for iteration) that is not important to this discussion. See Appendix E.

3) A primitive function, e.g., to accept a question from the student or perform bookkeeping (see Figure E.3 (in Appendix E) for a list of these).

Observe that the action part of a t-rule is a discourse procedure and discourse procedures invoke t-rules, so the representation is recursive.

The second kind of t-rule packet is for *modeling belief*--updating the communication model and rank-ordering d-rules according to an hypothesis revision strategy are two examples.[1] Generally, a packet of t-rules of this type is applied exhaustively. Communication model t-rules are described in Section 8.3. T-rules for choosing question formats are described in Appendix B.

6.1.3.2 Transitions for Alternative Dialogues

It is useful to formalize more than one way for the tutor to achieve a goal, e.g., more than one way to present a d-rule. Indeed, this is why there is generally a packet of t-rules for choosing a course of action. In this section we present the concept of ***alternative dialogues***, different ways something can be said and possible subdialogues that can occur. The concept is illustrated by a state transition diagram that represents the invocation structure of the discourse procedures in GUIDON. The links represent control expressed by tutorial rules within the procedures. These links signify choice points that lead to alternative dialogues, dictated by domain logic, economy, or tutorial considerations. Thus, these represent the management decisions in which the tutor takes the initiative to control dialogue situations.

---

[1] GUIDON uses *certainty factors* for representing the program's belief in something. Their value ranges between -1 and 1, with negative values signifying disbelief. See Shortliffe [Shortliffe & Buchanan, 1975] for discussion of their implementation and significance.

Figure 6.2 has been extracted from a tutorial session (using the same case that appears in Chapter 3, and analogous to page 59 of that interaction. Translations of MYCIN parameters have been simplified.) The discussion has reached the point of considering clinical information (non-laboratory data) to determine the organisms that might be causing the infection. The student asked whether the patient has a rash, and the tutor gave the reported case data (lines 5-7). At this point several remarks were made by the tutor: 1) lines 9-18, the datum requested by the student was related to the current topic by reciting the path that connects them in this case: rash -> Herpes Zoster virus -> infection type -> organisms; 2) lines 20-21, the tutor focused on the topic "Herpes Zoster virus," and stated a final conclusion; 3) lines 23-24, the tutor explicitly returned the student's attention back to the original topic of determining the organisms causing the infection.

```
                    .
                    .
                    .
     You should now proceed to ask questions that will enable you to
     make an hypothesis about the organisms that might be causing the
     infection.

 5   ** RASHES

     Pt538 does not have a rash.

     Your question is indirectly useful for determining the organisms
10        that might be causing the infection.

          Whether J.Smith has a rash can be used to determine whether
          J.Smith has symptoms of Herpes Zoster virus...

15        which can be used to determine the type of the infection...

          and this will enable us to determine the organisms that might
          be causing the infection.

20   The fact that J.Smith does not have a rash is evidence that J.Smith
     does not have symptoms of Herpes Zoster virus [RULE369].

     Back to our discussion of the organisms that might be causing the
     infection....
                    .
                    .
                    .
```

Figure 6.2. Subdialogue Involving a Deeper Subgoal

It now may be helpful to look at the dialogue transition diagram (Figure 6.3). Each node in this diagram stands for a discourse procedure, or sequence of t-rules. For example, when lines 1-5 were printed in Figure 6.2, the procedure GOAL was being followed, the basic procedure for discussing any MYCIN goal which appears in a domain rule. To answer the question about rashes, the DATA procedure was applied. Whether or not the patient has a rash is a "deeper subgoal" in that it does not appear

in any rule that can determine the current goal, but appears deeper in the AND/OR tree of goals and rules. The DATA procedure printed line 7 and set up the second-deepest subgoal as the new current topic (whether the patient has symptoms of Herpes Zoster virus). An arrow that loops back to a node signifies that the procedure that labels it is called one or more times, as one might expect. For example, we see that one or more "related rules" are presented at the end of discussion of a given rule. The italicized labels stand for the basis of the transitions--economy, domain logic, and tutoring goals--these distinctions are described below.

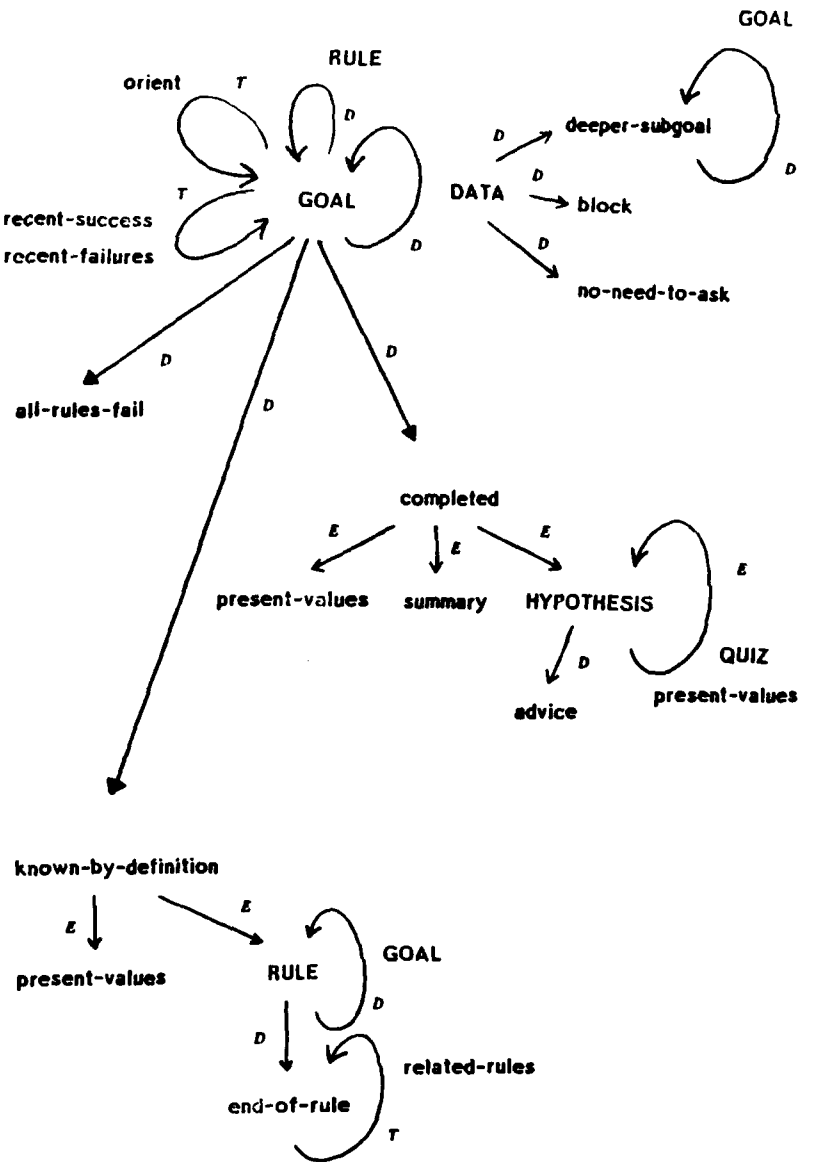


Figure 6.3. Dialogue Transition Diagram

Returning to our example, the tutor observed that the new topic (Herpes Zoster--see lines 12-13 in Figure 6.2) was completed because all of the case data that the expert needed to make a final conclusion had already been given to the student. Here a choice had to be made: Should the tutor present the final conclusion (as it did in lines 20-22)? Or should a summary of evidence be offered? Or should the tutor ask the student to make an hypothesis (as to whether or not the patient has symptoms of Herpes Zoster)? Figure 6.4 and Figure 6.5 show the procedure COMPLETEDGOAL. We see that T-RULE5.02 was applicable: a single domain rule was used by the expert program to make a conclusion and the student model indicated that the student knew this rule, so it was simply stated. The fact that the dialogue could have taken a different form at this point, at the tutor's initiative, illustrates the possibility of *alternative dialogues*. There are many more alternative dialogues than those illustrated by the different paths in Figure 6.3. For example, the procedure for discussing a rule (shown here as RULE) incorporates 18 different methods, e.g., clause-by-clause discussion, supplying case data and then asking for an hypothesis, and discussing a failed subgoal and then mentioning the conclusion that can't be made. These are described in Section 6.2 below.

COMPLETEDGOAL.PROC005

Purpose: Discuss final conclusion for the goal currently being discussed by giving the final answer or discussing an hypothesis

Step1: <Decide whether to finish with a summary>

Step2: <Discuss the final hypothesis for the goal>

Step3: <Wrap up discussion or record that it is complete>

Figure 6.4. Discourse Procedure for Discussing a Completed Topic

T-rule for Deciding Whether to Finish with a Summary:

T-RULE5.01 {*Summarize if >1 rule and at least one has not been discussed*}

If: 1) There are rules having a bearing on this goal that have succeeded and have not been discussed, and
2) The number of rules having a bearing on this goal that have succeeded is 1
Then: It is true that no summary of evidence should be offered for this goal

T-rules for Deciding How to Discuss the Final Hypothesis:

Apply the first rule that is appropriate:

T-RULE5.02 {*Directly state single, known rule*}

If: 1) There are rules having a bearing on this goal that have succeeded and have not been discussed, and
2) The number of rules having a bearing on this goal that have succeeded is 1, and
3) There is strong evidence that the student has applied this rule
Then: Simply state the rule and its conclusion

T-RULE5.03 {*Request hypothesis when rules may be unknown*}

If: You have examined the rules having a bearing on this goal that have succeeded and have not been discussed, and have found a rule under consideration for which there is not strong evidence that the student has applied the rule under consideration
Then: Substep i. If: 1) An introductory remark is to be made before requesting an hypothesis from the student, and
2) The student has not requested help for forming an hypothesis
Then: Say: hypothesis-ready
Substep ii. Discuss the student's hypothesis for the goal currently being discussed [Proc014]

T-rule for Ending the Discussion of the Goal:

T-RULE5.04 {*Now summarize or exit*}

If: 1) You have decided that no summary of evidence should be offered for this goal, or
2) There are no rules having a bearing on this goal that have succeeded
Then: Mark the goal currently being discussed as discussed
Else: Discuss a summary of evidence for the goal currently being discussed [Proc006]

Figure 6.5. T-rules for Carrying Out the Steps Shown in Figure 6.4

The COMPLETEDGOAL procedure illustrates two important constraints for choosing among alternative dialogues on the basis of the principle of ***economy***. Specifically, *surface complexity* features of the knowledge being discussed (e.g., the number of rules involved, whether a rule is definitional, that all relevant rules fail, and how many subgoals remain to be completed before an inference can be drawn) and the student model combine in these tutorial rules to select a presentation method that affects how extensive the discussion will be.

When a conservatively programmed tutor is not sensitive to the complexity of a situation or the student's knowledge, tedious, overly detailed discussion results; the program belabors topics that can be dealt with swiftly by a single remark or simply skipped over. For example, a striking improvement to GUIDON tutorials was based on the recognition that many MYCIN topics are "definitional," so the usual process of discussing all of the evidence explicitly and asking the student to make an hypothesis can be bypassed when the model indicates that the student knows the relevant rule.

Besides economical considerations, the dialogue transition diagram illustrates two other kinds of transitions: ***logical*** and ***tutorial***. The links leading from the DATA procedure are distinctions based on domain logic, as are the three straight links leading from GOAL. These links are based on domain facts and relations (e.g., that a topic is the name of a block of case data) and decisions made by the expert program (e.g., why a question need not be asked). In following these transitions, the tutor is reasoning about the subject material.

Tutorial transitions are based on the tutor's goals for teaching particular material

to the student and/or refining its model of his knowledge. The examples in Figure 6.3 are: 1) quizzing the student about rules that recently succeeded or failed because of case data just given to the student ("recent-success" and "-failure" in Figure 6.3); 2) providing initial orientation to get the student started on a topic that is new to him ("orient"); and 3) quizzing about rules related to the one just discussed ("related-rules"). We call these instances of *opportunistic tutoring* because they signify initiative taken by the tutor at appropriate times with the purpose of fitting as much information into the session as is practical. These quizzes constitute subdialogues that can be quite similar in motivation and format to SCHOLAR [Collins, 1976], WHY [Stevens & Collins, 1977], and ABLOCKS [Brown, et al., 1975] tutorials.

The appropriateness of the interruption is determined by its consideration within the sequence of a discourse procedure, a conventional dialogue pattern, and by heuristics that determine whether a remark should be made. The basic idea is that the program has certain topics that it wants to bring up (e.g., a couple of rules relevant to the current case that the student may not know), so it checks to see if the *kind* of remark that is appropriate at a given time provides the opportunity to raise a particular topic. Thus, after the tutor returned attention to the previous topic in Figure 6.2 (lines 23-24), it checked to see if there were any rules that it wanted to mention that had just then been applied by the expert program (there were none).

6.2 Rhetorical Structure of the Dialogue

This section surveys the dialogue situations of a GUIDON tutorial, organizing them according to the three types of transitions that bring them about: ***economical***, ***logical***, and ***tutorial/modeling***. In this organization, we will be emphasizing the underlying thrust of each shift in the dialogue, though particular t-rules will reflect a mixture of all considerations. For example, we consider t-rules for discussing a domain rule as examples of the *economic* thrust of the dialogue, though naturally the action of these t-rules will reflect the logic of the domain. Similarly, t-rules that present case data should adhere to principles of economy, though their thrust will be to cope with the domain logic: is a student's question redundant?

For each dialogue situation, we characterize (1) the tutor's goal; (2) actions that achieve that goal; and (3) conditions under which alternative actions are chosen. Where appropriate, we will consider general strategies that are incorporated in the design of a group of rules, for example the general approach for laying down a foundation and for filling in details. When there are just one or two rules that are significant heuristics for controlling the dialogue, they will be described in detail. Otherwise a reference will be given to a procedure listed in Appendix F.

The reader should keep in mind that the t-rules shown here are intended to illustrate plausible, interesting dialogues, not to advocate particular teaching methods. While there are in fact some general principles that we tend to favor, the point of this exposition, indeed the point of the initial implementation of GUIDON, is to exemplify what strategies are possible, rather than to prove claims about what is desirable. The

tutorial principles used by GUIDON are summarized in Section 6.3 at the end of this chapter.

6.2.1 Transitions I: Economy

Recall that dialogue transitions based on economy use information about the surface complexity of the knowledge being discussed and the student model to appropriately direct the dialogue. These management decisions reflect the conversational postulate that one should be perspicuous [Grice, 1975] [Bruner, 1966] or "to the point." It all boils down to giving GUIDON more than one way to do something, so that the program can tailor what it says to the specific situation at hand. For example, the program would have run perfectly well if it were only able to discuss a domain rule by taking each clause in sequence. Yet when all but one subgoal has already been discussed (in another context), why not simply discuss that one subgoal and mention the other information as an aside? This is the kind of alternative dialogue that t-rules based on economy bring about.

6.2.1.1 Economically Discussing a D-rule

In entering this situation, the tutor has already decided to discuss the application of a particular d-rule to the case at hand. GUIDON has 18 methods for presenting a d-rule; they are naturally categorized by whether they present d-rules that fail or d-rules that succeed. Thus, the tutor's goal in discussing a d-rule is to present an inference that may be made about the current case, or one that might have been made if conditions had been different. Discussing a successful rule clause-by-clause is sufficiently complex to warrant separate mention.

Discussing a Rule Clause-by-Clause

The protocol (page 82) illustrates how a rule that succeeds is discussed one clause at a time (see RULE.PROC001). The main steps are to announce the rule, present an initial quiz about the rule, consider each subgoal in order, and then discuss the rule's conclusion. This last step is a discourse procedure itself (ENDOFRULE.PROC017) that involves: (1) presenting the conclusion; (2) commenting upon possible circular reasoning, previous evidence, and the significance of this inference; (3) describing the completion status of this goal; (4) presenting related rules; (5) announcing the completion of discussion of this rule. This detailed method for discussing a rule is generally used in GUIDON when the rule is syntactically complex (e.g., it contains an iterative precondition) or the tutor wants to draw out discussion of the rule (e.g., to present an important rule that the student does not know).

Discussing Rules that Succeed: Specific Situations

This rule packet, SUGGESTRULE.RULEPK031, uses the following general methods for presenting a rule that successfully applies to the case:

1) ***Pose the rule as a question***, possibly giving new case data.

2) ***Discuss a subgoal***, and then discuss the rule clause-by-clause or wrap up discussion of the rule immediately.

3) Relate the rule to the context of the discussion, then ***discuss the rule clause-by-clause.***

4) ***Pose a hint*** about the rule's effect on the hypothesis discussed so far.

The conditions for selecting among these alternative dialogues are summarized below:

1) **The status of the rule's application:** Are there enough case data to evaluate it? What (how many) case data are missing? What (how many) subgoals (factors that involve rule application) remain to be considered?

2) **The recent context of the interaction** (focus record): mention of a deeper subgoal, a hint about this rule, or mention of a factor that appears in this rule.

3) **Properties of the rule:** whether a precondition is an iteration, whether the current topic is a true/false parameter.

4) **The student model:** does the student know this rule?

Discussing Rules that Fail

T-rules for presenting an inference that cannot be made are quite similar to t-rules for presenting successful rules. However, there are several special preconditions that are of interest:

1) A d-rule is said to *fail inappropriately* (for the purpose of forming it into a question) if there are factors in its ***precondition that didn't need to be pursued*** by MYCIN. This occurs when the interpreter determined that the rule failed before evaluating all preconditions, and skipped preconditions that weren't evaluated in other rules either. For example, the rule may have failed because the infection is cystitis, so MYCIN never needed to determine whether the patient had certain symptoms of that infection. Because facts are "missing," it is not possible to fit the rule into one of the question formats available to the program; the current t-rule that handles this case simply says that certain information is relevant without going into details (t-rule23.03).

2) It is especially interesting when a d-rule fails because a

single ***patient datum was unknown*** at the time of the consultation. GUIDON chooses a question format that singles out this information (t-rule23.04).

3) If a d-rule has an ***undetermined subgoal*** that is also useful for d-rules that succeed, it is worthwhile to discuss this subgoal (t-rule23.06).

Recap: How the Dialogue Situation Determines What is Economical

The rule packets for discussing inferences clearly illustrate how economical management of the dialogue stems from a collection of procedures for dealing with specific situations. From this we conclude that guiding the dialogue in an interesting, or efficient way involves (at least) an analysis of the superficial complexity of the knowledge being discussed. This is a domain-independent level of abstraction that deals with conditions we have listed above, e.g., whether a subgoal appears in other rules, whether more d-rules need to be considered before this inference can be attempted, how many subgoals remain to be considered, and so on.

It appears that, for a rule-based tutorial of the type we have formalized, the flow and structure of the dialogue is intimately tied to the abstract properties of the problem's decomposition into successful, failed, partially complete, and unknown subgoals and inferences. That this would be so is not surprising, given that the purpose of the dialogue is to "fill in" the abstract solution tree for the problem. It is not surprising that what will be interesting, what *needs to be said*, is directly determined by characterizing what remains to be done before the solution is complete. Thus GUIDON's t-rules detect the holes and fill them in. Each t-rule makes a small contribution by supplying a few more pieces of case data, wrapping up consideration

of a subgoal, and/or presenting an inference: in all cases moving up the tree in relatively simple, easy to follow steps.

6.2.1.2 Economically Discussing a Topic

Just as the long way to discuss a d-rule is to discuss each of its clauses, the long way to discuss a topic is to discuss each of the rules that conclude about it. It is here that the time element of a dialogue is critical: A single topic may involve 100 rules and it might require an hour to get through them all if they are all discussed. Moreover, no student is likely to sit still through a tedious consideration of the dozens of inferences that had no bearing on the current case. Many rules may deal with special, rare cases that are of academic interest; others may have been considered in previous tutorials and not require review.

Discussing Only Sufficient Evidence

For these reasons, GUIDON does not simply iterate over the list of rules for a topic. Instead, discussion of a goal focuses on the relevant evidence to be collected. Options discussed in Chapter 7 are designed to make it easy for the student to display lists of evidence to consider and to explore its use in the solution. GUIDON will be able to examine the student's understanding when he states hypotheses. The student model is updated then, and this is when the program decides which of the inferences require detailed discussion. Note that GUIDON does not attempt to follow the student's thinking by constructing a model to account for the case data he requests and the order in which he requests it. See Section 8.1 for further discussion.

While there are generally fewer pieces of case data to consider than rules that use them, it is still impractical to require the student to explicitly request each fact, or to wait for him to use one of the options that print blocks of information. For this reason, GUIDON uses a crude heuristic for detecting that sufficient evidence has been collected to warrant an end to the discussion. The current version of this t-rule (t-rule 1.11) compares the expert program's current hypothesis to the final hypothesis that it will reach once all evidence is considered. If all of the significant positive and negative conclusions are in agreement, then the student is told that discussion is substantially complete and asked whether he'd like to move on to another topic. If he agrees, then the procedure for completing discussion of a goal is invoked. A better t-rule might take into account the tutor's goals for covering particular domain facts and relations during this tutorial session.

Discussing Only a Single Rule

If the rules for determining a goal are mutually exclusive, as they often are, there is no need to consider each rule in every case dialogue. Other cases will illustrate the other possible outcomes. Instead of presenting all of the other rules, we might choose to review one or two "related" rules to prepare the student for more difficult cases.

Rather than annotate (or try to detect) the condition of mutually exclusive rules, we have chosen to implement a simpler heuristic (t-rule 1.02) that has more general applicability. We say that a topic is *known by definition* if it is determined (in the case under discussion) by a single inference that makes a definite conclusion (CF = 1.0). If

there are other rules for this topic, each either failed or was not tried. In either event, there is reason to deal with these rules as we would deal with a mutually exclusive set. A second reason for considering the single successful rule alone is that definitional rules tend to be easy to remember, so chances are good that the student will be willing to dismiss this topic quickly once he is shown or reminded of the relevant rule. (See Section 6.1.3 for related discussion.)

The procedure for discussing a definitional rule (DEFNRULE.RULEPK042) uses the student model and state of completion of the inference to decide whether to just state the conclusion, pose the d-rule as a question, or discuss it clause-by-clause. The economical presentation methods used here incorporate the t-rules for presenting conclusions during discussion of a student hypothesis (PRESENTVALS.RULEPK026).

6.2.1.3 Economically Completing Discussion of a Topic

The t-rules shown in Figure 6.5 presented earlier are designed to complete discussion of a topic. The tutor's goal is to make sure that the student knows the conclusion that can be drawn from the relevant case data and that he realizes that the discussion is complete. The t-rules used here reflect economical guidance of the dialogue by their choices concerning how the conclusion is discussed and whether a summary of evidence is offered. There are of course subtle issues involved in determining whether the student really understands MYCIN's conclusion and really agrees that no more remains to be said about this topic. Modeling the student's understanding is accomplished by subprocedures described elsewhere (Chapter 8). GUIDON has no capability to argue about factors that the student wants to consider

that are not reflected in MYCIN's rules, so after the conclusion is discussed and summary considered, GUIDON simply says the topic is closed and moves on (see, for example, page 64).

6.2.2 Transitions II: Domain Logic

Dialogue transitions based on domain logic are forced upon the tutor by a combination of the facts and relations of the domain and the program's adherence to a goal-directed dialogue. There are three situations that the tutor must detect and cope with: 1) requests for data that, from MYCIN's viewpoint, should not be requested; 2) requests for data that implicitly change the topic; 3) completion of discussion (when all data have been requested). We will consider improper requests for data and topic termination here because they are rooted in domain facts and relations; we will consider implicit changes of the topic in Section 7.1.1.2 because it is basically a problem of sharing and maintaining dialogue context.

6.2.2.1 Coping with 'Improper' Case Data Requests

Naturally, if the student requests information we should assume that he has a good reason for asking. When we say that a request is *improper*, we mean that MYCIN does not need the requested information at this time, and may in fact never need it. GUIDON makes no attempt to account for an improper request; in full generality, this might require forming new (plausible) domain rules that model the student's understanding (see Section 8.5). Instead, GUIDON's goal is to tell the student why a data request is improper. The possible reasons are listed below.

> 1) The request is ***premature***: MYCIN will eventually ask, but first wants to try to deduce the information from rules. That is, there are other relevant case data that should be considered (of more general interest) that may indicate that this specific question is unnecessary. If there is a parameter that screens this question, and it needs to be asked, then GUIDON presents the screening information (t-rule 32.04). The originally requested data will be

given by GIVENPARMDATA.PROC012 at this time, if it is no longer premature. (This is the case when the screening parameter appears in the only rule that needed to be considered before a question could be asked.)

2) The request is ***inappropriate for a consultant***: MYCIN never asks for this information (t-rule32.02).

3) The request is ***unnecessary*** for this case: MYCIN decided not to ask or determined the value by applying a definitional rule (EXPERTDIDNTASK.RULEPK033).

4) The request is relevant, but ***not useful***: MYCIN asked for the value, but given the order in which the student has acquired information, he does not need to know the value. That is, all rules that use this information can be evaluated on the basis of what is already known (t-rule 12.07).

6.2.2.2 Detecting Completion of a Topic

Discussion of a topic ends when there are no more relevant case data for the student to collect. Three situations occur:

1) All of the rules for determining the goal have been evaluated by MYCIN and a ***final conclusion is in hand*** (t-rule 1.09).

2) There are ***no rules to apply*** for this topic in this case. For example, if the case is about an infectious disease patient, there may not be cultures with organisms growing on them, thus there are no unknown organisms to identify. Any rule that mentions parameters of unknown organisms will not even be tried by MYCIN because there is no "context" in which to evaluate the rule (see T-rule1.01).

3) ***Rules were tried, but all failed***. While we could let the student go ahead and collect relevant data, in the interest of economical discussion GUIDON simply gives the data to the student all at once (ALLRULESFAILPARM.PROC035). The implicit strategy is

that it will make more sense to discuss this topic in detail when inferences can be drawn. (But note that this t-rule is flawed because it may be advantageous in certain situations to let the student discover for himself that no inference can be made. T-rule1.03 could be modified to take this into account.)

6.2.3 Transitions III: Tutoring Goals

The point of t-rules based on tutoring goals is to actively seize the initiative to present specific rules and relations among them. These t-rules will frequently call on t-rules based on economy to keep the dialogue moving and present information in easy to digest chunks. They include basic teaching strategies that structure the presentation of new material, as well as problem-solving strategies for solving a case. GUIDON's t-rules of this type fall into four categories: general orientation, providing help to work out the problem, opportunistically examining understanding, and responding to a student's hypothesis.

6.2.3.1 Tutoring: Orientation

By *orientation* we mean conveying the structure of the knowledge base to the student by establishing a foundation for acquiring specific rules. To a certain extent, we have based the t-rules and student options on a general approach that we feel is useful when learning about a new MYCIN-formalized domain. This approach is to be contrasted with trying to learn the rules by reading them in an unorganized listing. We offer a refinement strategy that is based upon the underlying inference structure of the domain, as follows.

1) What is the **goal rule**?

2) What is the **main subgoal structure** (sketch of a typical solution tree)?

3) What do **premises of rules** look like? Are there patterns?

4) What are the important input data? What kinds of judgments does the user have to make?

5) What are some typical values (outcomes) for the major subgoals and what are some typical rules to conclude about these values?

These organizational concepts and relations are designed to help a student grasp the contents of an existing knowledge base.[1] The power of this approach lies in the implicit use of an abstract solution tree that can serve to hold in place the various organizing relations: goal rule anchored at the top; intermediate concepts scattered, but tied together, throughout; and typical inferences to give substance to the judgmental concepts (typical data and typical outcomes). The student is freed from having to consider specific rules; instead he is told only typical, common, or important associations.

The extent to which GUIDON follows this *orientation strategy* is described below.

Presenting the Goal Rule and Sketch of the Tree

Every MYCIN-like system has a rule that stands at the head of the solution tree. It states the task to be completed ("determine the therapeutic regimen for the patient") and the major subgoals ("determine the organisms that require therapy," "consider secondary infections"). To carry out the orientation strategy, GUIDON's procedure for discussing a goal treats the top goal of the solution in a special way. The following t-rule states the goal rule (substep i.) and prints the sketch of the subgoal tree (substep ii.) (see Appendix D).

[1] It is interesting to note that they may be useful guidelines for helping a human expert to formalize what he knows so that we can fit it into the MYCIN representation scheme.

```
Step4: T-RULE1.04

       If: 1) The goal currently being discussed is the goal of the
           consultation, and
           2) A: The student's estimation of his sophistication (on a scale
              from 0 to 4) is less than  3, or
              B: The goal of the consultation has not been discussed before
       Then: Substep i. Say: desc-goalrule
             Substep ii. Show a sketch of the subgoal tree for the goal
                   currently being discussed
             Substep iii. Say: tell-optionname
             Substep iv. Discuss the goal rule of the task clause-by-clause
                   [Proc004]
             Substep v. Say: goodbye
             Substep vi. Record that the goal of the consultation has been
                   mentioned in the dialogue
             Substep vii. Exit this procedure
```

Figure 6.6. T-rule for Discussing the Top Goal (from GOAL.PROC001)

Substep i. invokes a procedure that states the goal rule in a case-independent terms. One problem here is that MYCIN's goal rule has evolved to an awkward form that needs to be greatly simplified if it is to be of much value to a new student. This could be done by collapsing the clauses into a few new intermediate concepts like "collect basic patient data."

Substep ii. invokes a procedure that recursively prints subgoals of the entire solution tree, showing only those that are "important." The current heuristic is that a subgoal is important if the number of rules in which it appears (rules that use it or conclude it) exceeds a prespecified threshold: 10 in the MYCIN system. The resulting "sketch of the subgoal tree" is intended to bring out the topics that will play a prominent part in the dialogue: subgoals that are concluded by a large number of rules will probably require more time to discuss, and subgoals that are used by a large number of rules will frequently enter into the discussion of other topics. Therefore, by considering these subgoals, the student will have a good beginning for evaluating his grasp on the problem.

Assuming that the student wants to use MYCIN's approach as a starting point for evaluating his understanding, the sketch of the tree might lead him to ask specific questions that will get him started on the problem, e.g., "What does 'type of the infection' mean?" "What will the patient's history of surgery tell me about organisms causing an infection?" To a beginning student who has very little knowledge of his own to go on, this sketch can serve as a map of the solution, and he can refer to it as *a checklist for filling in a comprehensive understanding of the expert's handling of the* case. For this reason, the FACTORS, SUBGOALS and PENDING options are available at any time during the dialogue to provide similar orientation information on a slightly more detailed level (see Section 7.1.4.1). Substep iii. of t-rule 1.04 tells the student about these options.

The strategy described here has one basic flaw: it will probably overwhelm the student with too much information about MYCIN's solution, and it will probably frustrate any student who knows enough so that he could have gotten started on his own. The orientation strategy may have merit, but clearly 3 or 4 pages of introductory text is excessive. One improvement would be to ask the student if he wants to see the goal rule and subgoal sketch, or perhaps show them the first time the student requests assistance.

Providing Orientation for an Arbitrary Goal

The strategy for providing orientation for a new topic is to show the student what typical rule premises look like (by listing the rule model) and to list typical values or hypotheses that one might consider. The rule model is similar to, but not

necessarily equivalent to, the subgoal sketch that appears below this goal. The typical values are determined by a batch-process procedure that examines the rule set and statistically ranks the values that actually appear (this is stored on the file with other rule and parameter annotations).[1] The t-rule that brings this all about is shown in Figure 6.7.

```
Step2: T-RULE36.02

       If: 1) The number of rules that conclude about the current goal is
           greater than  5, and
           2) The goal currently being discussed has not been discussed
           before
       Then: Substep i. Give a fixed-text definition for the goal currently
                        being discussed
             Substep ii. Provide orientation for finding out about the goal
                         currently being discussed by listing the subgoals
                         most commonly pursued in this context
             Substep iii. Say: orient-by-rule
             Substep iv. List (typical) values for the goal currently being
                         discussed
             Substep v. Record that the goal currently being discussed has
                        been mentioned in the dialogue
```

Figure 6.7. Orientation for a New Topic (from ORIENTATION.PROC036)

Substep iii. of this t-rule presents statistics about the number of d-rules that conclude about this goal (protocol page 47). This remark is intended to indicate to the student how much MYCIN knows about this topic, and thus forewarns him about the lengthiness of the ensuing discussion. Information about the extent of knowledge to be learned is a form of meta-knowledge like that investigated by Flavell [Flavell, 1977]. The pedagogical principle to be tested is that humans find meta-descriptions

[1] The list of "legal values" for a parameter is not suitable because it may include rare outcomes that are not relevant to MYCIN's area of expertise. For example, only half of MYCIN's list of organisms are mentioned in the meningitis and bacteremia rules. This list is useful for modifying the rules in quizzes; rare values will be obviously incorrect in a multiple choice question. Indeed, the program appeared to be making jokes before we restricted values to those that actually appeared in the rules.

like this to be useful orientation for learning new material and/or carrying on a prolonged dialogue.

Input data and rules

The student can see what case data are relevant to a goal in general by using the FACTORS options. The CONCLUDE option will describe the evidence that supports (or contradicts) a particular hypothesis. These options are discussed further in Section 7.1.4.

6.2.3.2 Tutoring: Providing Help

The HELP option is interpreted to mean "suggest something that will help me complete the current subtask." GUIDON's strategies for providing help are represented by two procedures, one for providing help specific to a relevant factor (SPECIFICHELP.PROC020) and another for providing general assistance (HELPFORGOAL.PROC016).

Specific Help

Help specific to a particular relevant factor (e.g., "help me see how information about the patient's head wound is incorporated in the solution) is provided by a sequence of t-rules that: 1) provide case data if appropriate; 2) describe hypotheses that the factor supports (USE option); 3) open up discussion of the factor (if it has undiscussed rules); and/or 4) discuss the final conclusion and summary of evidence. This discourse procedure illustrates that one aspect of *tutoring* involves methodical,

step-wise consideration of the solution process. GUIDON follows a logical procedure for making sure that the student makes the right connections: 1) Does he know what case data have been reported? 2) Does he know why it is important to have this information? 3) Has he considered how to work out this subtask? 4) Does he know the answer and what evidence led to it?

Observe that help specific to a subtask of the current topic is very much restricted in focus: the tutor looks down a subtree and "fastens" it, if you will, to the broader context of the current topic. General help, of the "what should I do next" variety, requires an entirely different strategy.

General Help

GUIDON has a three-pronged attack for helping the student complete the current goal. First, if the dialogue has not recently focused on a factor or hypothesis (focus record is empty) and the student has not yet seen the factors for this goal, then all of the factors for this goal are listed (t-rule 16.01). (This is the FACTORS option; see Section 7.1.4.1.) The idea is that it is better to have the student look over the list of questions and subgoals he might pursue, than for the tutor to immediately seize control by getting the student started on a particular solution step. Thus, the principle is to leave the initiative in the hands of the student whenever possible.

The second strategy is to look for snags in the problem solution, and so get the student past a difficult point. GUIDON looks for one particular snag: the presence of "second thought" (self-referencing) rules that cannot be considered until the "normal" inferences are all taken into account. This t-rule (t-rule16.02) has a very definite,

MYCIN-related function: GUIDON may be waiting for MYCIN to give the signal that the topic is done, but MYCIN is sitting still with one or more successful inferences (self-referencing rules) that will not be allowed through until some esoteric evidence that caused (normal) rules to fail is considered. This is a very likely time for the student to be confused about what to do next.

To get past this snag, GUIDON presents each of the (failing) rules that MYCIN wants to have evaluated. The resulting interlude is short because there are usually just a few rules of this type and economical presentation is used (never clause-by-clause). Of course, it would be possible for GUIDON to tell MYCIN to skip these rules, effectively pretending that part of the solution tree doesn't exist. However, we took a conservative approach in the initial implementation to see what would happen if the tutor and student were forced to consider each piece of case data acquired by MYCIN during its consultation.[1] As long as there are just a few failing rules to present, the "snag t-rule" probably will not need improvement.

The third strategy for providing general help is to select and present a d-rule relevant to the current task. The idea is that the most direct way to help the student complete the current task is to get him started on a particular inference for solving it. The presentation strategies were presented in Section 6.2.1.1; it is the selection strategies that are of special interest to us now.

To select a rule, GUIDON first uses the focus record to select a relevant subset of the d-rules for determining the current goal that have not been discussed

[1] This led ultimately to including options like TELLME and ALLDATA, and incorporating the *substantial completion* termination condition for a topic Section 6.2.1.2.

(FOCUSRULES.RULEPK037). The candidates are: the rule that was the basis for a recent hint, rules that mention a particular factor, and rules that support a particular hypothesis. So if the student just asked how to USE information about epidemics, only the rules that mention epidemics in their premise and have not yet been discussed will be considered (these are called the *rules under contention*).

The second step is to choose a single d-rule from the list of rules under contention. GUIDON applies a set of strategies (RULEINTEREST.RULEPK031) that assign a numerical "interest" score to each d-rule; the list is sorted on this basis and the top-ranking one is presented to the student as a form of help for continuing the diagnosis.

The t-rule that carries the most weight in the ordering process is given below (t-rule 27.01). The purpose of this t-rule is to capture the intuitive notion that, in general, change in belief is interesting: the more drastic the change, the more interesting the effect. The numbers in the conclusion of t-rule 27.01 are certainty factors that indicate our belief in this interestingness. Thus, the domain rule that changes the current hypothesis most drastically, according to the relative weights given by this rule, will be selected for consideration by the student (other factors being equal).[1]

[1] Recall that the value of a goal is *significant* if its CF exceeds the cutoff (Section 5.1.5). For example, if the goal is the "organism causing the infection" and the certainty associated with the value "pseudomonas" is .3, then this value is not significant. "Sum over the computed value" means that the effect of each value concluded by the rule will be taken into account, not just the particular value that will be mentioned if the chosen presentation method is to state a hint.

T-RULE27.01

If: 1) The rule succeeded in this context, and
2) The following value has been determined: a hint value can be suggested for this rule, based on its effect on the current hypothesis
Then: Modify the cumulative belief that this rule should be presented by the sum over the computed value as follows:
If the evidence contributed is still insignificant then .1
If a new insignificant value is contributed then .1
If "confirmation of related evidence" then .3
If a new significant value is contributed then .5
If a new strongly significant value is contributed then .75
If an insignificant value becomes significant then .8
If an old value is now insignificant then .85
If belief in an old value is strongly contradicted then .95

Figure 6.8. GUIDON's Strategy for Revising an Hypothesis

T-rule 27.01 embodies a particular strategy for revising hypotheses. When the selected d-rule is presented to the student, this strategy is presented indirectly to the student as a form of motivation for considering the d-rule at that point in the dialogue (see Figure 6.9 and page 68 of the protocol). GUIDON essentially implies, "If you knew this d-rule and were following this strategy, then you might pursue this evidence at this time." In this way, the program would be underscoring 1) the specific effect of the d-rule (when the d-rule is introduced as one that strongly confirms the "bacterial" hypothesis) and 2) the strategy of seeking evidence that strongly confirms a current hypothesis (though, in general, t-rule27.01 is designed to force discussion of evidence that overturns a current hypothesis if possible).

Of course, there is no claim that following the strategy of t-rule 27.01 is the best way to do diagnosis. Instead, the reader should observe that: 1) this t-rule is conceptually a meta-rule--a strategy for applying solution methods, 2) we might write other rules of this form to capture other strategies for revising hypotheses, and 3)

like other t-rules this is an heuristic that is designed for experimentation. For example, another strategy would emphasize orderly collection of basic data and forward-directed inferences to focus on most likely hypotheses.

Finally, there are other t-rules for ranking the rules under contention. These take into account the student model, whether the rule is on the lesson plan, how much work remains to be done before the rule can be applied, and whether the rule incorporates tables or lists in its premise. In the current version of GUIDON these are primarily place-holders that illustrate typical factors that might be taken into account.

. . .

{During discussion of the type of the infection, the student asks for assistance.}

** HELP
Hint: Look for strong, positive evidence that the type of the infection is bacterial.

{The previous request for help set some global variables (the "focus record") that are now used when the tutor provides assistance. Specifically, GUIDON selected d-rule501 (using t-rule27.01 and others) and chose to give a hint instead of presenting the d-rule (based on the student model and features of the d-rule). Now, with the second request for help, GUIDON presents the rule by discussing it step-by-step (t-rule16.04).}

** HELP

[Applying tutorial rule 16.04]
[Use-History of RULE501: 0 -> -400]

{The student model is altered to reflect the fact that the student didn't follow up on the hint by considering rule501. (It is not clear at this point what the modification should be.)}

Let's consider a rule that contributes strong, positive evidence that the type of the infection is bacterial....

(Now considering RULE501.)

. . .

Figure 6.9. Using a Problem-solving Strategy to Construct a Hint

Helping the Student to Make an Hypothesis

When asked to make an hypothesis, the student can request help. GUIDON's strategy for providing help is to select a rule that has been successfully applied by MYCIN and has not yet been discussed. The selection and presentation methods are the same as those described above. Though of course since the selected rule has already fired, no additional case data will be presented and no new subgoals will be set up for discussion; a question is usually posed. The tutoring principle is to encourage the student to consider inferences that can be made using information he has already been given.

A Few Complications

There is an inherent problem in trying to cleanly separate the rule selection strategies from the rule presentation strategies. In particular, recall the t-rule that represents GUIDON's strategy for revising hypothesis (t-rule27.01), which causes a particular d-rule to be chosen because of the interesting revision it brings about. The hypothesis value that GUIDON deems most interesting at this time should be the same value that GUIDON mentions to the student in a hint, if this is the chosen presentation method. Therefore, the t-rule that states the hint must know about the t-rule for revising hypotheses. How is this possible? When the d-rules are ranked, t-rule27.01 records with each rule the value that should be stated in a hint.

Interactions of this kind are rare in GUIDON, but they illustrate the general problem of control when the decision of what to do is separated from its execution. The program must maintain a record of possible and/or intended actions on a sort of

sketchpad memory, so that after all possibilities have been weighed, the program can return to the best plan and follow it. This is essentially what GUIDON does when constructing a hint.

However, the problem is more than just "remembering what you decided to do." In general, it is possible that the planning phase will require descriptions of the alternative actions, rather than independent measures of what is desirable. For example, it is possible that a good reason for presenting a rule is that the situation allows a preferred presentation method to be used. Suppose that we know that the student prefers for GUIDON to begin the discussion of a rule (by giving case data or setting up a subgoal), rather than providing a hint. The t-rules that rank the d-rules under contention would have to take into account the pros and cons of valid presentation methods, rather than simply looking at the student knowledge model, or abstract properties of the rules. Ideally, the ranking t-rules would directly examine the presentation t-rules, or perhaps a program-generated description of them. For the moment, we must hand-craft interactions between t-rule packets.

6.2.3.3 Opportunistic Tutoring

Opportunistic teaching strategies are intended to periodically examine the student's understanding and introduce new information. For example, a dialogue transition (from GOAL.PROC001 to RECENTSUCC.PROC043) occurs every time exchange of case data causes d-rules to succeed, on the chance that the student has just demonstrated knowledge of a d-rule (by requesting case data relevant only to this rule) that the model indicates is unknown to him. Thus, opportunistic tutoring

interruptions are made 1) in an appropriate context (as part of a discourse procedure), and 2) when the tutor discerns that interruption is justified. Tutoring interruptions are of course interesting because of their possible beneficial effect on the student's learning. Two plausible effects are: 1) the short-term effect of introducing judgments that will be useful when the student expresses his hypothesis for the given goal, and 2) long-term exposure to more rules over several sessions, and thus better retention (in this sense the tutor's interruptions will be deliberate attempts to review previously discussed material).

While it is not too hard to formalize when interruptions should be considered, it is quite difficult to formulate good strategies for deciding when such an option should be taken. Preliminary investigation indicates the tutor will have to balance the cost of interrupting the student against the benefit gained from improving the model of his understanding, and thus being in a better position to offer appropriate remarks later. For example, it might prove advantageous to verify the student model right away, rather than allowing a weak hypothesis of what the student knows to ripple down the line and pose problems later. Opportunistic tutoring experiments are considered at the end of this section.

Rules that Recently Succeeded

GUIDON has two strategies for exercising the option to present a d-rule that just succeeded. The first, detecting "unexpected expertise," was described earlier (t-rule 43.02). Based on the student model, we might have predicted that the student would not request the data used by a rule. By interrupting and asking a

question, we attempt to realign the model with the student's actual understanding, perhaps giving the student a rule that he didn't know about and can now apply. Note that this strategy reflects the only extent to which GUIDON follows the student's question-asking. Moreover, rather than trying to account for each piece of information he requests, we wait until the rules that use these data succeed. This elementary modeling method may be useful to indicate the worth of constructing a much more extensive, and expensive, scheme.

A second, significantly more complex interruption involves discussing an entire subtree of the solution so that the student can make an inference (t-rule 43.01).

```
T-RULE43.01
-----------

If: 1) The rule has not been discussed,
    2) There is not strong evidence that the student has applied
    the rule under consideration, and
    3) There is a subgoal that the student probably does not know
Then: Substep i. Print a design of a line of dashes
      Substep ii. Say: intro-recentsucc-tree
      Substep iii. For each of the rules in this subtree that the
                   student probably cannot apply because he is
                   unable to determine one or more subgoals
                   Do: Present missing rules to the student so
                       that he can determine the missing subgoal,
                       then quiz about rules that use these
                       subgoals [Proc046]
      Substep iv. Print a design of a line of dashes
```

Figure 6.10. Tutoring Rule to Detect and Present a Missing Subtree

Here we make a distinction between knowing a rule and being able to apply it to a particular case. For example, we may have (a priori) reason to believe that the student knows that "if the type of the infection is fungal, then treat for cryptococcus or coccidioides." For a student at a certain level of sophistication, this rule will be obvious. However, determining whether or not the type of infection is fungal is non-

trivial; it requires knowing several specialized rules for interpreting cerebral-spinal fluid tests. The purpose of t-rule 43.01 is to present the specialized rules in a context in which the student can immediately see their benefit in the light of what he already knows. Figure 6.11 graphically illustrates this strategy.

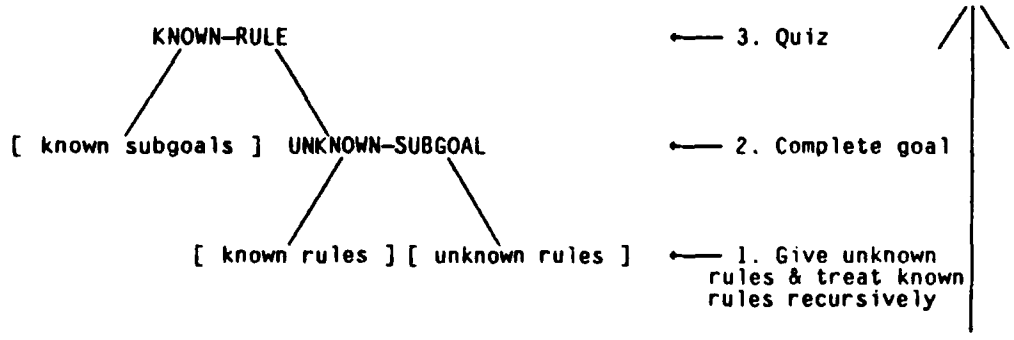


Figure 6.11. AND/OR Tree Representation of the Strategy Used by T-rule43.01

This diagram shows the discourse pattern followed for each level of the tree having known rules with unknown subgoals, i.e., subgoals that the student is not able to sufficiently determine so that he can apply the known rule. The presentation method (UNKNOWNRULES.PROC046) consists of presenting d-rules that the student doesn't know (step 1), asking him to make an hypothesis about the subgoal (step 2), and then quizzing him about the known rule that uses this subgoal (step 3). If there are any known rules (for determining the unknown subgoal) with unknown subgoals, then these are dealt with before step 1, in a recursive manner. The dialogue excerpt below illustrates this procedure. (In this example, all left-justified output in brackets ("[...]") is trace information that indicates which t-rules are being applied, labels of print statements, and remarks about the student model.)

****

{The tutor and student were discussing "the infection that requires therapy" when the procedure for discussing a goal noticed that a relevant rule (d-rule333) had just fired (based on evidence given to the student). The student model indicates that the student should know this rule (it is a simple definition), however the model also indicates that he can't apply the rule because it mentions a subgoal that he doesn't know how to determine (the infection associated with the CSF culture). GUIDON invokes the opportunistic tutoring procedure for rules that succeed in order to give the student enough information so that he can apply d-rule333.

```
TREATINF
   |
   |
D-RULE333  (recent success that is a definition
   |        and presumed to be obvious to the
   |        student)
   |
INFECTION  (subgoal used by d-rule333; it is known
   |        by MYCIN at this point, but the model
   |        indicates that the student can't
   |        deduce it: he is "missing" a rule)
   |
D-RULE336  (GUIDON assumes that the student
            can't apply this rule (even though it
            is a definition) because it uses
            a list that associates body sites
            with infections.)
```

GUIDON will present d-rule336 to the student and then encourage him to make an hypothesis that involves using d-rule333. If the model is right, information from d-rule336 is all the student needs to wrap up the current goal (TREATINF).}

[Applying tutorial rule 43.01]

[INTRO-RECENTSUCC-TREE:]

We will now consider evidence that leads to a conclusion about the infection that requires therapy.

[INTRO-NEWVALS:]

Here are some conclusions that you may not have considered....

The site of CULTURE-1 is evidence that the infection with which this pending csf culture is associated is meningitis [RULE336].

[Assuming that S can now apply RULE336 CULTURE-1]

[Reconsidering S's use of INFECTION CULTURE-1]
[Applying tutorial rule 41.01 (else)]
[Applying tutorial rule 41.02]
[Student applied (RULE333 CULTURE-1) = 875]
[S can probably apply RULE333 CULTURE-1 now.]

{T-rule packet 41 updates the student knowledge model. GUIDON now chooses a question format (randomly) using t-rule packet 24....}

[Applying tutorial rule 24.01]
[Applying tutorial rule 24.02]
[Applying tutorial rule 24.04]
[Applying tutorial rule 24.05]
[Applying tutorial rule 24.06]
[Applying tutorial rule 24.09]
[Premqtype = FACTS; Actqtype = ACTVAL]

{The question will state the rule in simple true-false form.}

Does the infection with which this pending csf culture is associated tell you that the infection that requires therapy is meningitis?
** YES

[Use-History of RULE333: 0 -> 200]
That's right (see RULE333).

{This turned out to be a trivial question because it is a "definition." Nevertheless, the point is made. This process of presenting "missing" rules and getting the student to use this new information in rules he already knows works to any depth in the search tree. That is, GUIDON will track down and present all of the missing rules and force hypotheses until a conclusion can be drawn about the topic that caused this opportunistic tutoring procedure to be invoked.}

****

It turns out that the tutor doesn't find many opportunities to apply this method of presenting information to the student. This in itself is interesting, because it raises the possibility that the most complex, cleverest tutoring methods will be so

specialized that the cost of formalizing and testing them will outweigh the benefits of using them for efficient presentation of new information. On the other hand, it is a body of rules like t-rule43.01 that lead us to ascribe intelligence to the program. Evidently we are impressed by the specificity of the sequence of goal-directed behavior.

Aside from contributing to the appearance of intelligence, opportunistic tutoring rules embody a particular teaching strategy that is worth considering further: they exemplify the active component of teaching, the presence of mind that leads a teacher to seize the initiative to improve the student's understanding. This suggests a demon-like aspect of teaching, in which t-rules are on the lookout for the right moment to wake up and guide the dialogue by stringing together just the right remarks, gems that might possibly give the student an "Ah-ha" feeling. This perspective also suggests the pragmatic aspect of teaching that is inherent in the mixed-initiative interaction: the tutor must often follow the student and wait for the right moment to say something, rather than incessantly prodding him along. Perhaps the validity of this approach is revealed by the success of the SOPHIE program, in which the tutor *only* reacts to the student's initiative.

To recap, using GUIDON as a model, teaching has two main components: listening to a student in order to understand his reasoning, and trying to control him by correcting, restating, and demonstrating problem-solving methods or strategies. Opportunistic tutoring rules tell GUIDON when to break out of the listening phase and attempt to control the student's reasoning.

Quizzing about Rules that Recently Failed

As the student acquires case data for a goal, many of the rules that use these data will be found to fail. While GUIDON's general strategy is to discuss the rules that succeed when the student's hypotheses for the goal are discussed, the program looks for remarks to make about failed rules at the time that they fail (RECENTFAILS.PROC010). In order to avoid a prolonged interruption, the program ranks the rules that just failed, using the "interest" strategies described in Section 6.2.3.2, discarding rules that are not on the lesson plan or unsuitable for discussion (Appendix B characterizes "unsuitability"). Additional case data are presented if necessary, and a question is posed.

Quizzing about "Related Rules"

At various points in the dialogue, after a given d-rule is discussed, GUIDON looks for related d-rules and presents them to the student (RELATEDRULES.PROC011). Two d-rules are *related* if they mention the same hypothesis in their conclusion and have some premise subgoal in common. *Presenting* the related d-rule currently consists of posing a question. This interruption strategy follows the general principle of trying to associate new information with facts the student already knows. In particular, the "related rules" strategy takes advantage of whatever benefit might be gained from presenting material that can be associated with subject matter that the student already holds in mind. (This might be a bad idea; maybe connection to previously learned material is more effective when *long-term* memory associations must be reconsidered.)

Opportunistic Tutoring Experiments

Modifying opportunistic tutoring strategies is perhaps the most likely way to experiment with GUIDON. Experimentation might shed light on:

1) The pedagogical and psychological ***effectiveness of alternate presentation and quizzing methods*** (is the recursive approach of the "unknown subgoals/missing rules" strategy acceptable?),

2) Constraints on the ***number of interruptions*** in a given time (similar to considerations useful in the WEST program), and

3) The ***cost/benefit*** parameters for very specific methods that are infrequently used (and hence pose system overhead), yet might be effective because of their directness.

4) The affect of tutorial remarks on ***student initiative*** (do interruptions encourage or inhibit exploration?)

6.2.3.4 Tutoring: Responding to a Student's Hypothesis

At any point in the dialogue the student can state an hypothesis for any subtask. GUIDON constructs a tree that relates each of the values of the hypothesis (e.coli, pseudomonas) to the rules that MYCIN used to form its own hypothesis. Only the d-rules that MYCIN has been able to apply up to this point in the dialogue are considered. The complications of the modeling process are described in Section 8.3.2; here we assume that the values have been grouped according to their correctness, and rules have been associated with each value according to whether the rule provides evidence for the value or provides evidence against it. The

strategy of the "hypothesis response" t-rules (HYPOTHESIS.PROC014) is to quiz the student about evidence that the student may probably know or may have forgotten. When the student model is strong (either claiming that the student knows a rule or does not know it), the tutor directly states evidence without quizzing.

Evidence is classified into many groups, partly on the basis of the logic of the model, and partly on the basis of distinctions that will be useful for tutoring. These groups are:

1) *correct* hypothesis values

a) all rules that MYCIN used are probably known by the student (model CF > .7).

b) some rules may be unknown (0 <= model CF <= .7).

c) some rules are probably unknown (model CF < 0).

2) ***wrong*** values (hypotheses MYCIN explicitly rejected)

3) ***missing*** values (significant hypotheses that the student did not mention)

4) ***significant rejected*** values (CF < -.2) that the student did not mention

5) ***insignificant*** values (hypotheses MYCIN has evidence for, but does not believe [-.2 <= CF <= cutoff])

6) values that are ***not concluded*** at all by the rules that MYCIN used

7) ***insignificant*** values that the student did not mention (currently ignored by GUIDON)

GUIDON deals with each group of values in the order given. The chief presentation method (see HYPOTHESIS.PROC014, HYPVALRESPONSE.RULEPK025, and PRESENTVALS.RULEPK026) consists of asking the student to supply a list of factors that support a specific hypothesis, be it correct, wrong, etc. This requires special question types that construct a question using a list of rules (Appendix B). Special t-rules single out cases involving a single (known or unknown) d-rule for economical presentation; for example, when it appears that the student made a wrong hypothesis because he is ignorant of a single rule, the program confirms the evidence that the student probably knows and then poses a direct question about the missing rule (see t-rule25.01).

In short, GUIDON's response to an hypothesis is an attempt to be logical and thorough. The main difficulty we have observed with this strategy is that the student is often still digesting the response to his wrong values, when GUIDON is already posing questions about the missing values, the "no evidence" values, and so on. In the protocol, the student wanted to get back into data collection mode before hearing more about his hypothesis. While it would be easy enough to tell the program to skip the rest of the evaluation, there are real problems if the student wants to return to hear more about his recent hypothesis.

The problem with student interruptions during the hypothesis evaluation procedure is that many of the internal lists set up during the modeling phase would need to be updated if the student finds out more about the case between questions. That is, if another rule fires, then the tree of evidence that supports and contradicts the student's hypothesis will be out of date. Since this tree controls which

"hypothesis response" t-rules are applied and is used to generate questions about current evidence, the tutor's remarks will not be in accord with the current knowledge model.

It is of course possible to update the evidence tree (data structure) before returning to finish hypothesis evaluation, but the problem involves more than time-consuming computation. In general, tutorial decision-making might involve carrying out dynamic plans (sequences of presentations), as opposed to following prepared procedures that rigorously preserve and maintain assumptions about the discussion and inference records. GUIDON's hypothesis response procedure does not exactly *plan* what it is going to do, but its modeling stage does implicitly enable certain actions that are to follow. Already we see that exchange of initiative in the tutorial involves complex interactions with the knowledge models and discourse procedures, given that student interruptions can disrupt a plan at any time. It appears inevitable that modeling and tutoring will have to go on in tandem. Specifically, in presentations of any considerable length, like hypothesis evaluation, it will be desirable for the tutor to periodically stop, listen, and reevaluate its present course.

6.2.4 *Situation Transitions*

Clarity of presentation is critical in productive teaching, as in any dialogue. In GUIDON tutorials this involves providing coherent transitions between dialogue situations and comprehensible summaries of inferences.

Providing clear transitions requires that discourse procedures take into account

the context in which they are invoked, so that the tutor's remarks are not disconnected or redundant. The chief design principle we have followed is that a tutorial rule that invokes a new discourse situation provides transition to the new situation and return to the former context. For example, T-RULE 5.03 prepares the student for the request for an hypothesis. Note that the predicate "an introductory remark is to be made..." is a flag set by the calling discourse procedure, e.g., it is false if the student has explicitly requested the opportunity to make an hypothesis.

Figure 6.12 illustrates the kind of disconnected dialogue that arises from inadequate communication between discourse procedures. Following the procedure that is invoked after an hypothesis ("advice" in Figure 6.3), the tutor asks the student if he wants to continue discussion of this topic or if he would like to wrap it up at this point. The student wishes to continue, so the dialogue shifts to discussion of this topic (the procedure GOAL), and the first thing the tutor remarks about is that the topic is essentially done (attempting to make an economical transition based on a comparison of the expert program's current hypothesis to the final conclusion it will reach after all evidence has been considered).

```
.
.
.
** HYPOTHESIS 3B

Is the meningitis partially treated?
** YES

You're right.

The infection which requires therapy, the fact that organisms were not seen
on the stain of the pending csf culture (CULTURE-1) and the time since
therapy with the cephalothin (DRUG-1) was started is evidence that the
meningitis is partially treated (.95) [RULE145].

It remains to collect further evidence for determining whether the
meningitis is partially treated.
Would you like to see a summary of all of the evidence that can eventually
be collected (as opposed to working it out yourself)?
** NO

Would you like to see a summary of the evidence we have discussed before
going on to ask more questions?
** NO

We have substantially completed our discussion of whether the meningitis is
partially treated.
Would you like to make an hypothesis now (as opposed to gathering more
information)?
** NO
.
.
.
```

Figure 6.12. A Disconnected Dialogue

A smoother dialogue would result if the procedure for giving advice after an hypothesis looked to see if the topic was substantially completed and so the student would only be asked once if he wanted to go on.[1] The interaction shown here is typical, and many of the changes made to the system after each trial tutorial session

[1] This change has been incorporated in the ADVICEAFTERHYP.PROC039 procedure, and is manifested in the protocol (page 58).

are designed to improve the dialogue in similar ways. While we are constantly allowing for more subtle variations in the dialogue, the framework of rules organized into procedures is working well.

Finally, when tutorial remarks are limited to lines of print on a teletype page (a constraint this research has adopted), there is no opportunity to pause or to emphasize words (by changing voice volume or inflection). Therefore, it is worthwhile to incorporate visual keys that help to organize presentations, and provide clear transition between segments of the dialogue. For example, in GUIDON, we make use of various designs that are intended to provide familiar transitions, making it easier for the student to anticipate what will be expected of him (e.g., the use of ">>>" before the remark that the expert has drawn a conclusion).

When summarizing evidence, GUIDON regroups goal/subgoal relations, separating key factors from contextual information. Grouping the display of goal/subgoal information can make the large amount of evidence easier to read, as shown on page 51. Moreover, only the key factors of domain rules are printed here, making it much easier to see the essential evidence (see Section 5.2.4.1 for discussion of key factors). Thus, the program omits clauses that are common to the set of rules that are being mentioned, stating the omitted clauses at the end, as a reminder of the context of the inferences.[1] It is significant that the usual, MYCIN-style listing of these rules would require several pages.

[1] Goldstein [Goldstein, 1977] has considered the problem of simplifying multiple inferences to clarify arguments. We have not yet found this to be necessary for MYCIN-like rule bases.

6.3 Summary of Tutoring Principles

The following is a list of principles or general strategies used in GUIDON's t-rules. Observe that a given principle may be exhibited in many individual rules, allowing for the various situations in which it is applied. For example, there are more than a dozen t-rules for discussing a domain rule economically, based on what constitutes economical presentation in different dialogue situations (number of subgoals left to discuss, complexity and sophistication of the rule). In general, for a given tutorial goal, several, sometimes competing, principles must be considered at once, so there is a matrix of considerations, rather than a single t-rule for each principle.

1) ***Be perspicuous***: economical presentation strategy (Section 6.2.1); lucid transitions (Section 6.2.4); adherence to conventional discourse patterns (Section 6.1.3.1).

2) ***Account for incorrect behavior in terms of missing expertise*** (as opposed to alternative methods and strategies). Explain clearly what is improper from your own point of view. (E.g., coping with improper requests for case data, Section 6.2.2.1.) (This is of course more a statement of how GUIDON models the student, rather than a principle of good teaching.)

3) ***Strictly guide the dialogue***: say when topics are done and inferences are completed, as opposed to letting the student discover transitions for himself (Section 6.2.2.2).

4) ***Provide orientation to new tasks by top-down refinement***: Provide the student with an organized framework of considerations he should be making, without giving away the solution to the problem (important factors, subgoals, size of the task), thus challenging him to constructively examine his understanding (Section 6.2.3.1).

5) ***Provide assistance by methodically introducing small steps*** **that will contribute to the problem's solution (Section 6.2.3.2).**

1) Assistance should at first be general, so as to remind the student of solution methods and strategies he already knows.

2) Assistance should encourage the student to advance the solution by using case data he has already been given.

6) Take advantage of opportunities to ***examine the student's understanding and introduce new information*** (Section 6.2.3.3).

7) When responding to partial student solutions, ***probe the student's understanding when you are not sure what he knows***, otherwise directly confirm or correct the solution (Section 6.2.3.4).

6.4 Dialogue Management Conclusion

While most ICAI research has focused on representation of domain expertise and construction of a student model, we have shown that there is a group of issues that center about the problem of carrying on a coherent, task-oriented mixed-initiative dialogue with a student. We have named this collection of issues the ***dialogue management problem*** and discussed it in terms of tutorials based on MYCIN-like rules. We presented a representation of dialogue knowledge in the form of a transition diagram in which the nodes are discourse situations and the links represent selection of alternative dialogues based on domain logic, economy of presentation, and tutorial objectives. Other issues, to be discussed in Chapter 7, are managing topic relationships and sharing initiative.

We stated that the main forcing function behind improvement of the program was the impossibility of explicitly discussing each inference made by the expert program. As illustrated in Chapter 9, verbosity continues to be a problem for GUIDON. It is interesting to note that the most effective pruning of topics in the current version of the program is a result of the student's initiative. Perhaps it is not necessary or desirable for the program to attempt to manage the dialogue too severely; deciding what should be discussed is naturally a shared task. We believe that a proposed case lesson plan (Section 10.4.1) will complement the procedures for non-exhaustive discussion by providing reasonable, time-sensitive goals for the session.

Chapter 7

Providing for and Coping with Student Initiative

CONTENTS

7.1 Sharing and Maintaining Context
- 7.1.1 Getting Case Data
  - 7.1.1.1 Options: DATA, BLOCK, ALLDATA
  - 7.1.1.2 Focusing for Data Requests
    - -- crossing paths
    - -- deeper subgoal
- 7.1.2 Inference and Discussion Record
- 7.1.3 Controlling Tutorial Remarks
- 7.1.4 Direct Retrieval of Domain Knowledge
  - 7.1.4.1 FACTORS, SUBGOALS, & PENDING
  - 7.1.4.2 USE & CONCLUDE
  - 7.1.4.3 DETAILS
    - -- remark indexing scheme
    - -- details for a factor
    - -- ... for a subgoal
    - -- ... for a rule
    - -- ... for a final conclusion
  - 7.1.4.4 QA
  - 7.1.4.5 SUMMARIZE
- 7.1.5 Dialogue Context
  - 7.1.5.1 Options: RULE, TOPIC, and ORIENT
  - 7.1.5.2 The Focus Record

7.2 Conveying what you know: IKNOW and HYPOTHESIS

7.3 Requesting assistance: HELP/HINT, QUIZ, ALLRULES, TELLME

7.4 Changing the topic: DISCUSS, TELLME, control-F interrupt

7.5 Standard user options: COMMENT, PROFILE, ?, OPTIONS

An essential part of tutorial dialogue management is allowing the student to express himself. To construct a mixed-initiative program, provision must be made for every potential kind of initiative that the student will be allowed to make. This includes being able to refer back to an early topic and provide more details (given that explicit presentation of evidence may be non-exhaustive), change the topic, and so on. We might summarize this by saying that we must allow the student to specify: what he knows, what he wants to know more about, and what he wants to ignore.

In addition, a tutor must cope with the initiative it recognizes. This means that the tutor should, to some degree, respect student initiative by patterning its own actions according to the lines of thought taken up by the student. For example, when assistance is provided, it should reflect what the student has done so far to solve the problem. Responsive tutoring requires continuous monitoring and record-keeping about the context of the dialogue.

GUIDON provides for student initiative in two related ways. First, a wide variety of options are available to the student, and second, many tutorial remarks are indexed so that the student can easily refer to them later (use them as arguments to options). To relate its own remarks to the student's, the tutor maintains simple records of the context of the dialogue, consisting of subgoals, hypothesis values, and inferences that were recently mentioned.

The options available to a student are summarized in a table that he can print during the tutorial (listed in Figure 7.1). As described below, each option generally has a discourse procedure associated with it. For example, the procedure for IKNOW

invokes HYPOTHESIS.PROC014 if the expert program hasn't made a final decision yet; otherwise COMPLETEDGOAL.PROC009 is invoked. Simple English phrases are also accepted by the program ("How can I conclude that the infection is viral?"). However, we believe that, for the student community with which we are working, this option format (often called menus) will prove to be more convenient.

Option type	Summary of options
Get Case Data	DATA BLOCK ALLDATA
Information Retrieval	FACTORS SUBGOALS PENDING USE CONCLUDE DETAILS QA SUMMARIZE
Dialogue Context	RULE TOPIC ORIENT
Convey What You Know	IKNOW HYPOTHESIS
Request Assistance	HELP HINT TELLME QUIZ ALLRULES
Change the Topic	DISCUSS STOP control-G
Special	? PTR PR JUSTIFY LITERATURE DEFINE COMMENT PROFILE OPTIONS

[Key: PR = print domain rule; PTR = print tutorial rule;
control-G = give initiative to the student; QA = enter MYCIN's
question-answer module; ? = "what is valid here?"
See Appendix C.]

Figure 7.1. Options Available During GUIDON Dialogues

The discussion that follows is organized according to these option groups. In Section 7.1 options for getting case data, retrieving information, and establishing the current context are related under the topic of sharing and maintaining problem-solving context: what is known so far?, what remains to be done?, what are we doing now? Subsequent sections discuss options by which the student conveys what he

knows, requests assistance, and changes the topic. Finally, we will consider some standard options that make the program easier to use.

7.1 Sharing and Maintaining Problem-Solving Context

The problem-solving context is represented in GUIDON by the AND/OR solution tree. The nodes are annotated as the dialogue proceeds to keep track of what has been mentioned to the student and what part of the tree has been completed by the expert program. Clearly, the overlay model is part of the record of problem-solving context; but because of its complexity we will consider it separately in the next chapter. The options considered here are designed to provide structured information to the student. The instructional goal is to provide windows on the solution that will encourage the student to explore his own understanding by raising curiosity about the expert's approach.

7.1.1 Getting Case Data

7.1.1.1 Options: DATA, BLOCK, and ALLDATA

The student can request case data by either giving the keyword name of the parameter, optionally prefaced by the command DATA. The "context," e.g., ORGANISM-1, is optional when it is unique. Blocks of data, a group of related tests, can also be requested. "BLOCK ?" will print all recognized block names. Note that the procedure for supplying case data (GIVENPARMDATA.PROC012) will automatically provide all related tests, once one is requested. Similarly, when a screening

parameter (e.g., surgery) is requested, the data it screens (neurosurgery) is provided as well. Finally, the student can ask GUIDON to print all of the case data that MYCIN requested (for the current goal) that hasn't been given to him yet by using the ALLDATA option.

Section 8.3.2 describes how the overlay model is updated when case data are given to the student. Below we consider rhetorical patterns for refocusing the dialogue when a data request violates the goal-directed nature of the dialogue.

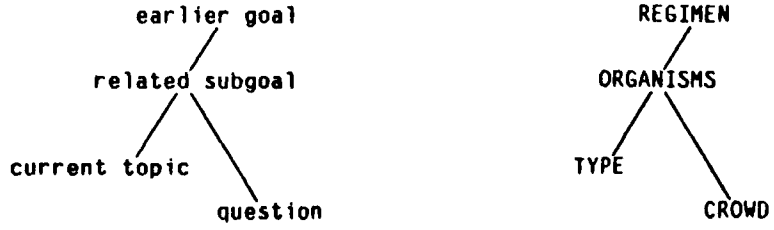
7.1.1.2 Focusing for Data Requests

Student requests for data can implicitly change the topic if the datum requested is not relevant to the current topic. If the datum cannot be used directly in any inference, that is, it is not a factor of the current goal, then it is not "relevant" to the current goal. If the question is unnecessary, the student is told why (as described in Section 6.2.2.1). Otherwise, the datum must be a factor of a deeper subgoal or a shallower goal. page 59 illustrates what occurs when the requested information is relevant to a deeper subgoal.[1] If the requested information is relevant to a previous and shallower goal, the tutor states this relation so that it is clear to the student what topic is currently being pursued (Figure 7.2).

In making these distinctions and connections for the student, the program maintains shared focus on the current goal while respecting the student's specific interest in other aspects of the case. It is here that the goal-directedness of the

[1] If this is not unique, the program chooses one arbitrarily.

dialogue is enforced; though note that the student can explicitly change the topic at any time by the DISCUSS option.



** DOES THE PATIENT LIVE IN A CROWDED ENVIRONMENT?

Pt538 does not live in a crowded environment.

Whether the patient does live in a crowded environment is not relevant to determining the type of the infection. It is a consideration we can use later when we return to our discussion of the organisms (other than those seen on cultures or smears) which might be causing the infection.

Figure 7.2. Crossing Paths in a Tree of Subgoals

7.1.2 Inference and Discussion Records

The inference and discussion records consist of marks placed on the AND/OR solution tree. The inference record, which indicates which rules have been applied by the expert program and the conclusions made by these rules, is a component of the overlay model; it is described in Section 8.3.2.2. The discussion record indicates which (case-specific) inferences have been mentioned in the dialogue and which goals have been completed (the tutor has mentioned MYCIN's final conclusion).

This record is also useful to the student who may find it difficult to keep track of the evidence that has already been discussed and what remains to be considered: two critical aspects of dialogue management. Without an accurate record of the

status of the dialogue, it would be difficult for the participants to reach a consistent, final agreement about the topics under discussion. Figure 7.3 illustrates one of the several options by which GUIDON shares its record with the student.

```
** CONCLUDE VIRAL

Using the following factors, it will be possible to conclude that the type
of the infection is viral:

     10a.  the duration of the neurological symptoms [RULE526]

     10b.  the white count from Pt538's peripheral CBC (in thousands) and
           the WBC from the CSF [RULE504]


Do you want to see contrary evidence?
** YES
Using the following factors, it is possible to conclude that the type of the
infection is not viral:

     11a.  the WBC from the CSF and the percent of PMNs in the CSF [RULE501]

- - - - - - - - - - - - -

Do you want information about evidence that we have already discussed (if
any)?
** YES
Using the following factors, we have already discussed how to conclude that
the type of the infection is viral:

     12a.  the duration of the neurological signs [RULE524]


Do you want to see contrary evidence?
** NO
```

Figure 7.3. Keeping Track of Evidence for an Hypothesis

In this example of the CONCLUDE option, observe that the program refers to incomplete inferences (RULE526), new evidence (RULE501), and evidence that has already been discussed (RULE524). Some tutorial strategies are designed to directly offer information like this to the student. For example, after a rule is discussed

explicitly, GUIDON states the cumulative hypothesis that has been discussed for that goal.[1] Examples of related information retrieval options are given in Section 7.1.4.

7.1.3 Controlling Tutorial Remarks

Another student-controlled means of sharing status information is through the student profile. At the start of a tutorial session the student specifies whether he wishes to be told about conclusions being made by the expert program (MYCIN) in the course of the dialogue. He can request nothing, ask for full information, or simply request a "hint," the suggested usage. See protocol page 43.

The components of the profile are:

Student sophistication

The student's estimation of his experience is matched against the a priori sophistication ratings for each rule. See Section 8.2.1.

Failure or Success of D-rules

A trace of domain rule applications during the tutorial provides a detailed window on each decision made by MYCIN. It is of much less value than the more abstract descriptions given by other profile options.

Rule Conclusions

Successful rule conclusions will be stated directly, held until the relevant topic enters the discussion, described in hint form, or suppressed. A hint simply states that a conclusion can be made, as shown in this excerpt:

---

[1] However, GUIDON does not model the student's forgetting during the session, so this information is given to every student every time.

```
The duration of the neurological signs is 7.2 hours.

>>> 37a.  MYCIN just made a conclusion about the type of the infection.
```

In terms of teaching strategy, this kind of remark is intended to encourage the student to examine his own understanding at this point, perhaps leading him to request more information.[1]

Final Goal Conclusions

Final conclusions for a goal will be stated directly, described in hint form, or suppressed. In many cases, this window on the expert program is less useful than the individual rule conclusions which can be associated with specific case data and completion of intermediate subgoals, as illustrated above.

Lesson Plan

The default lesson plan will tend to restrict tutoring (quizzes) to rules that are within two levels of the student's sophistication rating. This default provides a crude approximation of the selective tutoring goals that the lesson plan is ultimately intended to represent.

Debugging

Debugging information includes a trace of t-rule application, as well the setting of some important modeling and focus record variables. This option is of course not intended for students.

On the basis of limited experience, we feel that hints about rule and goal conclusions are not intrusive because the expert program is only infrequently able to apply a domain rule successfully during the dialogue. Two heuristics tend to prevent

[1] "DETAILS 37A" will open up discussion of the relevant rule; see Section 7.1.4.3.

redundant or useless remarks of this kind. First, a final conclusion is not mentioned if no rules were applied (this prevents the tutor from saying that MYCIN has made a final conclusion about case data immediately after the value is given to the student). Second, intermediate rule conclusions are not mentioned if only one rule succeeds for this goal (this prevents the tutor from saying that MYCIN has made a conclusion and then saying right away that MYCIN has made a final conclusion for the same parameter).

7.1.4 Direct Retrieval of Domain Knowledge

Here we consider the most basic options for exploring the AND/OR solution tree. A variety of options provide different perspectives on what has been accomplished in the task and what remains to be done. All of the options are interpreted in the context of the goal being discussed. Most of these options print remarks that are indexed for later reference by the student. Remark indices are discussed in the section about the DETAILS option.

7.1.4.1 FACTORS, SUBGOALS, and PENDING

The FACTORS option lists *all* parameters that appear in rules for the current goal. (This is the union of CONSEQ-SG!, FAIL-SG!, and ANTE-SG!, with duplicates removed.) Elements are starred (*) if they have been discussed. Entries that have not been discussed, but for which MYCIN has made a final conclusion, are mentioned at the bottom of the listing. This feature is intended to prompt the student to use the DISCUSS, TELLME, IKNOW, or HYPOTHESIS options to finish up these subgoals.

The SUBGOALS option prints the subtree with the current goal at the root. Subgoals are grouped (by indentation) under the goal to which they are relevant. Parameters that are determined by case data alone are not mentioned, only those factors that are determined by one or more rules.

The PENDING option prints only those FACTORS that have not been discussed. The listing is in three parts: subgoals that remain to be discussed, case data that remain to be requested, and subgoals for which the student has enough information to form a final hypothesis.

All three of these options take as an optional argument the name of a goal. Note that this information would not be available if MYCIN had not completely worked the case before the tutorial began. By reconfiguring MYCIN's rules into the explicit tree representation, it is computationally trivial to display goal and data relationships. If GUIDON did not have access to the final inference structure, it would only be possible to say what had been determined so far (starred FACTORS). The tutor could look ahead as MYCIN does in a consultation when asked HOW some incomplete goal will be achieved [Davis, 1976], but this might be slow and would not provide the map-like perspective that can be directly derived from a complete solution trace. On the other hand, human tutors don't always proceed from a worked-out solution, so we certainly would not insist that GUIDON's style is the best: it is the advantage of being able to lay down an an exhaustive itinerary of points to explore that is of interest.

7.1.4.2 USE & CONCLUDE

The USE and CONCLUDE options have evolved directly from their MYCIN question-answer program equivalents: How do you use a piece of information? How do you make a conclusion? Rules that *use* a factor are determined by finding the subset of successful rules (UPDATE-SUCC! of the current goal) that require the factor (SUCC-PARMS! of the rule). Note that GUIDON indicates the conclusions that will be made using this factor, and does not refer to failed inferences. This is the information that was deemed to be most useful to the student.[1]

Rules that *conclude* a value are determined by again examining the successful rules, in this case looking at the CONCLUSIONS property of the rules. Again, failed inferences are not mentioned. As discussed earlier (Section 7.1.2), GUIDON separates discussed conclusions from partially completed inferences and those that are known to MYCIN but have not yet been discussed. In addition, evidence that contradicts this hypothesis value is printed if the student is interested.

7.1.4.3 DETAILS

Remark indices are recorded in a list that is indexed by the number of the remark and labelled by one of a dozen remark kinds, e.g., a subgoal, or a rule that is mentioned in a summary of evidence for a topic.[2] They allow easy, unambiguous

---

[1] Complications arise when no rules succeed, when many rules succeed (the rule model is printed instead), and when the goal rule is involved (leading to giving away the case solution).

[2] The list is called INTERACTION.HISTORY. Elements are of the form: (⟨interaction no.⟩ ⟨remark label⟩ {optional arguments}).

references, greatly contributing to the flow of information between tutor and student. A remark index that stands for a goal can be used for any option that takes a goal as an argument, as can be seen throughout the protocol.[1]

The procedure for providing DETAILS knows how to provide details for any remark kind.[2] Here we list each kind of remark, indicate when the tutor prints it, and summarize the details provided.

LISTS: A domain list that appears in rules for a goal.
Printed by: the FACTORS option.
Arguments: name of the list, the goal
Details: definition of the list, example elements of the list, values concluded by rules that mention the list.

FACTORS: Parameter of rules for a goal.
Printed by: FACTORS option.
Arguments: the factor
Details: definition of the factor, how the factor is USEd.

FIRED: Rules that have been applied by MYCIN, but possibly not discussed.
Printed by: Summary of evidence for a goal, and CONCLUDE option.
Arguments: the rule
Details: For each rule factor that has not been discussed, offer the student a summary of evidence for determining the factor. Offer to print the rule.

PENDINGRULE: Rule that has not been applied.
Printed by: CONCLUDE option.
Arguments: the rule
Details: offer to discuss the rule clause-by-clause

SUBGOALS: Subgoal for a goal.
Printed by: SUBGOALS option, sketch of tree.
Arguments: the subgoal, the goal.
Details: Offer to show how the subgoal is USEd. Offer a choice of discussing the subgoal in detail or seeing the final conclusion and summary of evidence.

FINALCONCL: Final conclusion made by MYCIN
Printed by: profile option
Arguments: the goal
Details: If this is the current topic, suggest that the student use the TELLME option if he really wants to end discussion.

---

[1] If necessary, the procedure for converting remark indices to option arguments converts the history information to the proper form.

[2] This was originally implemented as regular INTERLISP code, but we find now that it can be easily converted to the format of discourse procedures.

```
If the goal was marked as discussed before MYCIN had a
chance to apply all of the rules, offer to discuss them
(uses NOTDISCUBRULES.PROC007).
If the goal has already been discussed normally, express
surprise and display the final conclusion and offer a
summary of evidence (FINISHGOAL.PROC006).
Otherwise, offer a choice of seeing the final conclusion
or discussing the goal in detail.
```

RULECONCL: Hint or description of a rule that succeeded.
Printed by: Profile option
Arguments: the rule, the goal
Details: Print the key factors and conclusion.

7.1.4.4 QA

MYCIN's question-answer facility (Section 4.3.1) is accessible from GUIDON. All of the standard options are available. The main reason for using the QA is to ask general questions about the rule base (How does MYCIN conclude whether an organism is a contaminant? What organisms are normally found in the mouth?). The only other feature of the QA not generally provided by GUIDON is that failed inferences are routinely mentioned (GUIDON prints them when it has nothing else to say).

7.1.4.5 SUMMARIZE

The final information retrieval option, SUMMARIZE, prints the evidence that MYCIN used to determine a goal. Key factors and conclusions of each rule are printed. (This might be contrasted with the summary method used in hypothesis evaluation which groups the evidence by hypothesis value.) GUIDON's summaries are better than a list of rules, but could be improved if the program abstracted the most important

evidence, especially by describing basic lines of reasoning in the rules. For example, when summarizing the final diagnosis, the program might say, "Organism-1 will not be treated because it is a contaminant," thus delving into the subtree and picking out a key factor from the rule that tipped the scale. Summarizing negative decisions could involve more than listing half-a-dozen weak factors. To some degree GUIDON's description of evidence could be improved by simply substituting prosaic characterizations for the certainty factors, e.g., "weak evidence" or "confirming evidence."

7.1.5 Dialogue Context

Previous sections have dealt with sharing context about the problem being solved (domain facts, what remains to be done). Here we will treat the topic history of the dialogue as another context to be shared (motivating the current topic, relating assistance to recent topics).

7.1.5.1 Options: RULE, TOPIC, and ORIENT

The student can directly request a description of the current topic by the TOPIC option. The goal that is being pursued at that time will be printed, with a list of typical hypotheses that might be considered. The hypothesis list is sometimes the best description of the current goal. "We are discussing the type of the infection" may be vague to a student: how does this differ from the *name* of the infection? The additional remark, "You should try to determine if the type of the infection is bacterial, viral, fungal or tb" is a better statement of the current goal (page 67).

The RULE option will state the rule number that is being considered. This information is available to the program by rebinding the variable CURULE when a discourse procedure begins discussion of a rule, e.g., this is done in RULE.PROC004 and SUGGESTRULE.PROC031.

The ORIENT option prints the stack of goals and d-rules that are being pursued (page 48). Discourse procedure macros (DISCUSS\\, STRATEGY\\) record their arguments on a history stack. A special procedure reads this stack, extracting calls to GOAL.PROC001, RULE.PROC004, and active t-rules (indicated by TRULE in the stack). Finally, the deepest (most recent) t-rule is retrieved from the source file and steps of the rule's action that remain to be performed are printed, thus demonstrating the capability to look ahead and say what will happen next.[1]

The current implementation of the ORIENT feature will probably be of little value to a student. Instead, it was implemented to demonstrate the possibility of having GUIDON answer questions about its teaching strategies during the tutorial. That is, the well-structured, compiled code will make it possible to implement a HOW and WHY explanation system, as in the MYCIN consultation system. The reasons for wanting GUIDON to be able to explain its reasoning are the same as for any other knowledge-based program: for a human to modify the program he must understand the expertise that the program is currently using as a basis for its decisions [Barr, Bennett, & Clancey, 1979]. Thus, the ORIENT option demonstrates the feasibility of developing

[1] Each STEP\\ and SUBSTEP\\ of a discourse procedure marks the history tree. Like all entries in the tree, this mark is removed (popped) when the operation is complete. Thus, GUIDON finds the *current step* number in the tree, e.g., (TRULE 48.04 (SS iv. (D! GOAL.PROC001 (MUMPSYM PATIENT-538))) indicates that step 4 of t-rule48.04 is being followed and the current topic is "symptoms of mumps."

an explanation and knowledge acquisition package for interactively modifying t-rules. It may also be useful for the tutor to be able to explain to the student why a certain topic is being pursued in the dialogue.

7.1.5.2 The Focus Record

GUIDON keeps a simple record of topics in which the student seems to be interested, and uses this record when taking the initiative to direct the dialogue. The idea is to adapt, at least in a simple way, the tutor's choice of topics to those the student may want to know more about. The **focus record** consists of variables that are set when the student uses various options. It is used by the tutor to select a domain rule to discuss when the student explicitly requests help or a hint. For example, the USE option mentions a topic (subgoal to the current topic) and the CONCLUDE option mentions an hypothesis (for the current topic). These can be used to index the rules for the current topic.

In addition, GUIDON sometimes has to keep track of its own remarks so that it does not repeat itself, and can follow up properly. For example, the program provides a HINT by describing the hypothesis that is supported by a rule that it would like the student to consider. When the student next requests HELP or another HINT, we don't want the program to repeat this hint, but instead want it to do something else with the chosen rule. A simple variable records that a rule has been mentioned to the student in the form of a hint, and tutorial rules for providing help look for this situation in order to follow-up appropriately (Figure 6.9). The means for keeping the dialogue focused are admittedly simple, but because of the other constraints on the dialogue (i.e., stacked topics and the option-oriented format), they seem to be sufficient.

7.2 Conveying What You Know: IKNOW & HYPOTHESIS

A student may directly offer an hypothesis for any goal at any time by using the hypothesis option. It is interesting to note that GUIDON uses MYCIN's consultation code to state a question and parse the values and certainty factors of the answer. The actual response to the hypothesis is given by the procedure HYPOTHESIS.PROC014, as described in Section 6.2.3.4.

The IKNOW option means "I don't think that we have to discuss this goal in detail because I've already figured out a solution." The rule packet IKNOWOPTION.RULEPK044 responds according to the status of the goal: whether it has been discussed, whether it is unrequested case data, and whether or not MYCIN has reached a final conclusion. If MYCIN is not done, the student is requested to state his hypothesis (t-rule 44.03). Otherwise, the procedure COMPLETEDGOAL.PROC005 will use the student model to decide whether the student's hypothesis should be discussed.

We believe that these strategies illustrate one of the strengths of the GUIDON t-rule framework. Given a rich context (the student says he knows something), a set of very specific t-rules can be written to detect alternative situations and then cause the tutor to act in a very precise, and hence apparently intelligent way. The framework is convenient because it focuses our attention (as teacher-experts) on a particular dialogue situation (signified by the rule packet and conditions for its invocation), and asks us to consider how we would *act* in this situation. Production rules appear to be particularly suitable for encoding this kind of "what would you do

when" knowledge. T-rules provide a language for stating tutorial dialogue actions: start up another dialogue pattern, say something, or make a record.

7.3 Requesting Assistance: HELP/HINT, QUIZ, TELLME, ALLRULES

The procedure for providing help (HELPFORGOAL.PROC016) is described in Section 6.2.3.2. Quizzes are generated by ranking the rules that have been applied by MYCIN, using the same "interest" strategy as for providing help. A question is then generated from the rule (Appendix B). If the student requested a hint, the "most interesting value" (Section 6.2.3.2) is stated as the hypothesis to be pursued.

The ALLRULES option directs the tutor to discuss the rules for the current topic. Subgoals will be set up and discussed in the normal way. This option leads to an overly long dialogue if there are many rules to be discussed (say, more than 5).

If the student finds himself short of time, bored, or hopelessly unable to work out a subproblem, the TELLME option will provide an immediate end to the topic, with the final conclusion and a summary of evidence provided upon request.[1] Like the procedure for dealing with the IKNOW option, TELLMEOPTION.PROC045 makes a number of distinctions based on the status of the topic the student would like to "close out." The TELLME option is particularly useful to get past some of MYCIN's intermediate decisions that are of little tutorial value, for example, the step that combines the list of hypothesized unknown organisms from cultures with the list of organisms that will be treated based on clinical evidence alone (page 82 and following).

[1] If he wishes to terminate the session itself, he can type STOP.

7.4 Changing the Topic: DISCUSS, TELLME, control-F interrupt

The student can direct the tutor to open discussion of a topic by the DISCUSS option. The procedure PLEASEDISCUSS.PROC022 manages the transition between old and new topics, by commenting on the relation of the new topic to the old, printing a design to mark off the shift in focus, and by, as usual, explicitly saying which transitions are taking place.

The TELLME option is included in this list because it might also be used to change the topic. Thus it might mean "let's move on, I've had enough" as well as "help me get out of this situation, I can't deal with it today."

By typing a control interrupt (control-F) the student can signify that he would like to say something. Currently, this interrupt character causes a flag to be set that is examined after the tutor poses a question. This is a particularly good time to check to see if the student has a question, especially since the program may have embarked on a long sequence of posing questions, as when it responds to the student's hypothesis for a subproblem. If the flag is set, the student is prompted for input, but his options are restricted to data retrieval and special options, like making a comment or changing his profile. The problem of letting him request case data or change the subject during an interrupt is discussed in Section 6.2.3.4.

7.5 Standard User Options: COMMENT, PROFILE, ?, OPTIONS

Options that we now consider to be standard for a program like GUIDON include the ability to file a comment about the program, to change the user profile, to get help for using an option (text, plus sample arguments), and to list valid options.

In displaying valid options we have been influenced by the work of Nelson ([Nelson, 1974], pp. DM49-50). We group the options according to the kind of operation the student may have in mind (saying that he knows something, changing the topic, getting case data, retrieving MYCIN facts) (see Figure 7.1).

BLANK

Chapter 8

Contents, Maintenance and Use of the Student Model

CONTENTS

8.1 Contents of the model

8.2 Sources of Evidence
- 8.2.1 Background Evidence
- 8.2.2 Implicit Evidence
- 8.2.3 Explicit Evidence

8.3 Growth and Revision of the Model
- 8.3.1 The Components of the Model
- 8.3.2 The Overlay Model
  - 8.3.2.1 Example of Modeling Process
  - 8.3.2.2 Implementation
  - 8.3.2.3 Deciding Whether the Student Applied a Rule
  - 8.3.2.4 Complications
- 8.3.3 Revision of Belief

8.4 Uses of the Model

8.5 Inadequacy of Performance Knowledge for Modeling Misconceptions

In any dialogue, it is useful for the speaker to have a model of what his listener knows, is thinking about, or would like to know more about. This is particularly true in a tutorial dialogue, in which the tutor's purpose is to be informative. In order to know what is new to the student, and so worth saying, the tutor will have to analyze the student's behavior for clues about what he knows. This is the problem of ***assigning credit*** to domain facts, relations and strategies to explain the student's behavior. We say, "he is asking about the chest x-ray, perhaps he is considering pneumonia." Or, "he says that the organism is e.coli, he must not have considered the gram stain." A second problem is to ***revise the model*** on the basis of new evidence. For example, if the tutor tells the student the correct answer to a question, should it expect him to get the question right next time? On a more practical level, simple record keeping of what has been discussed can be difficult when the dialogue jumps about or does not treat each subproblem in exhaustive detail.

Before we can consider the maintenance of the student model, its growth and revision, we must consider just what the model should contain: what aspects of behavior are we seeking to explain? This question is directly tied to what we will want to do with the model, and how it is represented. The final sections of this chapter review uses of the student model that are described elsewhere, and consider the adequacy of MYCIN-like rules as a representation for modeling the student's behavior.

8.1 Contents of the Model

We generally think of a student model as an evolving description of what the

student knows about the subject domain. From this perspective, we start with a corpus of facts and relations, say rules or a semantic net, and "paint in red" what we think the student knows. As described below, this dynamic process is based on various sources of evidence and ways to reason about the evidence, involving considerations like measures of a priori difficulty, structural relationships in the knowledge base, and common sense reasoning.

However, from another perspective, it is too simplistic to talk about the student model as if it is a description or list of facts contained in the student's head. We can get a better grasp on what the model should contain if we think of it in terms of what the student is doing, his cognitive state [Winograd, 1977]. A cognitive student model characterizes domain knowledge in terms of its purpose and application to problem solving. We might say that the student's attention is focused on a certain subgoal (determining the type of the infection), he is considering a particular method (analyzing CSF test findings), and that he is following a certain strategy (differential diagnosis). From the perspective of the cognitive process, we will naturally broaden our model to include process distinctions like whether the student is proceeding on the basis of what he knows, or whether he is following up on lines of reasoning suggested by the program. For example, this aspect of the model could be very useful for deciding whether to interrupt or what kind of suggestion to make.

In short, the tutor is modeling a thinking person; the cognitive model will be used to *explain what the student is trying to do.* A tutorial, like any interaction in a dialogue, can be described as a "shared process of trying to know." This requires that the participant models embrace more than a list of "known" facts. Given this criterion for

the student model, we use MYCIN's representation of knowledge to characterize the model's contents in terms of rules and strategies for doing diagnosis.

We consider a rule to be a skill or problem-solving method, and make a distinction among whether the student knows the method, is able to use it (can achieve prerequisite subgoals), and has actually applied it to a particular problem. Knowledge of skills has "local" value for explaining the student's requests for case data (a given method requires specific data to be applied), and explaining student hypotheses about subproblems (a given method contributes to the solution in a specific way). Skills are an important part of what the tutor wants to teach: does the student ask for relevant data and draw appropriate conclusions?

Strategies serve to select problem-solving methods. We make a distinction between domain-specific strategies and task-specific strategies. A domain-specific strategy orders methods on the basis of domain knowledge. A task-specific strategy is an approach for attacking the problem, e.g., hypothesis revision strategies and data collection protocols. Unlike problem-solving skills, strategies have a global effect and may be followed concurrently. Indeed, there is reason to believe that people need to follow meta-strategies for selecting strategies. However, we will not consider meta-strategies here (see Section 5.3.3 for related discussion).

An important consideration in tutoring is the use by the student of rules and strategies that do not correspond to "expert" knowledge already represented in the program. However, the technique of overlay modeling, used by GUIDON (introduced in Section 4.4.2), attempts to explain the student's behavior exclusively in terms of

MYCIN's rules and strategies (see Figure 8.1). This means that alternative, justifiable problem-solving behavior might be incorrectly diagnosed as wrong, and misconceptions might be detected, but never understood by GUIDON (so the program wouldn't be able to explain to the student what is wrong). Even when the student reaches the same conclusions as MYCIN, we have no guarantee that he is using MYCIN's rule set; there may be other reasoning paths based on a logically adequate, but different set of rules.

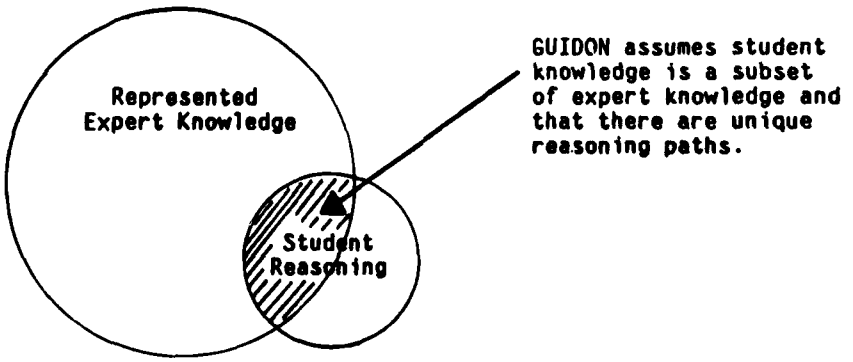


Figure 8.1. Assumptions of GUIDON's Overlay Model

Finally, note that the current version of GUIDON does not model student strategies. First, MYCIN does not have a well-developed set of problem-solving strategies (metarules), indeed there are only two metarules in the system. Protracted sessions with expert physicians may be necessary in order to formalize a useful set of such strategies. Second, considerable experimentation involving analysis of protocols will be required before we can say what the task-specific strategies are that we should be trying to detect. Once strategies have been formalized and

represented so that GUIDON can reason with them, a later version of the program could incorporate them in the student model.

In spite of these limitations, MYCIN's production rules constitute a complex and comprehensive body of knowledge, easily justifying their use as the foundation of an overlay model. Creating the model itself from consultation records is sufficiently difficult to warrant study. Pushing the program to its limits in trial dialogues can be expected to provide valuable experience for developing models of alternative or flawed reasoning methods and strategies. The final section of this chapter (Section 8.5) shows how such a study might proceed.

8.2 Sources of Evidence

Following Carr and Goldstein [Carr & Goldstein, 1977], we classify evidence for growing and revising the student model into three groups: background evidence for initializing the model, implicit evidence, and explicit evidence.

8.2.1 Background Evidence

When GUIDON wants to know if the student knows a domain rule (in order to follow some tutorial rule), it checks for evidence collected in previous tutorial sessions with this student (USE-HISTORY property of the d-rule). If the rule has never been considered before, then a t-rule packet (BACKGROUND.RULEPK040) is applied to initialize the model. The current set of t-rules illustrates typical background considerations. However, it has not been tested extensively, and may omit some useful relations. The background t-rules are:

1) The rule sophistication ranking is compared to the student's estimation of his problem-solving experience (Section 7.1.3). If the student's level is greater than the sophistication level of the rule, then this is strong evidence that the student knows the rule (for example, if the rule were stated as a quiz, he would answer the question correctly). If the levels are the same, then this is considered to be only moderate evidence. Rule sophistication levels are normative: they represent what a student at a certain level should be expected to know. These levels were determined informally by the rule authors, based on their experience and the source for the rules. For example, esoteric rules derived from medical journal articles were given the "specialist" (4 on a scale from 0 to 4) ranking.

2) A definitional rule (one that concludes a value with certainty) is considered to be easier to learn than one that is based on uncertain evidence. If the definitional rule is one that the student has probably seen before (at or below his sophistication level), then this strongly increases the tutor's belief that the student knows this rule. However, if the definitional rule references a domain list (e.g., the association between body sites and infection), then this weakens our belief that the student will know how the rule applies to a new situation, so we don't modify the student model.

3) Rules with a clause that involves iteration seem to be difficult to learn. This is considered to be strong evidence (again a priori) that the student will not be considering this rule during the tutorial (and would have trouble answering a question about it).

Background considerations that GUIDON does not explicitly take into account include relations among the rules: analogies, generalization-specialization links, and prerequisite ties. Some of these connections are no doubt implicitly recorded in the assigned rule sophistication ranking. For example, rules that are duals (Section 5.1.4) have the same sophistication level. Nevertheless, if we have evidence about one rule from tutorial sessions, this is probably a stronger basis for initializing the student model for the dual than the default ranking.

8.2.2 Implicit Evidence

Indirect clues can be used to reason about the knowledge guiding the student's behavior. In GUIDON this implicit evidence can be derived from the student's hypotheses and his choice of options. The current version of GUIDON doesn't analyze the student's choice of options, but it is clear that the data he requests, the forms of assistance he requires, and even the remark indices he refers to might be used to deduce the student's focus of attention, whether he is struggling to solve the problem, his domain problem-solving strategies, and so on. As a rudimentary form of this aspect of modeling, when the student requests help for solving the problem (HELP or HINT option), the program lowers its belief that the student knows the rule that is selected for presentation.

Undoubtedly the most important form of implicit evidence in a GUIDON tutorial is a student's hypothesis for a subproblem. The program attempts to account for a student's hypothesis in terms of d-rules. The reasoning involved in growing the model from this evidence is described in Section 8.3.2.5. An important problem involves apportioning credit or blame when multiple knowledge sources could account for the observed student behavior and information about intermediate student reasoning steps is missing.

8.2.3 Explicit Evidence

A tutor can also find out what the student knows or is thinking about by asking him directly. GUIDON constructs questions from one or more rules, and uses the

student's answer as explicit evidence about what he knows. (We do not ask what the student is doing, because we do not attempt to model this.) Appendix B details the process of constructing questions from rules, and evaluating and responding to the student's response. The paradoxical problem of revising the model after a question is asked is considered in Section 8.3.3.

8.3 Growth and Revision of the Model

Here we describe how GUIDON's overlay model is maintained during a single tutorial session. In contrast to the larger perspective on the modeling problem that we have considered so far (strategies, focus of attention, cognitive style), this section considers the problem of modeling the student's knowledge solely in terms of MYCIN's d-rules. In particular, we describe the components of the model and show how the program maintains this model by reasoning from the available background, implicit, and explicit evidence.

8.3.1 The Components of the Model

The components of the student model consist of properties of d-rules and properties of problem-solving goals. During the course of a tutorial, three properties are associated with each d-rule:

> 1) USE-HISTORY: The only component that persists from session to session, this represents the tutor's belief that the student knows the rule. That is, he is likely to get the right answer if he is quizzed about the rule. When a tutorial rule refers to "the student model," it is accessing USE-HISTORY.

2) SAPPLIED?: This is the tutor's belief that the student has applied the d-rule during this session. That is, he has been able to determine the evidence required by the rule's premise, and is aware of the conclusions made by the rule's action. He is likely to use these conclusions in hypotheses.

3) USED?: This is the tutor's belief that the student has used the rule to form the hypothesis (that he has just stated). That is, if asked to support his hypothesis, he would refer to this rule.

The following diagram shows how these components are related to each other and to the sources of evidence for modifying the model.

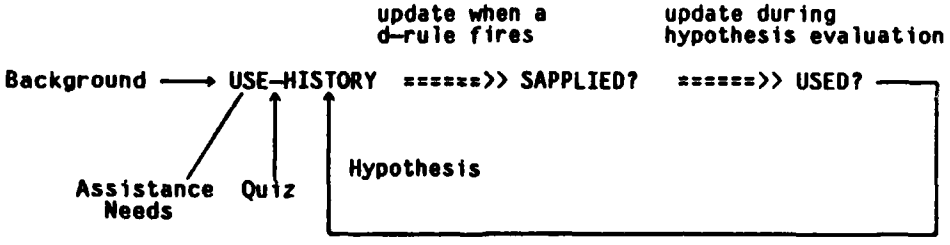


Figure 8.2. Maintenance Relations for Model Components

This diagram illustrates the distinction that GUIDON makes among having used a rule in the past (USE-HISTORY), being *able* to use it in a particular case (SAPPLIED?), and actually using it (USED?). All three components have certainty factors associated with them. Three ranges of belief are distinguished: strong, uncertain (between strong and 0) and negative (less than 0). "Strong belief" is a threshold determined by a global variable, STRONGCF, that is used throughout the program (currently .7).

On the left of this diagram we see the sources of evidence for the student model. The line from USED? to USE-HISTORY illustrates the feedback nature of the modeling process. Evidence that the student knows a rule (USE-HISTORY) contributes

to evidence that he has applied the rule (SAPPLIED?), which contributes to evidence that he actually used the rule to make an hypothesis. Finally, the hypothesis evaluation itself provides implicit evidence that the student knows the rule (feedback to USE-HISTORY).

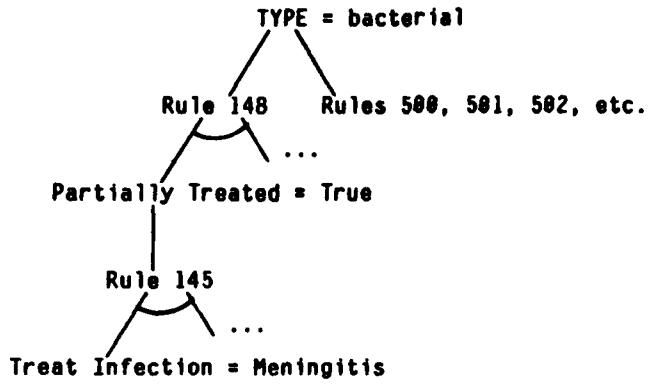
Relatively complex reasoning is required to keep the three belief components of the model up to date. Maintenance of the basic overlay model (SAPPLIED?), evaluation of an hypothesis (USED?), and revision of the cumulative model (USE-HISTORY) are described next.

8.3.2 The Overlay Model

As the student is given case data during the tutorial, d-rules that MYCIN used in the corresponding consultation are invoked in a forward direction. In this way, GUIDON knows at every moment what the expert program would conclude based on the evidence available to the student. Special t-rules (SAPPLIED.RULEPK041) determine whether the student is reaching the same conclusions. This section describes the basic operation of the overlay model, and then details a wealth of complications that are illustrated by the example protocol.

8.3.2.1 Example of Modeling Process

The following is an excerpt from the protocol (starting on page 55), showing the modeling process. The following subtree of MYCIN's solution is relevant:



The example illustrates when the components of the model are updated. Observe, in particular, the problem of accurately determining what the student knows at each step along the way.

****

>>> 16a. MYCIN has made a FINAL CONCLUSION about the infection which requires therapy.

{After determining the infection that requires therapy, MYCIN was able to apply Rule145. T-rules then show that the student does not know this rule (based on background information) and cannot apply it (it is assumed that he cannot perform the mapping operation clause 3 requires).}

[Applying tutorial rule 41.01]
BACKGROUND.RULEPK040--(RULE145 CULTURE-1)
[Applying tutorial rule 40.03]
[Backgroundcf of RULE145 = -900]
[SModel update for RULE145: NIL -> -900]
[S can't apply RULE145 CULTURE-1 due to clause 3]
[Applying tutorial rule 41.04]
[Student applied (RULE145 CULTURE-1) = -1000]

>>> 17a. MYCIN has made a FINAL CONCLUSION about whether the meningitis is partially treated.

{MYCIN applied rule148, using the information about the meningitis being partially treated. Since the student does not know this information, he cannot apply the rule.}

[Applying tutorial rule 41.01]
BACKGROUND.RULEPK040--(RULE148 CULTURE-1)
[Backgroundcf of RULE148 = 0]
[SModel update for RULE148: NIL -> 0]
[S can't apply RULE148 CULTURE-1 due to clause 2]
[Applying tutorial rule 41.04]
[Student applied (RULE148 CULTURE-1) = -1000]

>>> 18a. MYCIN just made a conclusion about the type of the infection.

.
.
.

{Later, the student states his hypothesis about the type of the infection....}

What is the type of the infection?
** BACTERIAL (9)

{T-rules compare this value to the rules applied by MYCIN. It so happens that rule148, which it is believed the student did not apply, is evidence for the hypothesis of a bacterial infection.....}

[Applying tutorial rule 14.04 (else)]
RULEUSED.RULEPK028--(RULE148 CULTURE-1)
RULE148 CULTURE-1:
R? -> (BACTERIAL 968 ((RULE148 CULTURE-1) (RULE501 CULTURE-1) (RULE500 CULTURE-1) (RULE502 CULTURE-1)) NIL)

{Bacterial is concluded by rule148, but other d-rules make the same conclusion, so this is only weak evidence (400) that the student used rule148 to form his hypothesis....}

[Applying tutorial rule 28.06]
[RULEUSEDCF of (RULE148 CULTURE-1) = 400]

.
.
.

{After all of the d-rules are considered separately, "self-referencing" strategies reconsider the evidence for the model. In particular, t-rule29.02 detects that the student model (USE-HISTORY) disagrees with evidence from the hypothesis. This is recorded, but no use is made of it in the current system.}

RULEUSEDSREF.RULEPK029--(RULE148 CULTURE-1)
[Student model: RULE148 CULTURE-1: VALUE = 0]
[Applying tutorial rule 29.02 (else)]
[Adding RULE148 CULTURE-1 to MODELSURPRISE agenda.]
[USEDSREF of (RULE148 CULTURE-1) = 400]

.
.
.

{Evidence from the hypothesis (USED?) modifies the student model (USE-HISTORY).}

[SModel update for RULE148: 0 -> 400]

.
.
.

{The evidence turns out to be decidedly mixed, so the tutor asks the student to support his hypothesis....}

What facts about this case tell you that the type of the infection is bacterial (.9)?
(Enter one factor per line.)
** CSFGLUC
** PMNS
The percent of PMN's in the CBC is not relevant for determining the type of the infection.
** CSFPOLY
** PROTEIN

{Alas, the student has indeed left out the evidence contributed by rule148, based on that missing information about partial treatment.}

You're right about the CSF protein, the percent of PMNs in the CSF and the CSF glucose value, however you should have mentioned the fact that the meningitis is partially treated (.95), the WBC from the CSF and the fact that the simultaneous blood glucose value is not known.

To summarize, we can conclude that the type of the infection is:
bacterial (considering:
whether the meningitis is partially treated [RULE148],
the WBC from the CSF and the percent of PMNs in the CSF
[RULE501],

.
.
.

{GUIDON has told the student about rule148 now, so it assumes that he has now "applied the rule."}

[Assuming that S can now apply RULE148 CULTURE-1]

.
.
.

{Later, the student asks for a summary of evidence about the type of the infection. One rule mentioned is rule148 which GUIDON thinks the student has considered.}

Summary of evidence that the type of the infection is bacterial (.98) viral (-.76) fungal (-.83) tb (-.83):

29a. The fact that the meningitis is partially treated (.95) is evidence that the type of the infection is bacterial (.19) [RULE148]

29b. The WBC from the CSF and the percent of PMNs in the CSF is evidence that the type of the infection is bacterial (.9) viral (-.6) fungal

.
.
.

{It turns out that the student was so excited about his correct evidence (CSFPOLY), he didn't read the tutor's remarks about the evidence he missed (rule148)! When the student asks for details about rule148, the program notes that two goals have not been discussed, and offers to review evidence for them. As we might expect, the student wants to know about the evidence for "partial treatment."}

** DETAILS 29A
For which of the following factors would you like to see a review of the evidence?
(select from the following):

1) the infection which requires therapy
2) the fact that the meningitis is partially treated (.95)
** 2

Summary of evidence that the meningitis is partially treated (.95):

32a. The infection which requires therapy, the fact that organisms were not seen on the stain of the pending csf culture (CULTURE-1) and the time since therapy with the cephalothin (DRUG-1) was started is evidence that the meningitis is partially treated (.95) [RULE145]

{Now that rule145 has been shown to the student, we check to see whether information about "partial treatment" changes our model of what he knows. In fact, it does not: rule148 was presented during hypothesis evaluation and was marked as applied at that time.}

[Assuming that S can now apply RULE145 CULTURE-1]
[Reconsidering S's use of PTMEN CULTURE-1]
Do you want to see RULE148?
** NO

****

8.3.2.2 Implementation

A d-rule, whether it succeeds or fails, is said to *fire* when MYCIN has completed all of the subgoals required by the left-hand side of the rule. A goal is *complete* when all of the rules that might update it have fired, and case data has been acquired, if necessary. The AND/OR solution tree is structured by inverse pointers that permit the program to repetitively reconsider d-rules (indexing them by the case data referenced in the premise part), without the high cost of reinterpreting premises from scratch. Associated with each goal pursued by MYCIN, and hence each potential topic in the tutorial dialogue, is a dynamic inference and discussion record.[1]

[1] Internally it is represented as a doubly-indexed property list. E.g, the topic

When a goal is completed, GUIDON marks it (EXPERTDONE property), and checks to see if each of the d-rules that need it can now fire (the USE-SUCC! and CAUSE-FAIL! properties of the goal). Each d-rule has a logical description of the goals that must be done before it can fire (SUCC-PARMS! or FAIL-PARMS!, depending on whether the rule succeeds or fails). For example, the (SUCC-PARMS!) description for rule578 (see Figure 5.2) applied in the context of culture-1 is:

```
(AND (TREATINF CULTURE-1)
     (OR (SPECSTAIN CULTURE-1)
         (EXAMSTAIN CULTURE-1))
     (TYPE CULTURE-1)
     (BURNED PATIENT-538))
```

This rule will fire as soon as TREATINF, TYPE, BURNED and either SPECSTAIN or EXAMSTAIN are complete.

When a successful rule fires, several operations take place:

1) ***Mark the rule*** (FIRED property)

2) ***Check the student profile*** to see if he wants to be informed of the rule's application.

3) Invoke SAPPLIED.RULEPK041 to ***decide whether the student applied the rule*** (described in Section 8.3.2.3 below).

4) For each goal that the rule concludes about, ***modify the expert's current hypothesis about this goal*** by incorporating the new evidence. If GUIDON strongly believes that the student applied the rule, then modify the student inference record in the same way.[1]

"identity of organism" has a record for each relevant context: organism-1 (perhaps from a blood culture), organism-2 (perhaps from the sputum), and so on.

[1] Thus, four values are associated with each goal of the solution tree: MYCIN's

5) Hint about or directly ***state the new evidence***, according to the student profile.

6) For each goal that has been updated (perhaps by more than one conclusion), ***check to see if the goal is now complete***.

The procedure for checking to see if a goal is complete after a rule that updates it succeeds is as follows.

1) ***Mark the list of rules that update this goal*** (UPDATE-SUCC!) by moving the rule that just fired past the DONEMARK.[1]

2) If self-referencing rules can now be applied (SREFMARK at the head of both UPDATE-SUCC! and UPDATE-FAIL!, and the student has requested the value of the parameter if it is LABDATA), then ***allow rules that have been waiting to fire to ripple their conclusions***. (Self-reference rules are caught just before they fire, and saved on the SREF-WAITING property of the goal.)

3) ***If the goal is now complete, then recur*** (check the rules that use it to see if they now fire). A goal is complete when its rules are done, and the student has already requested the value, if it is case data that needs to be asked for this case. If the rules did not contribute a final value, so MYCIN would now ask for a value, the goal is marked (NEED-TO-ASK property). Thus, t-rules that say "it is appropriate to request the reported value of the goal" will succeed if the goal is LABDATA that the MYCIN asked, but the student has not yet asked, or the goal is marked as NEED-TO-ASK (rules done, must ask now). Figure 8.3 shows protocol excerpts that illustrate this underlying process.

---

current hypothesis, the student's current hypothesis (this is GUIDON's belief and not to be confused with explicit hypotheses stated by the student), the (cumulative) evidence that has been discussed to this point in the dialogue, and the final value that MYCIN will eventually believe.

[1] UPDATE-SUCC! is of the form: <rules ... SREFMARK ... self-reference rules ... DONEMARK ... fired rules> Thus, self-referencing rules cannot fire until SREFMARK is at the head of the list. The rules for the goal are done when SREFMARK and DONEMARK are together at the head of the list.

The procedure for updating the model when a failing rule fires is much simpler. No conclusions are rippled, and the student model is not affected. However, we still check to see if the goal is complete, or self referencing rules can be applied.

```
** ALCOHOLIC
GIVENPARMDATA.PROC012--(ALCOHOLIC PATIENT-538) NIL T
[Applying tutorial rule 12.01 (else)]
[Applying tutorial rule 12.03]
NONEEDTOASK.RULEPK032--(ALCOHOLIC PATIENT-538)
[Applying tutorial rule 32.05]
PREMATURE-QUESTION:

While you will eventually find it useful to ask about whether Pt538 is an
alcoholic, it is premature to do so now.  There is still a possibility that
other data will enable you to deduce a value.

                                    .
                                    .
                                    .

** USE 42C

Pt538 is a compromised host.
[Adding ALCOHOLIC PATIENT-538 to NEED-TO-ASK agenda.]

[Adding IMMUNOSUPPRESSED PATIENT-538 to NEED-TO-ASK agenda.]
```

Figure 8.3. Protocol Excerpts Showing Effect of the Overlay Model on Data Acquisition

8.3.2.3 Deciding Whether the Student Applied a D-Rule

Special t-rules (SAPPLIED.RULEPK041) for updating the overlay model are invoked whenever the expert program successfully applies a d-rule. These t-rules determine the tutor's belief that the student has applied the d-rule. This decision is based upon:

1) The record of previous interactions with the student (USE-HISTORY) or the a priori (background) evidence.

2) If the rule was discussed during this session, then this is weak evidence that the student has taken the rule into consideration.

3) If the tutor believes that the student doesn't know how to achieve the subgoals that appear in the d-rule (factors that require the application of rules), the program concludes with certainty that the student has not applied the rule.

To determine whether the student can achieve subgoals that appear in the d-rule (criterion 3), each clause of the rule is evaluated (by interpreting the source code), using the model of the student's knowledge, rather than the conclusions used by MYCIN.[1] If the student cannot apply a clause, then this information is saved for later use in tutoring (SCANTAPPLY property of the rule).

It is important that the tutor revise the student model as the dialogue proceeds. In particular, if the program later gives information to the student that enables him to apply a rule, then the effect of this rule must be taken into account. For example, the protocol excerpt in Section 8.3.2.1 illustrated that the tutor considered the student's use of evidence for "partial treatment" after it told him about rule145. In practice, reconsideration occurs when a rule is marked as "discussed" (presented in a summary of evidence, quizzed about, worked through clause by clause in the dialogue, and so on).

The reconsideration process is undertaken when the model indicates that the rule was not applied by the student (SAPPLIED? less than STRONGCF). Each

[1] Each rule predicate eventually must invoke the MYCIN function that accesses dynamic patient data. A global flag is set that directs this function to use the overlay data base (evidence that the tutor believes is hypothesized by the student) instead of MYCIN's context tree.

conclusion made by the rule is allowed to modify the record of the student's inferences (ASSUMEDSVAL property of the goal). (The tutor prints "[Assuming that S can now apply <rule>."] Then, if the expert is done, the tutor reconsiders the rules that use these modified goals. That is, each rule (on the USE-SUCC! list of the goal) is reevaluated by the rule packet that determines if the student can apply the rule (SAPPLIED.RULEPK041). If the rule now fires (using the student inference record), the process recurs, with reconsideration of all rules that use goals that have been modified and have been completed by the expert.

In many of the tutorial dialogues we have seen so far, this reconsideration process is not very important. As shown in the typescript, the higher rules (like rule148) are usually discussed explicitly before the student requests details, and so finds out about the lower rules (like rule145). When the program reconsiders the student's use of lower goals, it finds that the higher rules are marked as discussed, and so were probably already "applied."

8.3.2.4 Complications: Interaction between Modeling and the Dialogue

It would be convenient if GUIDON could insist that the dialogue follow the AND/OR tree exactly. This would mean that a goal would not be marked as discussed until all of its rules had been discussed, and a rule would not be marked as discussed until all of its subgoals had been discussed. However, in the interests of time and economical presentation, GUIDON doesn't always wait for MYCIN to apply a rule or to finish a goal before telling the student about it and moving on to another topic.

First, it turns out that it is sometimes convenient to present a rule's conclusion without formally discussing the case data mentioned in the premise. This is illustrated below (compare to page 76).

****

** HELP

Try to determine whether Pt538 has a head injury defect.

GOAL.PROC001--(HEADINJDEF PATIENT-538) (COVERFOR CULTURE-1) T
[Applying tutorial rule 1.02]
DEFNRULE.RULEPK042--(HEADINJDEF PATIENT-538) (RULE509 PATIENT-538)

{GUIDON decides to discuss the single definitional rule instead of opening the goal for discussion. The student model indicates that the student knows this rule....}

[Applying tutorial rule 40.01]
[Applying tutorial rule 40.02]
[Backgroundcf of RULE509 = 940]
[SModel update for RULE509: NIL -> 940]
[Student model: RULE509 PATIENT-538: STRONG = T]

{Rather than discussing the rule step-by-step, the tutor presents it directly....}

[Applying tutorial rule 42.01]
PRESENTVALS.RULEPK026--NIL (HEADINJDEF PATIENT-538) (RULE509 PATIENT-538)
[Decided that RULE509 PATIENT-538 is on the lesson plan.]
[Student model: RULE509 PATIENT-538: STRONG = T]

{The student probably knows the rule, so it is stated, rather than posed as a question....}

[Applying tutorial rule 26.02]
The fact that Pt538 has not had an injury or insult to, or defect in the CNS
is evidence that Pt538 does not have a head injury defect [RULE509].

{However, the tutor must now force MYCIN to fire this rule, since the normal procedure for giving case data (injury to CNS) was not followed.}

[Forced discussed rule RULE509 PATIENT-538 to fire.]
[Applying tutorial rule 41.01 (else)]
[Applying tutorial rule 41.02]
[Student applied (RULE509 PATIENT-538) = 976]
[S applied RULE509 PATIENT-538]

****

Of course, the record of case data given to the student must still be maintained when rules are presented in this way. Similar problems occur when rules that have not fired are posed as questions (see Appendix B).

Second, a goal might be marked as discussed before all of is rules have fired. This occurs when the topic is finished prematurely ("substantially done" heuristic) or when the student requests for discussion to end (TELLME or IKNOW option). It was necessary to tell MYCIN to use evidence about the type of the infection as if it had already received its final value, just after the student used the TELLME option (page 61).

Finally, it is possible for MYCIN rules to conclude about some goal that has already been marked as completed. This redundancy is caused by rules that deliberately change the value of parameter (during a consultation) to have some side effect. Fortunately, there is only one rule of this type in MYCIN. Nevertheless, GUIDON must detect this situation to keep the overlay model consistent. This excerpt shows the error:

****

>>> 61a. MYCIN has made a FINAL CONCLUSION about the number of cultures
from this site since the onset of the infection.

```
Possible error in rules: RULE049 CULTURE-1 updates a done parm: TREATINF
CULTURE-1
Possible error in rules: RULE049 CULTURE-1 updates a done parm: WHENSYM
CULTURE-1
RULE-RETURN:

(Continuing our discussion of whether the organisms isolated from the
pending csf culture (CULTURE-1) should be considered for therapy [RULE049].)

**********************************************************************
```

It should be noted that waiting until a goal is completed by MYCIN before updating the model may be unrealistic in that the student may be proceeding on partial evidence, and even controlling his search by pushing tentative conclusions forward. For GUIDON to do the same, the modeling process would require rippling intermediate conclusions into the rules that use them, rather than waiting until a final conclusion for the goal had been reached. For example, as soon as there is evidence for a fungal infection, all rules that use this information would be examined to see if new conclusions can be drawn. There is no theoretical problem that prevents us from using MYCIN's rules for modeling in this way, but it would be a very slow process. It may be better to use this refinement of the modeling technique when less expensive assumptions fail to account for the student's behavior [Burton, 1979].

8.3.2.5 Credit Assignment: Response to a Student Hypothesis

When GUIDON attempts to account for student behavior in terms of MYCIN's rules it finds that the evidence is usually ambiguous. Even if we restrict the behavior we are modeling to that of making an hypothesis, as we have in GUIDON, the available evidence does not necessarily implicate a unique set of rules. A d-rule might make

several conclusions and the student might mention only some of them. A conclusion might be made by more than one rule. Therefore, the problem is to take the student hypothesis, the rules applied by MYCIN and their conclusions, and to construct an interpretation of which rules best account for the hypothesis. This interpretation is constructed by special t-rules. In this section, we describe these t-rules and consider related issues such as the special problem of modeling behavior that is itself based on uncertain judgments, the interaction of causal and cookbook domain knowledge, and the problem of student forgetting.

For the purpose of illustration, we will consider the following student hypothesis about the organisms causing the infection:[1]

"I think that the organisms present are DIPLOCOCCUS-PNEUMONIAE, PSEUDOMONAS-AERUGINOSA, and NEISSERIA-MENINGITIDIS."

Figure 8.4 illustrates how the overlay model is updated for this hypothesis. As mentioned above, more than one interpretation is possible. For example, three rules conclude Pseudomonas, rules 545, 557, and 578; any combination of these or none might have been considered by the student.

[1] GUIDON was interrupted during a tutorial, and its record of the session was modified to simulate specific modeling problems. Thus, the situation was contrived, but t-rules were actually applied in the way we describe here.

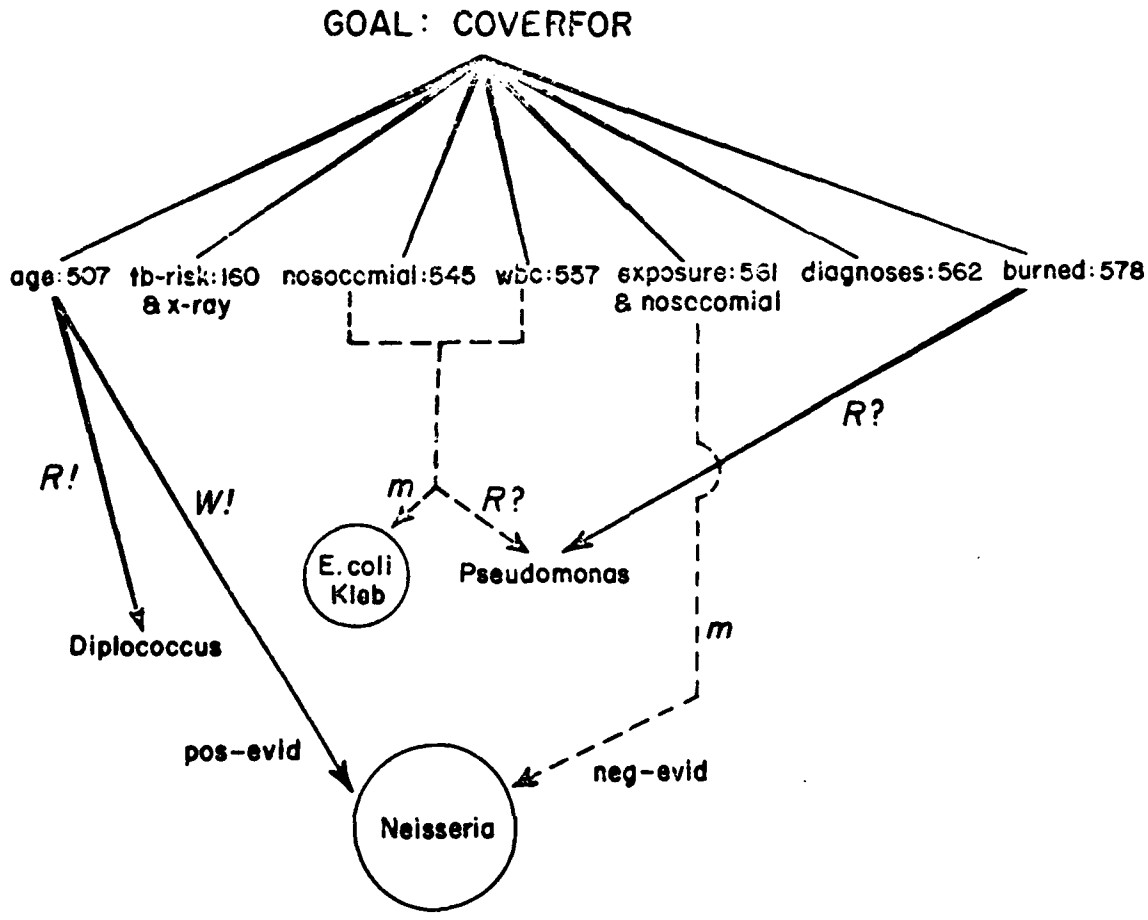


Figure 8.4. Constructing the Overlay Model from a Student Hypothesis

Key: D-rules that conclude about organisms to "cover for" are shown with their key factors. Circled values are missing from the student's hypothesis (e.g. E.coli) or wrongly stated (e.g. Neisseria). Dotted lines lead from rules the student probably did not use. $\hat{m}$ is an evidence link that the tutor decided is unknown to the student; *R* and *W*, links to right and wrong values that the tutor believes are known by the student; *!*, unique link (expert knows of no other evidence at this time); *?*, questionable (tutor is not certain which evidence was considered by the student). Thus, *R?* means that the student stated this value, it is correct, and more than one d-rule supports it.

A packet of T-rules (RULEUSED.RULEPK028) determines how strongly the tutor believes that the student has taken each d-rule that MYCIN applied into account. The t-rules that succeed will modify the cumulative belief that the given d-rule was considered by the student (the USED? property of the d-rule). In this example, we see that T-rule 28.05 succeeded when applied to d-rules 545 and 557. While the student mentioned a value that they conclude (pseudomonas) (clause 1 of the t-rule), he missed others (clause 3). Moreover, he did not mention values that can ONLY be concluded by these d-rules (clause 2), so the overall evidence that these d-rules were considered is weak (-.70).[1]

T-RULE28.05

If: 1) The hypothesis does include values that can be concluded by this d-rule, as well as other rules, and
2) The hypothesis does not include values that can only be concluded by this d-rule, and
3) There are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be -.7

After each of the d-rules applied by MYCIN is considered independently, a second pass is made to look for patterns. Two t-rules from this second rule packet (RULEUSEDSREF.RULEPK029) are shown below. T-rule 29.01 applied to d-rule 578: of the d-rules that conclude pseudomonas, this is the only one that is believed to have been considered, thus increasing our belief that d-rule 578 was used by the student. T-rule 29.05 applies to d-rules 545 and 561: the factor NOSOCOMIAL appears only in their premises, and they are not believed to have been considered. This is evidence

[1] As usual, the certainty factor of -.70 was chosen by the author. Experience with MYCIN shows that the precise value is not important, but the scale from -1 to 1 should be used consistently.

that NOSOCOMIAL was not considered by the student, increasing our belief that each of the d-rules that mention it were not considered.

T-RULE29.01

If: You believe that this rule was considered, it concludes a value present in the student's hypothesis, and no other rule that mentions this value is believed to have been considered
Then: Modify the cumulative belief that this rule was considered by .4

T-RULE29.05

If: This domain rule contains a factor that appears in several rules, none of which are believed to have been considered to make the hypothesis
Then: Substep i. Modify the cumulative belief that this rule was considered by -.3
Substep ii. Record on the agenda that this domain rule contains a factor that appears in several rules, none of which are believed to have been considered to make the hypothesis.

GUIDON records that NOSOCOMIAL is a "missed subgoal," but no t-rules have been developed for using this information in tutoring. We want to collect more data about the incidence and correctness of this part of the model before incorporating it in specialized rules for guiding the dialogue. In fact, experience with the current t-rules is too limited to report on their validity at this time, though it seems plausible that they form a basic set of considerations that should be incorporated in the model in one form or another.

It is possible to speculate on refinements to the current set of modeling rules. For example, the excerpt from the protocol (Section 8.3.2.1) illustrates that the history of the student's use of options with respect to a particular d-rule might be useful information to include in the model. If the student asks for details about

evidence for a conclusion, then we know that he has at least read the indexed remark. On the other hand, if these details were provided earlier, then we can deduce he forgot them, did not understand them, or may not have read a prolonged tutorial statement. Indeed, the excerpt demonstrates that forgetting and not reading remarks could defeat even very sophisticated modeling rules. Review, follow-up quizzes, and checking for consistency (by option use) are aspects of modeling that could be usefully incorporated in tutoring.

Besides using other sources of evidence in the model, the model might be based on d-rules that fail, or even simple variations on MYCIN's rules, should it be impossible to adequately explain an hypothesis in terms of the successful rules alone. For example, perhaps the student is not considering an *optional clause* (as NOSOCOMIAL might be considered to be optional in rule561--see Section 5.2.4.6). Similar errors include applying a rule in the wrong context (using meningitis information in a cystitis case), forgetting case data, or classifying evidence incorrectly (tabular rule).

Of course, it is not quite right to think of student errors solely in terms of the syntax of a rule. As the examples above illustrate, we don't believe that a student's knowledge is randomly different from expert rules. This is possible, but it seems more plausible that there is a reason that the student has disregarded or varied a condition. The model of what constitutes *satisfying evidence* provides one basis for generating plausible student errors. For example, the student might be disregarding the context (meningitis versus cystitis), omitting a condition that tightens evidence (NOSOCOMIAL), or confused about the principle of the inference (wrong belief, wrong conclusion in a tabular rule).

Taking our analysis of student errors a step further, accurate modeling might include interactions among the rules. Common errors that we might expect include over-generalizations and false analogies. We should be aware that attempting to explain the student's hypotheses solely in terms of MYCIN's performance rules might result in missing the reasoning steps that account for the student's behavior. In particular, the student may be forming an association that, on the surface, we do not recognize. Yet, this association could be based on the causal arguments that support MYCIN's rules. Following the terminology of [Brown, Collins, & Harris, 1977] (AILS), we cannot construct a "deep model" of the student's reasoning by using MYCIN's diagnostic rules alone. It may be necessary to consider causal and mechanistic arguments in order to explain an hypothesis. An example is presented in Section 8.5.

8.3.3 Revision of Belief

Recall that there are three components in GUIDON's student model: the cumulative belief that the student knows the rule (USE-HISTORY), the belief that he applied the rule in a given case (SAPPLIED?), and the belief that the rule was used to generate an hypothesis for a goal (USED?). We have considered background evidence for initializing USE-HISTORY, and modeling rules for determining and maintaining SAPPLIED? and USED?. It remains for us to consider how USE-HISTORY, the cumulative model, is revised. The evidence for the model includes the student's use of options (particularly, in this version of GUIDON, his request for assistance), his response to direct questions about the d-rules, and his hypotheses for subproblems.

As illustrated in the overview diagram (Figure 8.2), the revision process has the nature of a feedback operation: the model of what the student knows (and/or is doing) drives t-rules that press the student to reveal more of what he knows. The revision problem is to incorporate this new evidence into the model.

Programs like WEST and WUMPUS maintain a simple ratio of the number of times the student showed evidence of using the rule (skill or issue) divided by the number of times it was appropriate to use the rule. In keeping with our experimental use of certainty factors for representing belief, GUIDON modifies the model by combining USE-HISTORY with USED? after an hypothesis. Similarly, belief that the student knows a rule is decreased (by -.4) when GUIDON chooses to present the rule in response to a student request for assistance.[1] While the CF combining function will give a value different from taking a USED/APPROPRIATE ratio, the change in belief is in the same direction. It is doubtful that the precise value is of much significance, though the accumulation of belief is probably important, since it seems intuitively plausible that teachers consider how frequently a student *performs* appropriately.

Do certainty factors allow more evidence to be incorporated in the model than the USED/APPROPRIATE ratio? First, observe that the USED? CF could easily be converted to the ratio model (if USED? is above the "strong evidence" threshold, increment USED by 1, otherwise 0). Similarly, a wrong answer to a question or missed opportunity to apply a rule (at the time help is requested) could be modeled by

[1] GUIDON chooses the rule according to its strategy of hypothesis revision. The student's request for help is taken to mean that he doesn't know what rule to consider next, hence he doesn't know this rule. Of course, he might know the rule and just doesn't want to consider it because he is following a different strategy. Or maybe he asked for help just to see what the program would suggest.

incrementing APPROPRIATE, but not USED. However, the USED/APPROPRIATE ratio changes the model in the same way each time. Certainty factors can be used more flexibly, allowing finer distinctions to be made. For example, the tutor might want to modify its belief more when a direct question is missed than when an hypothesis seems to indicate that a rule wasn't used. Indeed, perhaps the model should be segmented according to the source of the evidence, and reasoning about the student's knowledge should be done more symbolically, based on the source. However, we don't know at this time what kinds of distinctions might be worth retaining, or how they could be used.

There is actually another basic problem involved in revising the student model. Revisions based on the student's use of knowledge, as abstracted from his behavior, pose no direct problem because the model we have described is, by design, a predictive indication of the student's use of knowledge. We at least know which way to modify the model when new evidence comes in: if his behavior is explained by the rule, then we increase our belief that he uses the rule appropriately, and vice versa. However, a quiz does not measure the student's use of knowledge, so much as his ability to recall a fact or relation upon demand. The distinction is particularly clear if you consider what happens when a student answers a question incorrectly and the *right rule is presented to him. Should this increase or decrease* the tutor's belief that the student will use the rule appropriately in the future?

If the student answers a question correctly, GUIDON increases USE-HISTORY by .7 for fill-in and multiple-choice questions, and .2 for true/false questions. If the student is wrong, USE-HISTORY is decreased by .2. If our model of problem-solving

insisted on a clear distinction between *recall* and *use*, we would have to keep a separate tally for quizzes. So we might say, "here is GUIDON's belief that the student will answer a question about this rule correctly," and "here is GUIDON's belief that the student will consider this rule (on his own) when presented with a case in which the rule is appropriate." However, the ability to recall a rule seems to be evidence, however weak, that the rule will be used appropriately as well. This is why we feel justified in combining the measures in this first version of the program.

Regardless of whether there are two measures or one, it is not clear how a corrected wrong answer will affect future student performance. One could argue that once the student is corrected, he will learn the rule and use it correctly next time. However, GUIDON models conservatively by assuming ignorance. Revising the belief model intelligently might require consideration of individual differences in learning and comprehension, a model of forgetting, and certainly some measure of how hard it is to incorporate the particular rule in behavior (on long and short term bases).

8.4 Uses of the Model

Many of the t-rules presented in Chapter 6 make use of the student model. These uses are summarized below.

1) ***Economical discussion.*** The model allows the tutor to match time and detail of a discussion to the needs of a student. This is particularly important to avoid boring the student by questions and protracted consideration of material that he already knows.

2) ***Responding to an hypothesis.*** Given the classification of d-rules into used, possibly used, and missed, the tutor

congratulates the student and tests him. The model enables the tutor to selectively match its response to the needs of the student, exhibiting both economical and opportunistic strategies.

3) ***Opportunistically examining understanding.*** The feedback nature of tutoring is illustrated well by opportunistic tutoring strategies. One reason GUIDON interrupts the student to present new facts or quiz him is to improve the model of what he knows. For example, the program quizzes about a rule that uniquely accounts for a student's request for data (when the student model indicated that the student did not know the rule). From this perspective, one goal of the tutor is to maintain an accurate model of what the student knows and is doing. Thus, the model drives certain tutorial actions for the main purpose of maintaining its own consistency. Indeed, the perspective that tutoring is a model-building activity suggested GUIDON's current set of opportunistic strategies for finding and resolving conflicts in the student model (see particularly Section 6.2.3.3).

4) ***Providing assistance.*** As presented in the introduction to this chapter, it is useful for a student model to indicate what the student is doing or thinking about, as well as what he knows (in a static sense). This component of the model is particularly valuable when providing assistance to complete a subproblem. We would expect a responsive tutor to pick up on the student's partial solution, rather than ignore it or suggest a different route without comment. (GUIDON's model does not currently incorporate the student's strategies.)

It should be noted that in no case does the model alone make the tutor efficient and responsive. On the other hand, a battery of clever instructional methods are not sufficient either for guiding the dialogue intelligently. Both must be coupled so that tutoring specifically exercises the skills that need work (BIP research [Wescourt, Beard, & Gould, 1977] [Wescourt, et al., 1977] emphasized this).

8.5 Inadequacy of Performance Knowledge for Modeling Misconceptions

In the description of MYCIN's knowledge base (Chapter 5), we emphasized that the rules only capture a superficial layer of associations between evidence and diagnosis. We gave examples of mechanistic reasoning and principles that implicitly justify the rules. That is, arguments for the validity of the rules refer to concepts and relations that are not represented in the knowledge base. Therefore, it seems plausible that when a student (or physician for that matter) is asked to justify an hypothesis, he may offer evidence that he feels is in turn justified by underlying causal relations and principles that he has learned. When the student's hypotheses and support are wrong because his underlying reasoning is faulty, GUIDON will be able to detect the superficial problem (the wrong answer), but not be able to offer any help at all in resolving the underlying confusion. In this respect, GUIDON tutoring capabilities are bounded by the rule set, which limits modeling to a relatively shallow level of diagnostic reasoning.

Figure 8.5 is an excerpt from a GUIDON tutorial that illustrates the limits of the program's modeling capability. When asked to support his (incorrect) hypothesis that the patient had had a splenectomy, the student offers evidence that MYCIN indicates is irrelevant. Our question should be, why did the student think that whether or not the patient was a compromised host could tell him whether the patient had had a splenectomy? We will examine the student's reasoning [1] and consider what additional knowledge, if any, GUIDON would need to be able to explain to the student where he went wrong.

---

[1] In this example, the author, who has no medical training, was the student. The confusion was not contrived.

```
.
.
.

** IKNOW SPLENECTOMY

Has Pt538 had a splenectomy?
** NO

What facts about this case tell you that Pt538 has not had a splenectomy?
(Enter one factor per line.)
** COMPROMISED
Whether Pt538 is a compromised host is not relevant for determining
whether Pt538 has had a splenectomy.

(Enter another factor if you wish.)
** DONTKNOW

It remains to collect further evidence for determining whether Pt538
has had a splenectomy.

.
.
.
```

Figure 8.5. A Student's Support for an Incorrect Hypothesis

It turns out that the student based his hypothesis on a false analogy that he derived from a relation he had previously learned, but misunderstood. The student "knew" that a compromised host was a patient whose immunological system was impaired. For example, a drug that the patient was receiving (alcohol, steroids) was affecting his ability to fight an infection. The student also knew that a splenectomy was an operation in which the spleen is removed. At this point, interesting connections with other facts could be derived (see Figure 8.6).

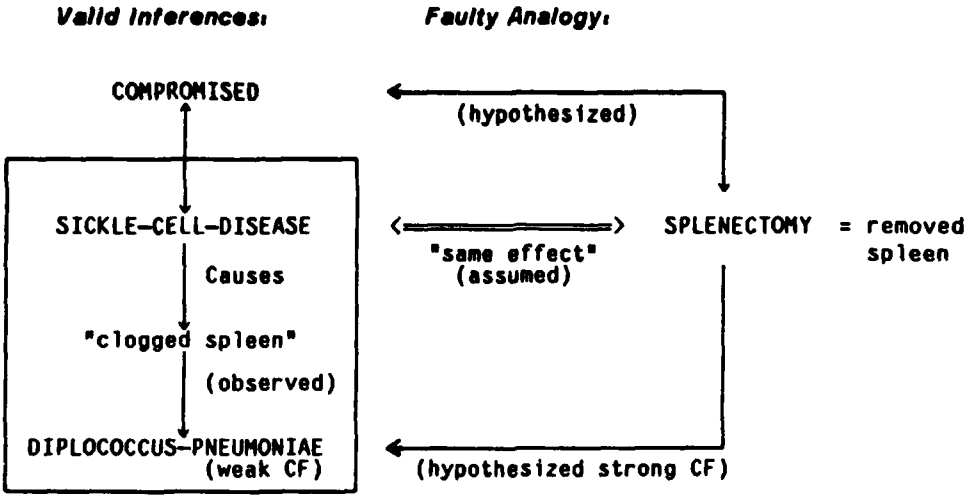


Figure 8.6. Student's Justification for an Hypothesis

The student knew that the effect of sickle cell disease was to impair the spleen, leading to infection caused by diplococcus (see Section 5.2.4.5 for discussion). Reasoning by analogy, he assumed that however impaired spleen function affected the body, the same effect was to be expected if the spleen were actually removed. He concluded that a patient with no spleen at all had an even greater chance of infection by diplococcus than a patient whose spleen was merely impaired. This was the student's first mistake, as we will see in a moment. MYCIN does have a rule that says "if splenectomy then prescribe therapy for diplococcus," but the certainty is even weaker than in the case of sickle cell disease. If the student knew why, he probably would not have gone on.

Next, having considered one way in which sickle cell disease and a splenectomy

were similar, the student recalled a rule that concluded that sickle cell disease was not present. This would be useful because he wanted to give GUIDON evidence that the patient had not had a splenectomy. The rule stated, "if the patient is not a compromised host, then he does not have sickle cell disease" (and similarly, if he has sickle cell disease then he is compromised). From here it was a simple leap by analogy to assume that a patient with a splenectomy would be a compromised host. And evidence that the patient was not a compromised host, would be evidence that he had not had a splenectomy. This is what the student told GUIDON, but it is wrong.

The student's analogies are faulty because they are based on an erroneous justification for the compromised/sickle-cell relation. To see why, consider another (correct) rule that concludes that sickle cell disease is not present: if the patient is not Black, then he does not have sickle cell disease. It would be ludicrous to assume that the patient had not had a splenectomy because he is not Black. When we say that an analogous relation between rules holds, we mean that underlying justification for one rule carries over to the situation of another rule. Since the Black/Sickle-cell rule is based on a genetic argument and splenectomy is an operation that can happen to anyone, there is no analogy to be drawn.

However, the student thought that his justification for the compromised/sickle-cell rule would hold in the case of a splenectomy as well. In particular, he thought that saying a sickle-cell patient was compromised was merely restating the fact that he was likely to come down with a diplococcus-caused infection. Indeed, an impaired spleen is a form of impairment of the immunological system. This reasoning successfully carries over to the case of a splenectomy because a removed spleen is

assumed to simulate the conditions of an impaired spleen. In fact, this analogy is perfectly correct, it is just based on the wrong justification for the compromised/sickle-cell rule.

To make a correct analogy with the compromised/sickle-cell rule, the student has to know the basis for that rule. Saying that the rule merely "restates" the fact that diplococcus infection is likely is not correct. The student did not realize that sickle-cell disease does more to the body than impair the spleen. Indeed, there are several effects that as whole severely impair the patient's health. These effects are not localized to the spleen, as in the case of a splenectomy. Therefore, a splenectomy does not fulfill all of the conditions that justify the compromised/sickle-cell rule, and no analogy can be drawn.

It is interesting to observe that GUIDON could theoretically find the relationship in MYCIN's rules that ties together compromised host, sickle-cell disease, diplococcus, and splenectomy. However, the best it could do would be to hypothesize a false analogy between sickle-cell disease and a splenectomy. The program knows nothing about the spleen, so it cannot begin to diagnose the cause of the false analogy (over-simplified understanding of sickle-cell disease).

We do not know if the confusion illustrated here is typical. It seems plausible that this is not an unusually complex reasoning error. And yet, it was surprisingly difficult to track down the misconception and state its origin succinctly. Indeed, the student's mention of compromised host as a factor to support his hypothesis was originally only a hunch, based on a vague recollection that there was some underlying

connection between a splenectomy and likelihood of infection. From here we can only speculate about revisions to GUIDON that will enable it to detect a bug like this and articulate the confusion for the student. It seems likely that a program with this capability will need to question the student about his reasoning, probably attempting to fit his rationale into a model of how bugs are formed, as our example illustrates a false analogy based on insufficient conditions for the basis relation. Stevens and Collins have made a beginning in this kind of analysis [Stevens, Collins, & Goldin, 1978]. In any case, to discuss underlying errors with the student, to say more than "compromised is irrelevant," GUIDON will need a grasp on the principles that support MYCIN's rules. The representation and acquisition of this knowledge are open research problems.

In more general terms, we should question just what modeling performance can be expected of a program that does not have voice and visual clues for following a student's thinking. How well do people do at this task? What resources do they use? What triggers revision of their model of the student? What kinds of assumptions do they make and how do they test them? Building ICAI programs like GUIDON may involve having a better understanding of expert human cognition--a research perspective we will return to in the final chapter.

Chapter 9

Results: Experiments with Other Knowledge Bases

CONTENTS

9.1 PUFF Tutorial

9.2 Discussion

9.3 SACON Tutorial

9.4 Discussion

GUIDON's design permits us to substitute a MYCIN-like knowledge base different from the infectious disease rule set used to develop the initial tutoring capabilities. This experiment was performed using the PUFF [Kunz, et al., 1978] consultation system for diagnosis of pulmonary function and the SACON [Bennett, et al., 1978] system for suggesting a structural analysis regime. By using GUIDON's t-rules for managing dialogues involving other rule sets, we hoped to demonstrate that the t-rules were in fact independent of the infectious disease rule set, and to shed light on the generality of the current set of t-rules. Aside from revealing a few simple oversights in the code, these experiments revealed that the t-rules are general enough, but GUIDON needs more strategies for economical discussion of complex cases.

Coupling GUIDON to another rule set is straightforward. About an hour of programming time is required to set up the proper files and pointers in the system. GUIDON needs to be told three facts about the consultation system: its name (PUFF, SACON); the name given to the object that a case is about (patient, structure); and the goal of the consultation (PULMONARY-DIAGNOSIS, ANALYSIS-REGIME). Furthermore, the names of the rule-model file, dictionary, and parameter and rule annotations files must be specified. GUIDON is then constructed in batch mode, using a modified (small) INTERLISP core image, the EMYCIN (*Essential MYCIN*) package (rule interpreter, user interaction package, utility functions), tutorial modules, and the expert knowledge base (parameters, rules, and domain variables). As usual, discussion of a case must be preceded by a consultation and analysis that produces the AND/OR solution tree. This is also done in batch mode.

It should be remembered that no additions were made to these knowledge bases for the purpose of tutoring. In particular, the student model was effectively disabled because all rules were assumed to be of expert level sophistication Section 8.2.1. Since the student was a complete novice, and there was no history of previous interactions with him, presentation methods were always selected that assumed that the student was ignorant of the domain rules. This tended to make the tutorial considerably more verbose since step-by-step discussion of rules was used rather than a direct quiz. On the other hand, the opportunistic interruption strategy requires that a rule used for a quiz be within two levels of the student's sophistication; the result was that no interruptions were ever made.

Other knowledge base additions that we know would be useful are parameter definitions and rule justifications. However, unlike MYCIN, rule schemas are not necessary in either PUFF or SACON for simplifying summaries of evidence. Rule models were available and appear in the tutorials.

9.1 The PUFF Tutorial

The PUFF knowledge base is much simpler than MYCIN's. There are only 60 rules and only one "object" in a case, the patient. The purpose of a PUFF consultation is to diagnose lung disease, e.g., bronchitis, asthma.

The typescript has been considerably shortened, but otherwise appears here in its original form. Omitted lines are indicated by annotations.

March 30, 1979 10:36PM in <CLANCEY>GUIDON.SAV;90330 by CLANCEY

This is the tutor for PUFF.

{...22...LINES...OMITTED...}

Tutorial Session 30-Mar-79 22:36:48

Which case?
** 7

{...8...LINES...OMITTED...}

[consultation of 23-Feb-77 11:55AM]

Initial Case Data

PATIENT-7:
The patient number of PATIENT-7 is 7446.
Asthma is one of referral diagnoses of the patient.

{...50...LINES...OMITTED... After general orientation, the program delves into the goal rule....}

(Now considering RULE001.)

We must now determine whether an attempt has been made to deduce whether there is an interpretation of potential obstructive airways disease.

Let's consider a way to determine this by definition....

(Now considering RULE002.)

We must now determine whether an attempt has been made to deduce the severity of obstructive airways disease of PATIENT-7.

{...4...PAGES...OMITTED... A major topic is introduced....}

We must now determine whether an attempt has been made to deduce the subtype of obstructive airways disease.

You should now proceed to ask questions that will enable you to make an hypothesis about the subtype of obstructive airways disease.

PATIENT-7 is a person. In this context, when we are considering the subtype of obstructive airways disease, we generally find it useful to consider:

16a. the severity of obstructive airways disease of the patient
16b. the subtype of obstructive airways disease
16c. The degree of diffusion defect of the patient

There are 11 rules used by PUFF to conclude about this topic. Altogether there are 10 factors considered in these rules.

The values for the subtype of obstructive airways disease are: asthma, bronchitis, emphysema and mixed.

** PENDING

We need to discuss the following subgoals before our discussion of the subtype of obstructive airways disease will be complete:

17a. the degree of lung restriction of PATIENT-7
17b. The degree of diffusion defect of PATIENT-7
17c. The degree of reversibility of airway obstruction of PATIENT-7
17d. whether there are post bronchodilation test results

We need to discuss the following case data that is directly relevant to the subtype of obstructive airways disease:

18a. the severity of coughing of PATIENT-7
18b. THE DLCO-Obs/DLCO-pred ratio of PATIENT-7
18c. the TLC(DLCO)observed/predicted of PATIENT-7
18d. the TLC(body box)observed/predicted of PATIENT-7

{The usual options are available.}

** USE 18A

The severity of coughing of PATIENT-7 is none.

If the severity of coughing of PATIENT-7 were greater than or equal to

mild, one could conclude that the subtype of obstructive airways disease is bronchitis (.3) (other information would also need to be available to make this conclusion) [RULE047].

** DISCUSS 17C

You should now proceed to ask questions that will enable you to make an hypothesis about The degree of reversibility of airway obstruction of PATIENT-7.

** FACTORS

The following factors will be useful to determine The degree of reversibility of airway obstruction of PATIENT-7:

19a. whether there are post bronchodilation test results
19b. The change in FEV1 due to dilation of PATIENT-7
19c. The change in Resistance due to dilation of PATIENT-7

{In a strange domain, the student might not even know what parameters need to be deduced and which are available simply by asking.}

** 19A
It is not appropriate for the consultant to ask about whether there are post bronchodilation test results.
You'll have to work this out for yourself. Try the DISCUSS option for further assistance on this topic.

** 19B

The change in FEV1 due to dilation of PATIENT-7 is 31.

** 19C

The change in Resistance due to dilation of PATIENT-7 is 20.

>>> 20a. PUFF has made a FINAL CONCLUSION about whether there are post bronchodilation test results.

>>> 21a. PUFF just made a conclusion about The degree of reversibility of airway obstruction of PATIENT-7.

>>> 22a. PUFF just made a conclusion about the findings about the diagnosis of obstructive airways disease.

>>> 23a. PUFF just made a conclusion about the findings about the diagnosis

of obstructive airways disease.

We have substantially completed our discussion of The degree of reversibility of airway obstruction of PATIENT-7.
Would you like to make a hypothesis now (as opposed to gathering more information)?
** NO

{Opportunistic tutoring: see Chapter 1 for discussion of this part of the dialogue.}

We will now consider evidence that leads to a conclusion about The degree of reversibility of airway obstruction of PATIENT-7.
Here are some conclusions that you may not have considered....

The change in FEV1 due to dilation of PATIENT-7 and The change in Resistance due to dilation of PATIENT-7 is evidence that there are post bronchodilation test results [RULE028].

Given the fact that there are post bronchodilation test results and The change in FEV1 due to dilation of PATIENT-7, does this tell you that The degree of reversibility of airway obstruction of PATIENT-7 is good (.5)?
** DONTKNOW
Given these facts, we can conclude that The degree of reversibility of airway obstruction of PATIENT-7 is significant (.8) [RULE020].
The conclusions in the question hold when The change in FEV1 due to dilation of the patient is between 5 and 10.

** TOPIC
We are discussing The degree of reversibility of airway obstruction of PATIENT-7.
Hypotheses you should be considering are: none, insignificant, slight, good and significant.

{GUIDON's phrasing is awkward for values like "none" and "significant," but the reasoning is valid.}

** HYPOTHESIS

What is The degree of reversibility of airway obstruction of PATIENT-7?

** SIGNIFICANT

You're right, there's good reason to suspect significant.

The fact that there are post bronchodilation test results and The change in FEV1 due to dilation of PATIENT-7 is evidence that The degree of reversibility of airway obstruction of PATIENT-7 is significant (.8) [RULE020].

{...3...PAGES...OMITTED}

* * * * * * * * *

Back to our discussion of the findings about the diagnosis of obstructive airways disease....

** USE 35D

The number of pack-years smoked is 17.

>>> 37a. PUFF just made a conclusion about the findings about the diagnosis of obstructive airways disease.

Using other information available to you now, the number of pack-years smoked enables you to conclude that Airway obstruction is consistent with the patient's smoking history. is one of the findings about the diagnosis of obstructive airways disease [RULE036].

** JUSTIFY RULE036
RULEINFOFILE does not exist

{*Again, the hint makes sense, but it could be worded better.*}

** HINT
Hint: Look for strong, positive evidence that Elevated lung volumes indicate overinflation. is one of the findings about the diagnosis of obstructive airways disease.

{...49...LINES...OMITTED...}

We have substantially completed our discussion of the findings about the diagnosis of obstructive airways disease.
Would you like to make a hypothesis now (as opposed to gathering more

information)?
** YES

{The student must enter internal identifiers for the text-valued parameter.}

What are the findings about the diagnosis of obstructive airways disease?
** $OVERINFL (9) $SMOKE (5)

You're right, there's strong evidence for $overinfl and reason to suspect $smoke.

We can conclude that the findings about the diagnosis of obstructive airways disease are:
$overinfl (considering the RV/RV-predicted ratio of PATIENT-7 and the TLC/TLC-predicted ratio of PATIENT-7 [RULE005]),
$smoke (considering the severity of obstructive airways disease of PATIENT-7 and the number of pack-years smoked [RULE036]).

What facts about this case tell you that the findings about the diagnosis of obstructive airways disease are: $asthma2 (1.0) $asthma1 (1.0) $rev4 (1.0) $dilation (1.0) $mmf (1.0) $fev1/fvc1 (1.0) $f5025 (1.0)?
(Enter one factor per line.)
** HELP

{...2...PAGES...OMITTED... Discussing "summary statements" poses problems because it, like "findings," is really a cluster of simple true/false parameters that would have distinct labels in the infectious disease system.}

We must now determine whether an attempt has been made to deduce the summary statements about this interpretation.

Try to make a conclusion about the summary statements about this interpretation.
What are the summary statements about this interpretation?
** HELP

What facts about this case tell you that Obstructive Airways Disease of the asthmatic type is one of the summary statements about this interpretation (.51)?
(Enter one factor per line.)
** 350
Referral diagnoses of PATIENT-7 can be used to determine the subtype of obstructive airways disease...
and this enables us to determine the summary statements about this interpretation.

** 35G
The degree of reversibility of airway obstruction of PATIENT-7 can be used to determine the subtype of obstructive airways disease...
and this enables us to determine the summary statements about this interpretation.
**

You're right about the subtype of obstructive airways disease, however you should have mentioned the severity of obstructive airways disease of PATIENT-7.

(Continuing our discussion of the interpretation [RULE001].)

We have already determined that an attempt has been made to deduce whether the test was done properly.

We have already determined that an attempt has been made to deduce whether there are inconsistencies in the data.
At this point, PUFF would print the interpretation of the pulmonary function tests.
(End of our discussion of RULE001)

This completes our discussion of PATIENT-7.
Saving student model...done.

Do you have any comments?
** NO

Thank you.

9.2 Discussion of the PUFF Tutorial

The PUFF tutorial demonstrates that GUIDON's t-rules are in fact independent of the rule set of the expert system. The author played the role of student; his background was insufficient for solving the problem without assistance. The subgoals list was useful for getting some grasp of the structure of the rule set, though an explanation of the flow volume loop model of pulmonary function would have been more

useful. It was hard to remember the meaning of abbreviations like FEV1/FVC. PUFF does not have either a definition file or justifications for its rules.

A PUFF *finding* is a text string that is printed after the consultation, constituting the interpretation of the case, e.g., "Good response to bronchodilators is consistent with asthmatic condition, and their continued use is indicated." Unfortunately, GUIDON does not give a text-string-valued parameter the special treatment it requires, so "is one of the findings about the diagnosis of obstructive airways disease" is tacked on to statements unnecessarily. More seriously, the hypothesis evaluation and question-asking routines expect the student to type in the internal identifier for findings ($smoke, $overinfl). Clearly, some form of parser is required here.

The careful reader will observe that the program says, "PATIENT-1 is a person" (just as in SACON it will say, "STRUCTURE-3934 is a structure"). When an object in the context tree need not be qualified (as a culture can be pending, positive, or negative), then this description can be left off.

The most serious theoretical problem that we observe is the tutorial's verbosity. When most of the parameters are determined by single rules, as in PUFF, the tutor methodically talks the student through the case, goal by goal, providing infrequent opportunity for student initiative. (Though, in part this is exacerbated in the PUFF and SACON tutorials by assuming that the student knows nothing.) GUIDON appears to do a decent job of generating coherent text by traversing the AND/OR tree, but has no awareness that dozens of "economical" presentations are adding up to a two page monologue. We saw this problem at the end of of the MYCIN protocol, and it recurs in more extreme form in SACON.

9.3 The SACON Tutorial

SACON's rule set is comparable to MYCIN's in size, and this case (for the "College Union Building") is more than 3 times the size of any tutorial previously run on GUIDON (in terms of the amount of case data and number of rule invocations).

April 7, 1979 7:49PM in <CLANCEY>GUIDON.SAV;90407 by CLANCEY

This is the tutor for SACON.

Tutorial Session 7-Apr-79 19:49:40

Which case?
** 3934

{...6...LINES...OMITTED... Observe the large amount of initial information describing the structure to be analyzed....}

[consultation of 7-Jul-78 4:45PM]
Initial Case Data

STRUCTURE-3934:
The name of STRUCTURE-3934 is college union building.
The analysis error (in percent) that is tolerable is 10.
Both is one of the integrity evaluation goals of the analysis.
The college union building does not have time dependent terms in its equations of equilibrium.
The college union building does not have temperature dependent terms in its equations of equilibrium.

There were 2 sub-structures, as follows...
SUB-STRUCTURE-1:
The name of SUB-STRUCTURE-1 is floor section.
The material composing the floor section (SUB-STRUCTURE-1) is concrete.
The length of the floor section (SUB-STRUCTURE-1) is 384 inches.
The weight of the floor section (SUB-STRUCTURE-1) is 108800 pounds.
The floor section (SUB-STRUCTURE-1) is solid.
The configuration of support for the floor section (SUB-STRUCTURE-1) is four-sides.

There was 1 loadings of the sub-structure, as

follows...
LOADING-1:
The name of LOADING-1 is total loading.
The number of cycles LOADING-1 is to be applied
is 1000.

There was 1 loading components of the loading,
as follows...
LOAD-COMPONENT-1:
The distribution of LOAD-
COMPONENT-1 is distributed.
The surface to which LOAD-
COMPONENT-1 acts normal is
width-length.

{...58...LINES...OMITTED...}

[Now giving initial case data to SACON... this may take a few minutes.]

>>> 2a. SACON just made a conclusion about the deflection phenomena in the north-south columns (SUB-STRUCTURE-2).

>>> 3a. SACON just made a conclusion about the deflection phenomena in the floor section (SUB-STRUCTURE-1).

>>> 4a. SACON has made a FINAL CONCLUSION about the stress criterion of the floor section (SUB-STRUCTURE-1).

>>> 5a. SACON has made a FINAL CONCLUSION about Young's modulus of the material.

>>> 6a. SACON has made a FINAL CONCLUSION about the density of the material.

>>> 7a. SACON has made a FINAL CONCLUSION about the stress criterion of the north-south columns (SUB-STRUCTURE-2).

>>> 8a. SACON has made a FINAL CONCLUSION about Young's modulus of the material.

>>> 9a. SACON has made a FINAL CONCLUSION about the density of the material.

[SACON is now considering additional questions to ask....]

We will now begin to discuss this case, limiting our discussion to one topic at a time.

In order to determine the analysis strategy of the college union building we will want to consider:
1) the analysis class of the college union building, and
2) the analysis recommendations to be considered when preparing the college union building for modelling

{...64...LINES...OMITTED... The goal rule has been started; orientation is given.}

(Now considering RULE001.)

We must now determine whether the analysis class of the college union building is known.

You should now proceed to ask questions that will enable you to make an hypothesis about the analysis class of the college union building.

STRUCTURE-3934 is a structure. In this context, when we are considering the analysis class of the college union building, we generally find it useful to consider:

11a. the types of nonlinearity in the structure
11b. the deflection phenomena in the structure
11c. the stress behaviour phenomena in the structure

There are 36 rules used by SACON to conclude about this topic. Altogether there are 3 factors considered in these rules.

Some sample values for the analysis class of the structure are: nonlinear-geometry-strength, inelastic-stiffness-degradation, inelastic-stress-failure and nonlinear-stress-margin, etc.

** FACTORS

The following factors will be useful to determine the analysis class of the college union building:

12a. the types of nonlinearity in the college union building
12b. whether the support conditions of the college union building are nonlinear
12c. the non-dimensional deflection of the floor section (SUB-STRUCTURE-1)

** DISCUSS 12A
There are no rules which successfully apply to this case that can tell us about the types of nonlinearity in the college union building.

>>> 13a. SACON has made a FINAL CONCLUSION about the types of nonlinearity in the college union building.
Back to our discussion of the analysis class of the college union building....

{GUIDON proceeds to delve into the tree, introducing three topics in succession, each a level deeper, and each concluded by a single rule that is presented clause-by-clause.}

** DISCUSS 12C
Let's consider a way to determine this by definition....

(Now considering RULE098.)

We must now determine whether you have examined the loadings of the sub-structure, selecting those for which the maximum deflection bound at a point, due to all components of this loading is known.
Let's consider a way to determine this by definition....

(Now considering RULE112.)

We must now determine whether you have examined the loading components of the loading, selecting those for which the deflection magnitude of this load component in inches is known.
Let's consider a way to determine this by definition....

(Now considering RULE158.)

We have already determined that the surface to which LOAD-COMPONENT-1 acts normal is width-length.

We have already determined that the distribution of LOAD-COMPONENT-1 is distributed.

We have already determined that the configuration of support for the floor section (SUB-STRUCTURE-1) is four-sides.

We must now determine whether the shape of the floor section

(SUB-STRUCTURE-1) is one of: the surface shapes.

{A definitional rule is presented here by giving new case data and quizzing.}

Here is some relevant data you could have asked for: the geometry of the floor section (SUB-STRUCTURE-1) is planar, the modelling dimensionality of the floor section (SUB-STRUCTURE-1) is 2, and continuum is one of the constructions of the floor section (SUB-STRUCTURE-1).

What does this tell you about the shape of the floor section (SUB-STRUCTURE-1)?
** HELP
Can you conclude that the shape of the floor section (SUB-STRUCTURE-1) is
(select from the following):
1) beam -- suggestive evidence (.62)
2) semimonocoque -- suggestive evidence (.62)
3) not shell -- strongly suggestive evidence (-.95)
4) plate -- with certainty (1.0)
** 4

Yes, that's right (see RULE100).

>>> 14a. SACON has made a FINAL CONCLUSION about the shape of the floor section (SUB-STRUCTURE-1).

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

{Clause-by-clause treatment of this lengthy rule continues....}

We must now determine whether the magnitude of LOAD-COMPONENT-1 (in psi) is known.
The magnitude of LOAD-COMPONENT-1 (in psi) is information about a case that is usually available to the consultant.

The magnitude of LOAD-COMPONENT-1 (in psi) is 1.3.

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We have already determined that the length of the floor section (SUB-STRUCTURE-1) is known.

We must now determine whether the depth of the floor section (SUB-STRUCTURE-1) is known.
The depth of the floor section (SUB-STRUCTURE-1) is information about a case that is usually available to the consultant.

{This single piece of information is the key to many rules. Observe how many conclusions are now made....}

The depth of the floor section (SUB-STRUCTURE-1) is 8.5.
>>> 15a. SACON has made a FINAL CONCLUSION about the moment of inertia of the floor section (SUB-STRUCTURE-1).

>>> 16a. SACON has made a FINAL CONCLUSION about the effective width of the floor section (SUB-STRUCTURE-1).

>>> 17a. SACON has made a FINAL CONCLUSION about gamma.

>>> 18a. SACON has made a FINAL CONCLUSION about alpha.

>>> 19a. SACON has made a FINAL CONCLUSION about the stress magnitude of LOAD-COMPONENT-1 (in psi).

>>> 20a. SACON has made a FINAL CONCLUSION about the maximum stress bound at a point, due to all components of LOADING-1 (in psi).

>>> 21a. SACON has made a FINAL CONCLUSION about the non-dimensional stress of the floor section (SUB-STRUCTURE-1).

>>> 22a. SACON has made a FINAL CONCLUSION about the stress behaviour phenomena in the floor section (SUB-STRUCTURE-1).

>>> 23a. SACON has made a FINAL CONCLUSION about the deflection magnitude of LOAD-COMPONENT-1 in inches.

>>> 24a. SACON has made a FINAL CONCLUSION about the maximum deflection bound at a point, due to all components of LOADING-1.

>>> 25a. SACON has made a FINAL CONCLUSION about the non-dimensional deflection of the floor section (SUB-STRUCTURE-1).

>>> 26a. SACON has made a FINAL CONCLUSION about the types of nonlinearity in the floor section (SUB-STRUCTURE-1).

>>> 27a. SACON has made a FINAL CONCLUSION about the deflection phenomena in the floor section (SUB-STRUCTURE-1).

>>> 28a. SACON just made a conclusion about the deflection phenomena in the

college union building.

{Still at the deepest level, GUIDON continues with rule158....}

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We must now determine whether alpha is known.
The length of the floor section (SUB-STRUCTURE-1) and the effective width of the floor section (SUB-STRUCTURE-1) is evidence that alpha is .063 [RULE107].

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We must now determine whether gamma is known.
The length of the floor section (SUB-STRUCTURE-1) and the effective width of the floor section (SUB-STRUCTURE-1) is evidence that gamma is .062 [RULE109].

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We must now determine whether the moment of inertia of the floor section (SUB-STRUCTURE-1) is known.
The shape of the floor section (SUB-STRUCTURE-1), the floor section (SUB-STRUCTURE-1), the depth of the floor section (SUB-STRUCTURE-1), the weight of the floor section (SUB-STRUCTURE-1), the density of the material and the length of the floor section (SUB-STRUCTURE-1) is evidence that the moment of inertia of the floor section (SUB-STRUCTURE-1) is 51 [RULE106].

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We must now determine whether Young's modulus of the material is known.
The material composing the floor section (SUB-STRUCTURE-1) is evidence that Young's modulus of the material is 2000000 [RULE160].

(Continuing our discussion of the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

We can now conclude that the deflection magnitude of LOAD-COMPONENT-1 in inches is .0073 inches (considering the surface to which LOAD-COMPONENT-1 acts normal, the distribution of LOAD-COMPONENT-1, the configuration of

support for the floor section (SUB-STRUCTURE-1), the shape of the floor section (SUB-STRUCTURE-1), the magnitude of LOAD-COMPONENT-1 (in psi), the length of the floor section (SUB-STRUCTURE-1), the depth of the floor section (SUB-STRUCTURE-1), alpha, gamma, the moment of inertia of the floor section (SUB-STRUCTURE-1) and Young's modulus of the material [RULE158]).

However, there are other rules that we might discuss that use information you have already been given.

(End of our discussion of a rule to determine the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE158].)

{Finally, the discussion pops up a level. Quickly, the reasoning chain unwinds to the top.}

(Continuing our discussion of the maximum deflection bound at a point, due to all components of LOADING-1 [RULE112].)

We can now conclude that the maximum deflection bound at a point, due to all components of LOADING-1 is .0073 (considering the deflection magnitude of LOAD-COMPONENT-1 in inches [RULE112]).

However, there are other rules that we might discuss that use information you have already been given.

(End of our discussion of a rule to determine the maximum deflection bound at a point, due to all components of LOADING-1 [RULE112].)

(Continuing our discussion of the non-dimensional deflection of the floor section (SUB-STRUCTURE-1) [RULE098].)

We can now conclude that the non-dimensional deflection of the floor section (SUB-STRUCTURE-1) is .0073 (considering the maximum deflection bound at a point, due to all components of LOADING-1 [RULE098]).

However, there are other rules that we might discuss that use information you have already been given.

(End of our discussion of a rule to determine the non-dimensional deflection of the floor section (SUB-STRUCTURE-1) [RULE098].)

Back to our discussion of the analysis class of the college union

building....

{We are now back to the point where the student asked to discuss the non-dimensional deflection of the floor section (12c).}

** PENDING

We need to discuss the following case data that is directly relevant to the analysis class of the college union building:

29a. whether the support conditions of the college union building are nonlinear

** 29A

The support conditions of the college union building are not nonlinear.

We have substantially completed our discussion of the analysis class of the college union building.
Would you like to make a hypothesis now (as opposed to gathering more information)?
** YES

{The student makes an hypothesis.}

What are the analysis class of the college union building?
** LINEAR-ANALYSIS

You're right, there's strong evidence for linear-analysis.

The fact that the types of nonlinearity in the college union building is not known and the fact that the support conditions of the college union building are not nonlinear is evidence that linear-analysis is one of the analysis class of the college union building [RULE003].

It remains to collect further evidence for determining the analysis class of the college union building.
However, we have substantially completed our discussion of the analysis class of the college union building.
Would you like to end discussion of this topic now (i.e., get the chance to see a summary of all evidence that can eventually be collected instead of working it out yourself)?

** YES

>>> 30a. SACON has made a FINAL CONCLUSION about the analysis class of the college union building.

Our final conclusion is that linear-analysis is one of the analysis class of the college union building.
Would you like to see a summary of the evidence that supports this conclusion?
** NO

* * * * * * * * *

Our discussion of the analysis class of the college union building is complete.

{Discussion of the goal rule continues.}

(Continuing our discussion of the analysis strategy of the college union building [RULE001].)

We must now determine whether an attempt has been made to deduce the analysis recommendations to be considered when preparing the college union building for modelling.

You should now proceed to ask questions that will enable you to make an hypothesis about the analysis recommendations to be considered when preparing the college union building for modelling.

There are 18 rules used by SACON to conclude about this topic.
Altogether there are 4 factors considered in these rules.

Some sample values for the analysis recommendations to be considered when preparing the structure for modelling are: $rec13, $rec11, $rec14, $rec19, $rec6 and $rec16, etc.

{...11...PAGES...OMITTED...}

* * * * * * * * *

Our discussion of the analysis recommendations to be considered when preparing the college union building for modelling is complete.

(Continuing our discussion of the analysis strategy of the college union building [RULE001].)
At this point, SACON would Using the information collected during the consultation, recommend an analysis method for this structure.
(End of our discussion of RULE001)

This completes our discussion of STRUCTURE-3934.
Saving student model...done.

Do you have any comments?
** NO

Thank you

9.4 Discussion of the SACON tutorial

Why is the tutorial so verbose? When GUIDON was developed using the infectious disease rule set of MYCIN, the main problem was the tediousness of requiring the student to specifically request each piece of data. Methods were developed for cutting short a discussion, acquiring data in blocks or all at once, and presenting new data to the student in combination with a quiz. In SACON, however, most of the data are given initially as part of the problem definition (this is to be expected for well-defined computational problems, in contrast to medical problems in which there are too many world facts that could possibly be relevant to list them all initially). Consequently, the program rarely gives the student an opportunity to collect more data, and instead reads out the many intermediate, definitional conclusions. The student is not asked to make an hypothesis because it is assumed that the rules are too sophisticated for him. Indeed, they were.

SACON's task is a computation-intensive problem that requires table look-up and

formula calculation (to find out parameters like "alpha," and "young's modulus"). Observe that the analysis class parameter in SACON involves 36 rules, but only 3 factors. The second major topic, analysis recommendations has the ratio of 18 rules to 4 factors. There are many more intermediate subgoals than terminal case data nodes in the tree. This explains why SACON suddenly makes 14 conclusions when the depth of the floor section is known. Clearly, this kind of problem is poorly suited to the kind of case method dialogue formalized in GUIDON's t-rules. GUIDON behaves better when there is a large number of rules, using many factors. This is why the strategy for detecting substantial completion of a goal was implemented (Section 6.2.1.2). This strategy is not needed in SACON because a topic is finished after asking just a few questions. Instead, for discussing SACON cases, GUIDON needs methods for cutting down on the number of subgoals that are mentioned, not just the number of rules.

Leaving out rules is somewhat easier than leaving out subgoals. When GUIDON omits discussion of a MYCIN rule, it is usually omitting just a single factor. If the rule fails, or contributes insignificant evidence, this omission is usually of little importance. However, to leave out a subgoal is to omit a subtree of the solution, not just one of the terminal nodes. The current set of t-rules methodically sets up each subgoal for formal discussion. The program needs methods for skipping over a topic or mentioning it in passing.

To know that a subtree can be omitted, the program must reason more deeply about a case. Current t-rules only look one level ahead and choose a strategy to discuss the next subgoal or next rule. In SACON the result is a lengthy, depth-first

tour through the rules. This shortcoming is quite similar to the disconnected dialogue shown in Section 6.2.4, in which the procedure for giving advice after an hypothesis did not look ahead to see if the subproblem was substantially complete. Managing the dialogue requires more than clever methods for discussing a single rule or a single goal. The tutor must look ahead to get a more comprehensive picture of the structure of the solution.

In fact, this experiment has revealed a basic property of rule sets that determines what kinds of dialogue management problems will arise. In MYCIN, the solution tree tends to be shallow and broad, with about as many rules as terminal case data nodes. Pruning rules from the discussion is about equivalent to leaving out a piece of data. However, in SACON the number of rules is much greater than the number of terminal case data nodes, and pruning rules must involve deliberately not discussing a subgoal. Moreover, most subgoals are determined by a single rule, so the main strategy for shortening discussion, detection of an incomplete subgoal, is not important. Instead, management of the dialogue requires omitting some topics entirely.

In conclusion, the exploratory experiment of coupling GUIDON to other knowledge bases has been very useful. In particular, it has revealed how the basic structure of a rule set determines the kinds of dialogue management problems that GUIDON must address, showing that the current set of t-rules is tuned to a certain extent to the shape of the infectious disease reasoning tree.

Chapter 10

Summary and Conclusions

CONTENTS

10.1 Accomplishments

10.2 Dimensions for Comparing and Critiquing ICAI Research
- 10.2.1 Formal Clarity
- 10.2.2 Articulateness and Understanding the Student
- 10.2.3 Mixed-Initiative Dialogue and Task Difficulty
  - 10.2.3.1 Dialogue in ICAI Programs
  - 10.2.3.2 Relation to Natural Language Research

10.3 Contribution to Medical CAI

10.4 Future Directions for the GUIDON System
- 10.4.1 Dialogue Management
  -- economy and focus
- 10.4.2 Tutoring
  -- undeveloped, untested
- 10.4.3 Modeling
  -- strategies and style
  -- critic
  -- record-keeping
- 10.4.4 Knowledge Representation
  -- non-rule-based studies
- 10.4.5 Learning
  -- feedback to t-rules

10.5 Speculation about the Future: Cognitive Science

This chapter presents a summary of the research accomplishments of this dissertation, a characterization of GUIDON's place in the ICAI field in terms of dimensions for comparing research interests, a review of future research directions for GUIDON, and speculation about the future.

10.1 Accomplishments

The contributions of this dissertation to research in ICAI can be grouped into four categories: method, discourse, knowledge representation, and pedagogy.

METHOD

I. Applied knowledge engineering to development of a tutor: Designed a framework for formalizing tutorial dialogue expertise, and developed a corpus of rules using the same approach taken for developing knowledge-based consultation systems.

II. Extended concept of a problem-solution trace to construct a complete AND/OR tree from a rule-based consultation. Used the tree for guiding tutorial dialogue, testing the student's understanding, and (overlay) modeling for hypothesis evaluation.

DISCOURSE

III. Analyzed capabilities of MYCIN's question/answer system in terms of the basic requirements of an explanation system.

IV. Applied and extended natural language ideas about discourse to tutorial dialogues; described the "dialogue management" problem in terms of these constraints:

- -- knowledge situation complexity
- -- the student model
- -- domain logic
- -- tutorial goals
- -- conversational postulates, e.g., economy

V. Emphasized importance of providing for and coping with student initiative: Formalized discourse procedures and means for sharing context that make the tutor responsive to what the student knows, wants to know more about, or wants to ignore.

KNOWLEDGE REPRESENTATION

VI. Analyzed and characterized the levels of knowledge and kinds of abstractions contained in MYCIN's rules.

A. Other forms of knowledge useful for understanding and remembering the performance rules: structure, support, and strategy

1. used rule schema for grouping rules to simplify summaries of evidence

2. showed how "models of sufficient evidence" can be used to explain the rules

3. clarified relation of problem-solving strategies to object level methods (rules), and illustrated how they can be used to provide assistance to the student

B. Corollary: Showed that the student model is superficial if it contains only performance rules (overlay model).

PEDAGOGY

VII. Developed techniques for conveying the structure of the rule set using rule models, subgoal trees, and statistics.

VIII. Implemented an hypothesis revision strategy for providing assistance to the student

IX. Provided an unobtrusive window on the expert program's reasoning by indexed remarks for which details are available: based on strategy of challenging the student to examine his understanding.

X. Constrained dialogue to goal-directed exploration, but provided the means for changing the topic arbitrarily, thus permitting flexible exploration of the expert's reasoning

XI. Defined *opportunistic tutoring* to be a tutorial interruption for the purpose of diagnostic modeling or presentation of course material, accommodated and suggested by the conventional or logical properties of the dialogue situation.

10.2 Dimensions for Comparing and Critiquing ICAI Research

What do ICAI techniques buy us? What are the dimensions by which we measure the contributions that these programs make to our understanding of effective teaching? We have chosen five dimensions for comparing and critiquing the contributions of ICAI research to a theory of instruction:

1) ***Formal Clarity***. ICAI research involves more than writing programs that can teach. Ideally, the product we aim for is a body of explicit teaching methods and strategies, with an understanding of when and why they are appropriate.

2) ***Articulateness***. The program must be capable of more than judging the correctness of the student's solution. It should be able to talk about alternative solution steps, strategies, provide metaphorical models, and draw analogies to explain its methods.

3) ***Understanding of the Student***. The program should be able to do more than determine whether a student's solution steps and

answer are consistent with an expert approach. It should be able to interpret the student's behavior and construct a description of his methods and strategies.

4) ***Mixed-Initiative Dialogue***. Dialogue flexibility should allow the program to vary the level of detail of the discussion from fine-grained causal arguments to summaries and overviews of material. The program should be sensitive to principles for controlling verbosity and persistence of focus.

5) ***Task Difficulty***. The usefulness of ICAI hinges in part on making contributions to areas that are currently difficult or time-consuming to teach by conventional methods, such as diagnostic reasoning in medicine and electronics. An important reason for developing ICAI programs is to make available individualized experiential learning, an effective but labor-intensive teaching method.

Subsequent sections briefly characterize GUIDON's contributions along these dimensions, emphasizing the relation to other work in the field.

10.2.1 Formal Clarity

In frame-oriented CAI, branching logic implicitly represented the author's teaching methods. Even when the same principles might have applied in several situations, the designer had to explicitly specify which material should be presented in each frame, and what transitions should be made based on student response. In designing a tutorial program that separates domain knowledge from teaching expertise, we are forced to make explicit the teaching strategies that had been "hard-wired" in frame systems and earlier drill and practice programs.

The advantages of this separation include: (1) multiple uses of the teaching strategies, i.e., apply them to multiple problem situations and perhaps multiple domains,

(2) the ability to directly communicate the strategies to other researchers, (3) the possibility of experimenting with methods without altering subject material, (4) modularity that enhances debugging of the program, e.g., in GUIDON it is especially easy to determine which tutoring rule is responsible for a tutorial action, and all changes are local to the discourse situation.

It was natural to represent GUIDON's teaching expertise as a separate body of knowledge because we wished to leave the pre-existing rule base of MYCIN intact, and we wanted to experiment with each of the consultation systems that used MYCIN's representation. We chose to represent GUIDON as a set of explicit rules that could be translated directly to English form because of the advantage of this discipline for organizing and clarifying knowledge. Indeed, we observe that it is common today to list knowledge for discourse and teaching in the form of production rules **[Collins, 1976] [Carbonell, 1978] [Burton, 1979]**. GUIDON is the first program that we know of that internally represents tutorial dialogue knowledge in the form of production rules.

10.2.2 Articulateness and Understanding the Student

One advantage of frame-oriented CAI is that it could guide presentation on a fine-grained level. This degree of control, which tied very specific responses to particular student behavior, was achieved by providing the program with fixed-text for many possible situations. Some of the most sophisticated systems achieved a very human-like interaction by conditionally choosing remarks on the basis of the history of the dialogue **[Feurzeig, et al., 1964]**. However, these systems were useless when the student behaved in a way that was not anticipated by the author.

It turns out that a human tutor (like the frame author) draws from many resources when detecting student errors and presenting new information. An important dimension for comparing ICAI systems is the kind of domain knowledge accessible to the system and what it allows the tutor to do. The following list is a progressive characterization of the capabilities of ICAI systems:

1) judge overall correctness of an answer (SCHOLAR, BIP)

2) judge consistency of an answer with respect to known data (SOPHIE)

3) articulate "issues" or considerations that a student (or expert) solution exhibits (WEST)

4) articulate an understanding of student behavior in terms of a chain of concepts and reasoning steps that an expert might use (WUMPUS, GUIDON)

5) articulate methods and strategies that vary from expert performance (BUGGY, WHY)

It is implicit in this list that capabilities are cumulative. For example, WHY can (or could, if programmed to do so) explain a deduction in terms of specific reasoning steps, and SOPHIE can judge the correctness of an answer, but not articulate the reasoning that makes it right in causal terms.[1] GUIDON's capabilities derive directly from the production system representation of MYCIN that makes it possible to construct a deep structure interpretation of a problem. Evidence is tied to the

[1] SOPHIE *could* give model-theoretic (logic) arguments. For example, to explain why two measurements are redundant, it could say that the second measurement does not alter the space of hypotheses that are possible. However, this kind of explanation is not very useful to a student.

solution in terms of intermediate concepts and problem-solving methods (the rules). This level of representation gives the potential for providing "best next step" assistance to the student, as well as detecting lines of reasoning he may be ignoring. GUIDON's student model makes a distinction between knowledge of a problem-solving method and its actual application for the given task. This distinction becomes important within the dynamics of the dialogue in which new knowledge about the problem's solution is gradually revealed to the student.

However, GUIDON's diagnostic capabilities are limited by its inability to construct a plausible interpretation of non-expert reasoning, as BUGGY can. And GUIDON cannot use multiple models for understanding causal rules, as has been proposed for the WHY system. It should be understood that these are not necessarily crippling weaknesses, so much as simplifications that must be kept in mind as research when evaluating the tutorial strategies and dialogue capabilities described in this dissertation.

10.2.3 Mixed-Initiative Dialogue and Task Difficulty

A good author of a frame-oriented CAI program could organize and individualize lesson material so that the course would engage the student and keep pace with his understanding. Just as subject material wasn't separated from teaching strategy, no distinction was made between principles of discourse and pedagogical methods. ICAI research has just begun to formalize the principles of a motivating, easy-to-follow style of presentation. A major contribution of GUIDON is its framework of tutoring rules which make the program sensitive to nuances of domain structure, student differences, and dialogue conventions.

10.2.3.1 Dialogue in ICAI programs

While one distinguishing feature of ICAI programs is frequently cited to be their capability to engage the student in a mixed-initiative dialogue [Hart, 1975] [Hart & Koffman, 1975], we find that the "natural language understanding" in these programs involves parsing student input [Burton, 1976] and not conversational interaction.

In the *reactive environment* of the early SOPHIE lab [Brown, Rubinstein, & Burton, 1976] the tutor never takes the initiative at all, and in the games of WEST [Burton & Brown, 1976] and WUMPUS [Goldstein, 1977], the tutor's remarks are all interruptions or reactions enabled by the immediately preceding move taken by the student.[1] SCHOLAR [Carbonell, 1970] [Collins, 1976] and WHY [Stevens & Collins, 1977] [Stevens, Collins, & Goldin, 1978], the geography and meteorology tutors, follow a Socratic dialogue format of repetitive questioning using topic selection rules and strategies for testing a student's understanding; the session itself involves no overarching problem or shared task to be discussed. ABLOCKS [Brown, et al., 1975] follows similar initiative strategies for getting the student to understand that a question is redundant, or that a deduction can be made, but again the sequence of the dialogue interaction is short and focused on one reasoning step. Thus, ICAI research has emphasized the use of domain expertise for modeling the student, and tutoring principles for correcting his misconceptions, but it has dealt only tangentially with the problems of carrying on a prolonged, purposeful tutorial dialogue.

Discourse knowledge is represented explicitly in GUIDON, using some of the

[1] Of course the evolving model of the student and history of past moves is an important and non-trivial reason for making an interruption.

same techniques for formalizing expertise that were developed in AI problem-solvers like MYCIN. Tutoring rules select *alternative dialogues* on the basis of principles of economy, domain logic, and tutoring or modeling goals. Arranged into procedures, these rules cope with various *recurrent situations* in the tutorial dialogue--e.g., examining a student's understanding after he asks a question that shows unexpected expertise, relating an inference to one just discussed, and giving advice to the student after he makes an hypothesis about a subproblem.

10.2.3.2 Relation to Natural Language Research

The discourse problems dealt with here include maintaining and sharing context as solution of the task proceeds and providing means for the student to express initiative as he unfolds the complexity of the problem and encounters limitations in his understanding. Other researchers have reported various theoretical aspects of task-oriented dialogues, such as recognizing and generating intentions and plans [Faught, 1977] [Carbonell, 1978] and focusing on objects and subtasks [Grosz, 1977]. We distinguish our work from other natural language research in two ways. First, GUIDON does not incorporate a comprehensive model of discourse like GUS [Bobrow, et al., 1977]. Second, we are formalizing tutorial knowledge in terms of specific performance rules for various situations, rather than studying human dialogues and enumerating general dialogue interaction principles.

We view the discourse problem as requiring (a) interpretation of intent and (b) reasoning about this understanding in order to act. We have minimized the understanding problem by restricting the student to a command-oriented input, thus

avoiding the problem of recognizing new dialogue situations in free text. This has allowed us to concentrate on the tutorial knowledge for generating remarks and guiding the dialogue. Consequently, we have not dealt in much generality with many theoretical issues of task-oriented dialogues, e.g., the problem of focus in the SRI assembly task [Deutsch, 1974] [Grosz, 1977]. Generation of remarks is on the level of choosing the course of the dialogue, rather than building up output from grammatical components. We devote just a small amount of computation towards determining the student's intentions, chiefly by relating student requests for data to the current topic.[1] Moreover, the AND/OR tree of topics and rules strongly constrains the tutor's choice of dialogue situations: in less rigidly pedantic dialogues, generation of conversation is sensitive to considerations such as the social context of the participants and emotional connotation [Winograd, 1977]. (Some of Burton and Brown's kibitzing strategies are like this [Burton, 1979].)

10.3 Contribution to Medical CAI

GUIDON is the first medical tutorial program that we know of that uses ICAI techniques. Relevant contributions to medical CAI include: greater individualization of tutorials, a framework for expressing and accumulating tutorial dialogue expertise, and a language for diagnostic problem-solving strategies. By constructing a model of problem-solving strategies in a student model, something traditional medical CAI systems do not even consider, ICAI systems could provide a basis for critiquing and teaching diagnosis in terms that even go beyond classroom or clinical experience.

---

[1] However, we do intend to build a more complex model of the student's strategies, and this might involve parsing complex input (as in Sleeman's program for understanding student explanations [Sleeman, 1977]).

10.4 Future Directions for the GUIDON System

Initially, we will experiment with and make straightforward additions to the existing GUIDON program. For example, 25% of MYCIN's rules have iterations in the premise, yet there are no methods for discussing these rules except clause-by-clause. It should be possible to find patterns in these rules and design concise methods for presenting them. Moreover, there are a host of subtle t-rules that were left out of the initial system because of the difficulty of implementing seemed to outweigh the advantages of their infrequent application. For example, a rule that fails to apply for the sole reason that case data are missing is interesting if the conclusion made by the rule would significantly alter the outcome of the case. Unfortunately, detection of this situation involves a fairly complex analysis of the hypothetical case, essentially requiring construction of a parallel AND/OR tree that simulates the changed conclusions.[1]

Aside from practical extensions of current techniques, there are basic theoretical problems that remain to be worked out. Many of these issues of discourse, tutoring, modeling, knowledge representation, and learning have been described at some length in the body of this dissertation. These issues, with some new twists we have not considered previously, are summarized below.

[1] The same facility could be used to respond to student questions like, "Would you still treat for E.coli if the patient were younger than 12?" This form of hypothetical inquiry was made possible in SOPHIE by simulation of the electronic circuit.

10.4.1 Dialogue Management

GUIDON's guidance of the dialogue is limited to one level of the solution: there are methods to discuss a goal and methods to discuss a rule. Analysis of the SACON dialogue shows that dialogue management should be more comprehensive. There are at least two reasons for this. First is the need for overviews: we saw that the SACON dialogue was tedious when GUIDON methodically guided the dialogue in a depth first way, only motivating the discussion one step at a time. GUIDON cannot convey a unified sense of purpose because it has no idea which topics are more important than others.

Second, the constraint on time for the session prevents treating each "context" (e.g., culture, loading, organism) with equal emphasis; some subproblems (parameters) must be skipped over or mentioned in passing. While GUIDON has a number of methods for reducing the number of rules that are mentioned and has methods for mentioning them economically, it never looks down into the solution, beyond one level of the tree, to decide what topics it should really be spending time on.

It is clear that more research remains to be done on the subject of how people focus discussion of a long, complicated problem. The program will need discourse methods for partial presentation that allow it to call subproblems to the student's attention without "discussing" them in the kind of formal procedure that GUIDON follows. This might just be practical extension along the lines of methods we have already implemented for special goals (no rules succeed, one definitional rule gives the answer). On the other hand, more domain knowledge will be required for the

program to know what is special or essential about the case being presented to the student. Given that the solution is already worked out in the AND/OR tree, a discussion might be planned ahead of time. Dialogue planning would probably relate directly to strategies for case selection, so that topics could be omitted in one case on the basis that they would be raised at another time.

A rudimentary **lesson plan** would be a list of topics and rules that GUIDON would attempt to discuss during the case session. This plan would control opportunistic tutoring strategies and methods for discussing goals and rules. Ideally, lesson planning would incorporate causal models and strategies for applying rules (doing diagnosis), so that discussion of material would be progressive and coherent. Thus, the program would quiz about a rule not just on the basis of whether the student knew it or not, or its "sophistication." Instead quizzes would be motivated by an underlying model of the domain or diagnostic strategy that the program was really trying to teach.

Case selection strategies might be based simply on statistical properties of the library of cases. A core set of cases might be chosen to cover the rule set at a certain level of sophistication. However, if we are trying to teach more than individual rules, then even if the statistical analysis captures a good set of examples, the program will not be able to articulate the principles that make the cases worth studying. For example, if we wish to convey underlying models of disease, then some topics will have *intrinsic importance*: they will be essential for interpreting the solution of the case in terms of an analogy with the model. Moreover, if we wish to teach prototypical information, like a typical case of meningitis in a child, we will find it

useful to have information about normal and expected ranges of case data. In this case, some topics will have *extrinsic importance*: they will be interesting findings in relation to the range of cases we might expect to encounter. (Information that is rarely available is probably worth mentioning when it is known, no matter how insignificant the evidence it contributes.) Given these distinctions about types of cases, case selection rules would be able used to choose "training examples" for particular students.

Finally, getting back to the more local kind of dialogue problems that have been the focus of WHY and ABLOCKS research, we note that real give and take is not possible in GUIDON dialogues. Instead of pursuing the student's reasoning, the program corrects him immediately. Using the existing framework, we should consider incorporating tutoring transitions that track down a misconception, similar to those used in WHY. Some of the problems of doing this were mentioned in Section 6.2.3.2 in the context of the hypothesis response part of the dialogue.

10.4.2 Tutoring

The current version of GUIDON is best considered as a default collection of tutoring rules that have been culled from the literature and intuition. Research will proceed by conducting informal experiments at first with students having different levels of expertise. Examination of protocols will lead to tuning and extension of the current set of 200 tutoring rules. From this experience, formal experiments for testing specific strategies might then be carried out. One experiment might try to determine whether continued development of the system should emphasize passive

devices by which the student can discover new connections at his own pace or whether the program should take a very active role of presenting new rules and strategies.

Before testing the effectiveness of GUIDON's tutoring methods further, we will want to make explicit the discourse principles that many t-rules have in common. In particular, the kind of constraints that justify transitions between discourse situations should be made explicit. For example, it appears that the dialogue excerpt of Figure 6.13 is disconnected because the tutor did not follow the principle of logically considering all aspects of the situation before offering the student possible ways of proceeding.

10.4.3 Modeling

Development of the current student model has focused on evaluating a student hypothesis for a subproblem. It would be interesting to investigate the other portion of the dialogue--gathering data. Why does the student ask a particular question? Can we silently follow his approach, understanding it to be a coherent application of domain rules? Research into diagnostic question-asking should be relevant here (e.g., [Sprosty, 1963] and [Elstein, Shulman, & Sprafka, 1978]). If we determine that the student is gathering data in blocks (e.g., laboratory data first), or that he is working on a given hypothesis, we will be in a better position to offer relevant help when he requests it. The sample protocol suggested good leads for this research, including a model of cognitive style, or approach to the problem (see Section 3.4.2).

An accurate model of why the student is asking a question would be particularly useful if we were to make the dialogue *free-form*, instead of maintaining goal-directedness. An "irrelevant question" would then be an inquiry that the program could not fit into its model of the student's strategy. Certainly the tutor could interrupt like a human tutor would when unable to follow the student's reasoning. Following the student might enable the tutor to provide much better assistance, for suggestions could reflect the approach, as well as the work that had been completed so far.

A completely different improvement to the overlay model would be to incorporate a critic that would provide feedback about the accuracy of the modeling process. The overlay model is updated on a rule-by-rule basis that leads the program to conclude that the student probably knows evidence based on a chain of reasoning steps. If the student's hypothesis does not agree with this prediction, then the tutor might reason backwards to determine the rules and subgoals that the student did not consider. The critic would then step in to reexamine the tutoring rules that were used to construct the student model, for these rules are now suspect. Perhaps parameters could be altered that would tune the modeling rules to the individual student. More speculatively, we might imagine that the program could learn better modeling rules as a result of its interactions with many students. (See [Carr & Goldstein, 1977] and [Burton, 1979] for a related discussion of adaptive modeling.)

Finally, we have not discussed the problem of keeping records of tutorial sessions. CAI provides a unique opportunity for compiling data about student behavior that can be used to simulate students and experiment with different modeling and

teaching strategies [Wescourt, et al., 1978]. The current version of the program maintains two typescript files, one that is an exact copy of what appears on the student's terminal, and another that indicates internal states of the program. The most relevant data to be collected are: 1) student use of options including requests for data and assistance; 2) modeling decisions made by the program; 3) strategical decisions made by the program (application of t-rules) and 4) discourse situations encountered during the dialogue. A practical extension to the current system would be to make the internal trace succinct and machine-readable so that it could be used in simulation studies.

10.4.4 Knowledge Representation

Chapter 5 described forms of knowledge that might be added to the program: structure, support, and strategy. In the discussion of automatic case selection, we pointed out that prototypical examples would also be useful. Considerable theoretical research remains to be done to decide how this material should be represented and acquired.

It would be interesting to study the applicability of GUIDON's discourse methods to non-rule-based representations. For example, suppose that the representation of knowledge consisted of relations that were inference triggers (e.g., as in INTERNIST [Pople, 1977]), causal links (e.g., as in CASNET [Weiss, Kulikowski, & Safir, 1977]), or even hierarchical mechanistic models [Brown, 1977a], we still might find it convenient to think of the dialogue as being structured into goals (something to find out), case data, possible outcomes (alternative hypotheses), and inference relations.

It seems probable that many of the issues of dialogue management we have discussed will be important, though we may need to augment the situations we have formalized (Figure 6.3). For example, for an hypothesis-oriented discussion we would refine the GOAL procedure to focus on one outcome value at a time, and incorporate transition t-rules for detecting and discussing shifts to another outcome (hypothesis revision).

10.4.5 Learning

Few ICAI systems include a *learning module* [Goldstein, 1977] [O'Shea, 1979] to serve as a critic of the tutoring methods used by the program. The problem is to assign credit and blame to the tutoring methods on the basis of evidence that they succeeded or failed in teaching material to the student. The system then adapts its tutoring methods to improve itself, as we described in the case of adaptive modeling. However, it is very difficult to separate effectiveness of tutoring methods from the noise contributed by student forgetting and the fact that certain material might be mentioned in different ways on several occasions in different problems.

10.5 Speculation about the Future: Cognitive Science

Intelligent Computer Aided Instruction is one subfield of AI that has become infused with ideas about human cognition. In the late 1970s we observe a shift in the kinds of questions researchers are asking. Collins is asking not just how to make a program answer a student's questions, but "What is a humanly plausible way of reasoning with uncertain knowledge?" Burton and Brown are asking more than whether a student knows a game rule, but "What kinds of assumptions might he be making

about the purpose of the game?" Goldberg wanted to do more than correct a student's proof, but also complete a partial solution, and asked, "What kinds of proofs do students write?" Stevens goes beyond telling the student that a causal argument is insufficient, he asks, "What model of the world would suggest this incorrect prediction?" Similarly, in examining justifications physicians gave for MYCIN's rules, we asked, "What makes an explanation of these rules satisfying?" These are some of the questions of cognitive science, a study rooted in the powers, limitations and habits of human thinking.

In a way, it is not too surprising that researchers attempting to design programs that can teach have turned to a study of how people think. We are starting to realize that there are constraints that affect how people make sense of new information and how they explain it to others. We don't know very much about these constraints today. We see evidence of memory limitations and a tendency to associate knowledge in certain ways, but we don't know much more than that. Maybe the evolving studies that characterize what people know, what they really remember about what they claim to know, and how they use this knowledge to solve problems will turn out to more useful to teaching than these basic "hardware" limitations.

The title of this dissertation mentions the *transfer of expertise*. The very fact that we are attempting to transfer knowledge through an interactive process between man and machine makes the study of human thinking important. Indeed, after several years of acquiring expertise from human experts and formalizing it for our programs, we have learned that we are capturing but a small part of what these people know. Moreover, we have witnessed the laborious process of writing down these principles,

and realized that the real heart of our problem might be to understand better how experts prefer (or need to) talk about what they know. One physician hands us a flowchart and we say, "Fine, now convert them to rules." An engineer is told, "Write rules!" and in a day has 200 of them, but still can't explain to the computer scientist how he solves a problem. It seems obvious that ICAI will benefit from the study of how people, who already know what we want to teach, organize and make use of their knowledge. There may be many moltings in between, but at least we will know what we are aiming for.

In fact, the symmetry of knowledge acquisition from a human expert and computer-aided instruction suggests that some day these programs may be one [Barr, Bennett, & Clancey, 1979]. By integrating the tutor with the knowledge-acquisition program, we would be in a better position to teach the material later, particularly if the way that the expert talks about what he knows reflects an underlying set of human constraints that is shared by the student who wants to make sense of the same material. We envision a tutor that would ask questions of the domain expert that would be relevant for teaching the knowledge it is acquiring. To do this, the tutor would need to have a good model of the constraints that will guide its interaction with the student, that is, a good model of how people learn. As matter of fact, the tutorial program would be simulating a student as it posed these questions and fit new facts into its knowledge base. Rather than reading a rule's premise and conclusion and hooking it into a list for later retrieval, *knowledge acquisition* by a program would have to be something closer to human learning.

Finally, the principles of discourse that now constitute GUIDON's tutorial

expertise could be thought of as being inverted as knowledge is acquired by the system. Once we manage to capture the expertise that good teachers use, then what GUIDON would say in a certain situation if it were teaching a student should be analogous to what it hears when a human expert is explaining something to it. We might imagine a single corpus of knowledge that characterized how people learn and explain what they know, and this *transfer of expertise expertise* could be used both to drive the interaction with a human expert, making provision for later teaching to a student, and to drive the tutorial interaction with a student. These principles might be the foundation for a science of cognition.

Appendix A

Formation of the AND/OR Solution Tree

The AND/OR tree used by GUIDON is a reconfiguration of the output of a MYCIN consultation. This reconfiguration requires a non-trivial analysis of the rules and conclusions made by MYCIN. The analysis occurs on three recurring levels: a goal, the rules that are used to determine it, and the subgoals that are needed to evaluate a rule.

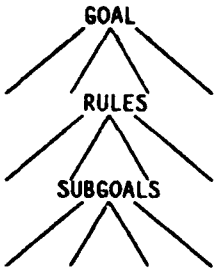


Figure A.1. Three Levels of Analysis for Constructing the AND/OR Tree

The difficult part of this analysis is on the third level--forming a logical representation of the rule in terms of subgoals using only the functions AND and OR. For example, the following is a rule and the representation of why it failed in a particular consultation:

RULE517

```
If:  1) The infection which requires therapy is meningitis,
     2) A: A smear of the culture was not examined, or
        B: Organisms were not seen on the stain of the culture
     3) The type of the infection is bacterial, and
     4) The age of the patient is less than 9 years,
     5) There is evidence of congenital malformation involving the central
        nervous system,
Then: There is evidence that the organisms (other than those seen on
        cultures or smears) which might be causing the infection is
        e.coli (.75), staphylococcus-coag-pos (.75), klebsiella-
        pneumoniae (.75) and pseudomonas-aeruginosa (.5)
```

```
FAIL-PARMS!: (OR (4 (AGE . PATIENT-538))(5 (CNSMALFORM . PATIENT-538)))
```

Notice that goals are indicated by a pair of the form (<parameter> . <context>). The entire AND/OR tree consists of pointers between parameter/context pairs and

rule/context pairs.[1] This representation enables GUIDON to determine efficiently whether or not a rule "fires" without re-evaluating the preconditions of the rule. For example, the rule above will fire as soon as the age of the patient or whether he had a congenital malformation is known. (Section 8 describes the overlay model in detail.)

The difficulty of the analysis stems from the variety of predicate functions that appear in MYCIN rules. Figure A.2 classifies these functions into categories that we will refer to frequently during the discussion that follows.

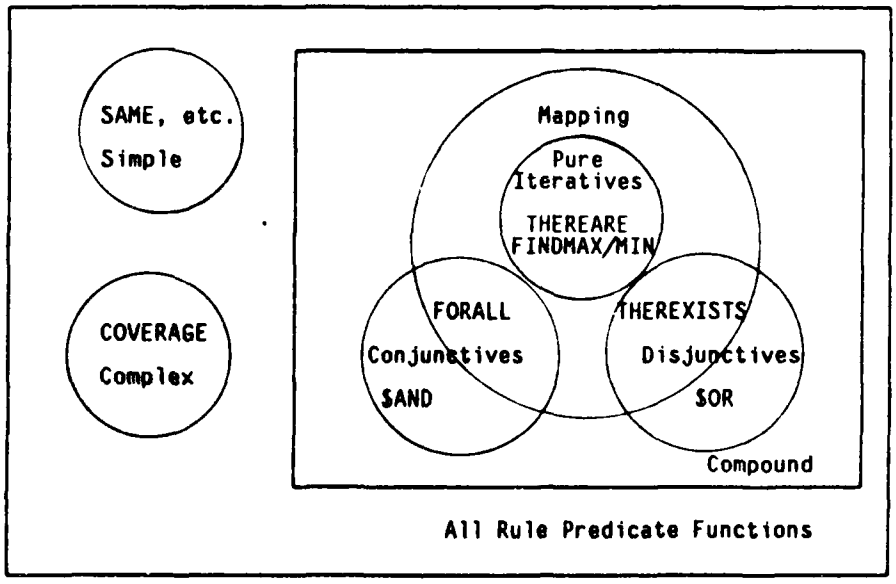


Figure A.2. Classification of MYCIN Predicate Functions

There are three basic kinds of functions:

1) **SIMPLE** functions such as SAME and MIGHTBE generally involve one subgoal, e.g., "the site of the culture is blood" references the subgoal SITE.

2) **COMPOUND** functions such as $AND are composed of simple functions.

[1] MYCIN parameters, such as SITE ("the site of the culture"), are said to be *traced* in particular contexts. Similarly, rules are *applied* in specific contexts. Thus, each node of the AND/OR tree is really a dotted pair.

3) COMPLEX functions perform arbitrary computation, generally involving several subgoals. For example, one of the two complex functions in MYCIN is described, "Count the number of cultures from the same site which were taken since this time, and . . . find the most recent of these cultures which was positive for an organism with the same identity as this organism...."

The input to the program that builds the AND/OR tree consists of the MYCIN knowledge base of rules and parameters and the result of a particular consultation. It was necessary to augment the consultation record (originally designed for use by the post-consultation question/answer program) to indicate in more detail which rule clauses were evaluated and whether or not they were found to be true. Overall, the analysis is a two-step process:

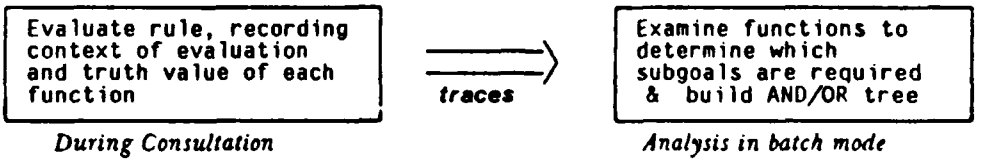


Figure A.3. Two-step Process for Constructing the Tree

It would be possible for the construction of the AND/OR tree to proceed during the consultation by modifying the predicate functions themselves. However, to keep the tutorial package independent of the consultation system, and to prevent slowing down the consultation, it was deemed better to separate the analysis into two steps.

The consultation record consists of rule conclusions, final values for goals that were pursued, and a trace of the evaluation of each rule (see Figure 4.6). This trace is stored as a property of the rule. It is built recursively by the predicate functions as each clause of the rule is considered. The format of the trace is shown in Figure A.4 in Backus-Normal form.[1] An APPLTRACE is stored as the APPLIED-TO property of a rule.

In the simplest form, a trace consists of the context in which the rule was evaluated and the fate of the evaluation: NIL if the rule succeeded, and a clause number if the rule failed, indicating which clause was not true. EXTRA is T if the rule was found to fail during a preview phase in which clauses having known subgoals are considered before new subgoals are pursued. For each rule clause that is a compound predicate function, an EXTRA trace is included that names the logic or mapping function and includes traces about its application. Note that the details for an internal

[1] A. Carlisle Scott helped the author design this part of the consultation program.

mapping function are of the same format as the trace for the rule as a whole. This is because each mapping function consists of a list of contexts (objects) and an internal $AND predicate, just like a rule. On the other hand, details for an internal logic function are provided by an EXTRA trace, consisting of a clause number or information about an compound clause.

```
APPLTRACE ← ( CNTXT FATE (EXTRA)* )

CNTXT ← an object in the context tree, e.g., CULTURE-1

FATE ← NIL | C#

C# ← a clause number, e.g., C3.

EXTRA ← T | C# | (C# FAIL/SUCC MAPFN (APPLTRACE)* ) |
            (C# FAIL/SUCC LOGICFN (EXTRA)* )

FAIL/SUCC ← FAIL | SUCC

MAPFN ← FINDMAX | FINDMIN | THEREARE | FORALL | THEREXISTS

LOGICFN ← $AND | $OR
```

Figure A.4. Format of the Consultation Record Trace

Figure A.5 shows a skeleton of a rule and a possible APPLTRACE that might be associated with it after a consultation. The trace indicates that the rule failed because the second clause was not true (a[1]). The EXTRA trace (b) indicates that the second clause, a FORALL iteration, failed because in the context of ORGANISM-3 the third clause of the internal $AND failed (c). The EXTRA trace associated with this third clause indicates that the $OR succeeded (d). (In other words, the $OR was true and negated, so the $AND of the FORALL was not true for ORGANISM-3, thus the second clause of the rule fails.)

```
PREMISE: ($AND  (SAME ...)
                (FORALL  organisms
                        [$AND   (NOTSAME ...)
                                (THOUGHTNOT ...)
                                (NOT ($OR (SAME ...)
                                          (SAME ...))
                                (SAME ...])
                (SAME ...))

APPLTRACE:
(CULTURE-1  C2  (C2  FAIL  FORALL (ORGANISM-3  C3  (C3  SUCC $OR)))
            a   b                                 c   d
```

Figure A.5. Rule Skeleton and Associated Consultation Record

[1] Small letters point to parts of the trace

The discussion that follows deals with the second phase of the analysis (refer to Figure A.3). Note that we will not refer to the APPLTRACE again. All references to whether or not a predicate succeeds or fails are determined by decoding the APPLTRACE.

As stated, the analysis takes place on three levels. An overview is given below, referring to program functions by name. Indentation indicates that a function is invoked by the one above it. Level of analysis is indicated by the functional arguments following function names.

```
EXPANDNODE -- goal

        SETUPDATE! -- rule
                MARKSUBGOALS -- rule
                        MARKCLAUSES -- rule

                                MARKCL1 -- clause
                                        MARKMAP -- mapping clause
                                        MARKSG1 -- subgoal
```

Figure A.6. Calling Structure of Functions for Constructing the Tree

Expanding a goal node (EXPANDNODE)

Expansion of a goal node (parameter/context pair) consists of analyzing each rule that might have been used to determine the goal. One might view this as a process of attaching rule/context pairs to the goal so that they hang below it.[1] For later access, these rules are grouped into those that fail and those that succeed. So the tree we are building looks like this:

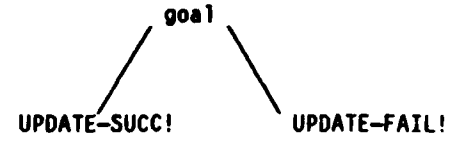


Figure A.7. Grouping Rules below a Goal Node

The format of the UPDATE-SUCC! and UPDATE-FAIL! properties is shown in Section 8.3.2.2. Each is a list of rule/context pairs with a marker separating self-reference rules from normal rules. Note that if a rule was not tried at all (because other rules gave the answer with certainty), then it is put on UPDATE-NOTUSED! (not shown here).

---

[1] This use of the term "expansion" is the same as, and was motivated by, its use for searching AND/OR game trees [Nilsson, 1971]

In short, process of expanding a goal consists of going down the list of rules for the goal (union of UPDATED-BY (consequent) and UPDATED-IN (antecedent)), and analyzing each rule in each context that it was applied. SETUPDATE! places each rule/context pair in the appropriate group of rules hanging from the goal, and then invokes MARKSUBGOALS to analyze the rule.

Analyzing a Rule (MARKSUBGOALS)

A nested history list is set up, starting with OR if the rule failed and AND if the rule succeeded. As subgoals are discovered, they are added to this logical representation of the rule.[1] Next, the consultation traces, premise of the rule, and whether or not the rule succeeded are passed on to the MARKCLAUSES.

Analyzing a Set of Clauses (MARKCLAUSES)

If a $AND succeeded, then all clauses are analyzed by MARKCL1 [2] so that all subgoals are placed in the history being built. If a $AND failed, only clauses that failed have their subgoals placed in the history. Subgoals in clauses that succeed are *unused*, and this information is passed on to MARKCL1.

The main problem is determining whether or not a clause succeeds or fails. If this can't be determined from the trace, i.e., the clause was not evaluated during the consultation, then the clause is evaluated at this time. Since additional case data may required to evaluate the clause, it may not be possible to classify it as failing or succeeding. In this case, subgoals in the clause are marked "unused," and the UNKNOWNPARMS property is placed on the rule for later use in the tutorial, e.g, when generating a quiz from the rule.[3] [4]

---

[1] A rule that fails is represented as a logical OR because there may be more than one clause that is false.

[2] "Mark a single clause"

[3] Currently, an iterative clause can only be marked as failing if it is the clause that is known to have caused the rule to fail, otherwise it is marked as "unused." To do better than this, we would have to carefully examine the clause to make sure that all information is available for it to be evaluated. Even then other rules might need to be invoked to supply parameter values. We have simplified the initial implementation by avoiding this problem. The effect in a tutorial is that GUIDON may present a failing rule after the student already knows that it is not applicable (due to a false iterative clause that GUIDON is ignoring).

[4] Self-referencing rules require special treatment. When EXPANDNODE detects that is is passing self referencing rules on for analysis, it sets an internal pointer so that the value of the goal is the value that had been computed in the consultation after all normal rules had been done. Thus, the rule will be evaluated just as it was during the consultation (see [Shortliffe, 1976] for details)).

The presence of a NOT in the rule complicates matters. It is then possible that a clause that succeeds may be "unused" and one that fails may be included in the history of why the rule succeeded. To bring about this inversion, MARKCLAUSES is parametrized to look for a particular fate (success or failure), and this fate is treated as a switch that is inverted when NOT is encountered by MARKCL1.

Marking a Particular Clause (MARKCL1)

The next step is to extract the subgoals that must be known for a clause to be evaluated. The analysis separates into a handful of cases dependent on the function that represents the clause: $AND, $OR, NOT, mapping function, complex functions, and simple functions. In all cases, an AND or OR is pushed onto the history list when analysis begins, subgoals are added (by MARKSG1) as they are discovered, and the list is popped (closed) when analysis of the clause is complete. In general, AND is pushed when the clause is to be marked "succeed" and the function is a conjunction (FORALL and $AND), or the clause is to be marked "fail" and the function is a disjunction (THEREXISTS and $OR). OR is pushed for failing conjunctions and successful disjunctions. If the clause is within a NOT, then AND and OR are reversed.

Analysis of $AND and $OR involves sending the clauses of the conjunction or disjunction as a set to be marked by MARKCLAUSES, with appropriate switches set. When NOT is encountered, a flag is inverted and the clause it qualifies is sent to MARKCL1. Mapping clauses are analyzed by MARKMAP (discussed below). Templates are used to pick out the parameters mentioned in complex and simple clauses. Parameters are paired up with relevant contexts (determined from the context in which the rule was applied by examining the context tree), and then sent to MARKSG1 to be added to the history list.

Analyzing a mapping clause involves first computing the list of contexts over which the iteration was performed. The internal $AND is then analyzed in each context. If the clause failed because there were no objects of that type in the consultation, e.g., no current therapy, then the parametrized name of this list is sent to MARKSG1, e.g., "current therapies of patient-538" is represented internally as the pair (CURTHER . PATIENT-538).

In general, iterations that succeeded are so marked and others are "unused." For example, a THEREXISTS might have evaluated the internal predicate ($AND) in several contexts (ORGANISM-1, ORGANISM-2...) before a true case was found. The true $AND is sent to MARKCL1 to be marked so that each subgoal is added to the successful history of the rule.[1] Again, this analysis is inverted if a NOT surrounds the

[1] Note that a successful THEREXISTS clause should be recorded in the form: (OR (AND ...) (AND ...) ...) in which there is one AND for each successful iteration. However,

mapping clause. Note that all contexts (iterations) of a pure iterative function are marked identically.

Entering a Subgoal into the AND/OR Tree (MARKSG1)

At this point we have parameter/context pair and a mark: SUCCEED, FAIL, or UNUSED. Moreover, this mark indicates (if we have done all of our inversions correctly) whether the rule succeeds or fails, or is not used at all. If the mark is SUCCEED, the rule is entered on the USE-SUCC! property of the subgoal (rules that succeed that use the subgoal) (refer to Figure 4.5). The subgoal is placed on the CONSEQ-SG! or ANTE-SG! property of the goal that the rule updates (this is a global variable set by EXPANDNODE). If the mark is FAIL, the rule is added to the CAUSE-FAIL! property of the subgoal and the subgoal is added to the FAIL-SG! property of the goal. If the mark is UNUSED, the relevant properties are UNUSED-PARMS! and UNUSED-SG!. Inverse pointers from subgoal to goal have not been found to be useful.

Finally, if the mark is not UNUSED, the subgoal is added to the logical representation of the rule. When MARKCLAUSES has analyzed the entire rule, MARKSUBGOALS will save this history as the SUCC-PARMS! or FAIL-PARMS! property of the rule.

---

we only mark the single (first) iteration that succeeded (and the rest are UNUSED). This is because we do not evaluate iterations that MYCIN did not attempt (because other rules would probably need to be invoked to do this). The effect during a tutorial is that a student might know that the THEREXISTS succeeds, based on a later iteration that MYCIN never tried, while GUIDON will believe that it is too early to fire the rule because there is insufficient evidence to evaluate the only successful iteration that it knows about. The best reason for ignoring this problem for now is that there are in general a small number of possible iterations, which reduces the possibility of there being multiple iterations that are true.

Appendix B

Question Generation and Response

A basic feature of GUIDON is its ability to generate questions from MYCIN rules. There are four distinct steps in this process:

1) Selecting a question type

2) Converting the rule and printing the question

3) Evaluating the student's answer

4) Responding to the student.

A question format is described in terms of the format of the premise part of the question and the format of the action or conclusion part. There are 10 possible premise formats and 5 possible action formats, yielding 19 valid combinations that have been implemented. Because of the number of question formats and complications deriving from features of the rules, the code for generating questions is very complex. This appendix describes the valid question formats, relevant rule features, and strategies for responding to the student.

Question Formats

Question formats fall into two groups, 13 valid combinations that are selected randomly and 6 special purpose formats used to respond to a student's hypothesis or present a rule. The randomly chosen formats are summarized in Figure B.1. A dollar sign ($) separates the premise and action parts of the question.

ACTION \ PREMISE	FACTS	RIGHT-HYPOTHESIS or WRONG-HYPOTHESIS	MULTIPLE CHOICE
ACTUAL VALUE	"Given <prem facts>, does this $ tell you <actual concl>?"	"Given <prem facts>, if <hypothesis>, would this $ tell you <actual concl>?"	"Given <prem facts>, what <hypothesis> would $ tell you <actual concl>? Select from 1) ... N)"
WRONG VALUE	same as ACTUAL VALUE	same as ACTUAL VALUE (invalid for WRONG-HYPOTHESIS)	invalid
MULTIPLE CHOICE	"From <prem facts> one can $ conclude that <goal> is: Select from 1) ... N)"	"Given <prem facts>, if <hypothesis>, one could $ conclude that <goal> is: Select from 1) ... N)"	invalid
CONFIRM	"Do/Does <prem facts> $ tend to confirm the conclusion that <other rule's concl>?"	"Given <prem facts>, if <hypothesis>, would this $ tend to confirm the conclusion that <other rule's concl>?"	"Given <prem facts>, what <hypothesis> would $ tend to confirm the conclusion that <other rule's concl>? Select from 1) ... N)"

Figure B.1. Randomly-selected Question Formats

Key: <prem facts> = listing of case data used by clauses
<actual concl> = conclusion made by the rule
<hypothesis> = certain clauses of the rule stated hypothetically
<goal> = parameter to be determined
<other rule's concl> = conclusion made by another rule

The six special formats are given below with comments. ("S" replaces "student.")

PREMISE/ACTION	FORMAT	COMMENTS
INVALIDFACTORS/ ACTUAL VALUE	"What facts about this case tell you that <actual concl>?"	Requests S to support his hypothesis by stating relevant factors. None will be accepted.
KNOWNFACTORS/ ACTUAL VALUE	same as INVALIDFACTORS	Quiz based on multiple rules. Many factors might be valid. Schema (context) factors are accepted, but not required.
NEWFACTORS/ ACTUAL VALUE	"What factors would $ tell you that <actual concl>?"	S is asked to mention factors that haven't been supplied for this case yet. Used to present a rule.
NEWSUBGOALS/ ACTUAL VALUE	same as NEWFACTORS	Here the factors must be established by applying other rules. Used to introduce discussion of the rule that uses these subgoals.
GIVENFACTS/ FILLIN	"Here is some relevant data you could have asked for: <case data>. $ What does this tell you about <goal>?"	For quick presentation of new data and a rule that uses it.
KEYFACTS/ FILLIN	"Consider <key facts>. $ What does this tell you about <goal>?"	This is a standard format for testing the student's knowledge of rules that have succeeded.

Figure B.2. Special-purpose Question Formats

Selecting Question Formats

The rule packet GENQTYPE.RULEKP024 determines which formats are logically valid for a given d-rule, and chooses randomly from the candidates. Besides interactions between the premise and action formats, the most relevant rule features are the outcome of the rule's application (success or failure) and the presence of true/false parameters. No attempt will be made here to describe the kinds of interactions we have found; the interested reader will find them in GENQTYPE.RULEPK024.

It is important to note that randomly generating questions by modifying rules is risky. Resulting questions can be trivial, or, even worse, a modification might produce another correct rule (e.g., see how T-RULE24.02 handles this). Even without modifying a rule, GUIDON can run into trouble if the student is asked to fill in an answer or choose from all possible values. Consider the following quiz:

PREMISE: *multiple choice*
ACTION: *actual value*

```
Given the infection which requires therapy, what value for the diagnoses
which are consistent with Pt538's clinical history would tell you that Virus
is one of the organisms (other than those seen on cultures or smears) which
might be causing the infection (.2)?
(select from the following):
        1) OTITIS-MEDIA
        2) EPIGLOTTITIS
        3) MUMPS
        4) LEPTOSPIROSIS
        5) COCCIDIOIDOMYCOSIS
    ** 3

No, you could make this conclusion if epiglottitis were one of the
diagnoses which are consistent with Pt538's clinical history [RULE580].
```

Figure B.3. Question with Unrecognized Correct Answer

It so happens that MUMPS is a correct answer that can be deduced from MYCIN's rule set, but is not stated directly in this rule. (In particular, there is a rule "MUMPS => type of infection is VIRAL" and another rule that states "VIRAL type => VIRUS might be causing the infection.") Part of the problem here is that parameters are not always used consistently and related inferences are separated into several rules. When disallowing an inference of the form "A => B," GUIDON really needs to look beyond the single rule it is quizzing about, to prove that there isn't another rule of this form [1] or to prove that there isn't a chain of reasoning that ties A to B, as shown in the example here.

---

[1] Often analogous to the rule being quizzed about, say C => B, where C and A are in the same class, e.g., kinds of rashes.

Modifying the rule

Modifying the rule so that it can be printed as a question requires function templates, the list of parameter/context pairs that caused the rule to succeed or fail, the key factors of the rule, alternative values for parameters used in the rule, and the actual conclusions made by the rule, if it succeeded. To make the question more concise, premise factors are printed in general terms ("information about the type of the infection") rather than in full detail ("the fact that the type of INFECTION-1 is bacterial (.75)"), and only key factors are mentioned. Ad hoc code changes verbs to plural form, converts verbs to subjunctive form, inserts punctuation, and capitalizes initial words.[1] Random values and certainty factors are used to create "wrong" premise and action parts to the question. Values are chosen from among those that actually appear in MYCIN rules so that they are plausible, and certainty factors are shifted enough to be meaningfully different, without being obviously wrong.

It would, of course, be useful to generate questions that make a point, as is done in WHY. Tabular rules offer more opportunity for generating meaningful questions because premise and action are related by a case statement that pairs alternative situations with appropriate inferences. This is a convenient representation for generating hypothetical questions. Though, again, without the logical underpinnings that justify the rule's conclusions, GUIDON cannot quiz the student about the principles involved (the point of Section 5.2).

Two other points about GUIDON's question generation are worth mentioning. "DONTKNOW" is always a valid reply; it is treated like "HELP," which causes a fill-in question to be converted to multiple-choice format. WRONG-HYPOTHESIS and RIGHT-HYPOTHESIS involve restating a clause in subjunctive form that caused the rule to fail. RIGHT-HYPOTHESIS prints the clause as it appears in the rule; WRONG-HYPOTHESIS replaces the value of the clause with another possible value, e.g., "petechial rash" is changed to "purpuric rash."

Evaluating the Student's Answer

During question generation, flags and variables are set to indicate the nature of the question ("multiple rules flag," "original tabular case," "correct values"). These are used in a straightforward way to logically evaluate the student's reply.

---

[1] William van Melle's improvements to MYCIN's string manipulation and printing functions were very helpful.

Responding to the Student

Code for replying to the student is as complex as that used for generating a question, for it again breaks down into 19 separate cases involving different data structures. The basic response strategy is to list right, wrong and missing answers, and state what could have been concluded given the facts stated, or when the conclusion stated would be true. If the rule has been modified, details are provided even when the student is correct. The rule's key factor is mentioned if it has not been distinguished in the question.

Ideally, these tutorial actions should be stated in tutorial rules to make them more accessible for change. However, most of the strategies are implicit in print statements that go through many contortions to combine fixed-text with translations of parameters and values. This problem has not been solved in t-rules either, where "SAY:" with a tag indicates a print statement. A higher level specification language would be useful here.

Summary of the Questioning Strategy

GUIDON's 19 formats for generating questions can be classified according to the Socratic probing and correcting tutorial methods formalized by Collins and Stevens [Stevens & Collins, 1977]. The formats of Figure B.1 correspond to "ask the student to form or verify the correctness of a rule." Figure B.2 contains two methods (split by the double line): "Ask for relevant factors" and "Ask for the value of a dependent variable".[1]

Generally, GUIDON's distinctions about what method to apply (its questioning strategy) is based on the convenience of fitting a given rule to a particular format and the assumption that Figure B.2 questions are more difficult than Figure B.1 questions. Thus, the strategy is based on variety, difficulty, and the dialogue situation (what has been discussed). The main exception occurs during response to a student's hypothesis. Then, selection of question type parallels the Collins/Stevens model, e.g., when debugging an incorrect student hypothesis, GUIDON asks the student for prior factors that support his value for the dependent variable (KNOWN or INVALID factors format). At other times in the dialogue, the Collins/Stevens enumeration of questioning strategies would be useful if we wanted GUIDON to follow up on wrong student replies to questions, rather than always correcting the student by a statement. For example, when the student mentions insufficient factors, the program could form a hypothetical question to point out the missing factors, instead of stating them.

[1] Note that the distinction among invalid, known, and new factors and between given and key factors is based on whether or not the rules involved can be applied at this point in the dialogue; these formats do not represent different methods.

Appendix C

GUIDON Handbook

This is a condensed form of an operating manual for GUIDON. This handbook is generated by a program from text files that serve as program-aids during the tutorial. Note that the text of section VII is specific to the rule base on which GUIDON is built.

I. What is GUIDON?

GUIDON is a tutorial program that can discuss particular problems (cases) with a student in mixed-initiative dialogue. A "diagnostic game" is played: the student is given some information about the problem (the "case") and must pursue other information, leading to solution of subgoals and a final hypothesis about the problem (a "diagnosis"). While the tutor adheres to a goal-directed discussion, the student can change the topic under discussion at any time.

GUIDON has access to a fully worked-out solution of the case. It uses the solution as a basis for offering assistance at any time, making hints about conclusions the student should be reaching, and interpreting and responding to student hypotheses about relevant subgoals. In addition, a wide variety of options are available to the student that make it possible for him to get an overview of the given case and explore subgoals according to his interests. For example, the student "profile" can cause GUIDON to indicate when a final conclusion about a subgoal can be drawn, and to mention when any conclusion at all about the current topic can be drawn.

It is assumed that the student is familiar with types of domain specific "objects" and "measurements" involved in solving a case. For example, if a case dealt with diagnosis of car horn failures, it is assumed that the student would know what a car horn is and realize that mechanical and/or electrical principles are involved in failures. The instructional objective of GUIDON is to teach the student METHODS for reaching conclusions about the problem (e.g., a rule for relating horn symptoms to possible faults) and STRATEGIES for applying these methods.

The sections below duplicate most of the text that is available online to a student (rules appear in a separate document). It is presented here to serve as an introduction to the program and for routine reference during a session.

II. The Role of the Student and His Options.

Brief Description of the Tutorial Session

You are to play the role of consultant for a patient suspected to have an infection. After initial data has been presented, you should request more information about the patient, decide which organisms (if any) require therapy, and prescribe treatment. The tutor will direct you, according to the profile you set. Its behavior can vary from leaving you alone until you are ready to make a diagnosis, to directing your thinking at every step. Feel free to change your profile or request HELP to solve the problem at any time. Use a question mark (?) whenever you want to see the list of options open to you. Remember that the purpose of this session is to present MYCIN's rules of infectious disease diagnosis and, in the second part, the criteria

of therapy selection.

Type control-G to call GUIDON. You will receive the initiative after the next question is asked.

III. The Student Profile

The user profile is a set of flags that are used by the EXPERT program. As the student receives more data about a case, the EXPERT applies domain rules that use the information and draws new conclusions. These flags can be set to control what the EXPERT tells the user about these rules. A special flag for mentioning rule conclusions can be set for "relevant" conclusions only, so rules that aren't relevant to the topic under discussion won't be mentioned. The PROFILE option permits the student to change the settings of these flags at anytime, so experimentation is encouraged.

The standard profile setting is for a goal-directed dialogue. The expert indicates when a final conclusion can be drawn for each topic, and mentions when any conclusion at all can be drawn about the current topic. The student is assumed to be a novice (1 on scale from 0 to 4).

{-- *omitted here; see Section 7.1.3* --}

Meaning of each profile flag:

{-- *omitted here; see Figure 7.1* --}

IV. Option Summary

V. One Line Restatement of Options

- Options available during discussion of a topic:

OPTION	INTENT
Options for Determining Dialogue Context	
RULE	"What rule are we discussing?"
TOPIC	"What topic we are discussing?"
ORIENT	"Where are we? How did we get here? Where are we going?"
Options for Acquiring Case Data	
DATA	"Give me this case data."
BLOCK <block name>	"Give me all of the case data in this block."
ALLDATA	"Tell me all of the case data relevant to this topic."
Options for Getting Direct Assistance	
HELP	"Suggest something for me to consider."
HINT	"Give me a hint."
TELLME	"I give up; tell me the conclusion the expert made."
Options for Information Retrieval	
USE	"How can apply information about this topic to this problem?"
CONCLUDE <value>	"How can I conclude that the current topic is this value?"
SUMMARIZE	"List the data and conclusions that the expert made."
DETAILS <remark index>	"Give me details about this remark."
FACTORS	"What case data and subgoals should I consider?"
SUBGOALS	"Show me a tree of goals that we will pursue."
PENDING	"What case data and subgoals are still pending for this topic, and for which goals should I try to make an hypothesis?"
Options for Conveying What You Know	
IKNOW	"I know this."
HYPOTHESIS	"I want to tell GUIDON a conclusion I have drawn."
Options for Changing the Topic	
DISCUSS	"Let's discuss this topic."
QUIZ	"Ask me a question."
ALLRULES	"I don't want to ask any more questions about this topic; let's discuss the relevant rules."
QA	"I've got a general question for you."

```
     control-G          "Stop asking me questions; I want to say something."

Special Options

      ?                 "What can I do here?"
     OPTIONS            "Show me a table that summarizes my options."
     <option> ?         "Tell me more about this option."
     COMMENT            "I want to send a message to the person in charge!"
     DEFINE             "Give me a definition of this word."
     PR <rule no.>      "Show me this rule."
     JUSTIFY <rule no.> "Justify this rule."
     LITERATURE <rule no.>"Give references for this rule."
     PTR <T-rule no.>   "Show me this tutorial rule."
     PROFILE            "I want to change what the Expert tells me."
     STOP               "I want to stop now.  Terminate the dialogue."
```

When none of these options is specified, the tutor assumes that you are asking for a piece of data, i.e., the DATA option.

Remember: HELP is a request for assistance to solve the problem; ? prints your options; use this when you aren't sure what is a meaningful reply or action to take. Also, follow an option name by "?" if you want details about it.

{-- *omitted here* --}

VI. Detailed Description of Each Option

VII. Parameters of the MYCIN System

Below are listed all of the possible topics that can be discussed during a MYCIN/GUIDON tutorial. They are grouped by the "object" that is qualified. In a given case, there may be more than one object of each type, e.g., more than one culture, so we will refer to CULTURE-1, CULTURE-2, etc.

For many of the options you will find it easier to use the abbreviated name of the parameter, e.g., ABDOWOUND in the command: "DISCUSS ABDOWOUND." Alternatively, you can also use phrases like "How is the fact that drugs were given adequately to the patient relevant in this case?"

{At this point, the parameters are listed, grouped by culture, organism, patient, etc. For MYCIN, this listing requires six pages. Here we only show the parameters that appear in Figure 5.20.}

A. The Parameters Associated with a patient are:

```
AGE:              The age of the patient
ALCOHOLIC:        Whether the patient is an alcoholic
                       patient's age group
ANIMALEXP:        Whether the patient has frequent contact with
                       animals
BURNED:           Whether the patient has been seriously burned
CARDIACSURGERY:
                  Whether the patient has had cardiac surgery
CBC:              Whether the results of a recent CBC of the
                       patient are available
CLINEV-TUBERCULOSIS:
                  Whether Tb is one of the diagnoses consistent with
                       the patient's clinical history
CNSMALFORM:       Whether there is evidence of congenital
                       malformation involving the central nervous
                       system
CNSRADIATE:       Whether the patient has undergone recent
                       radiation therapy to the CNS
CROWD:            Whether the patient does live in a crowded
                       environment
CXRAB:            Whether the patient's X-ray is abnormal
CYTOTOXIC:        Whether the patient has received cytotoxic drugs
                       just prior to or during this infection
DIABETIC:         Whether the patient is a diabetic
DIAGNOSES-LYMPHOMA:
                  Whether lymphoma is one of the diagnoses of the patient
DIAGNOSES-LEUKEMIA:
                  Whether leukemia is one of the diagnoses of the patient
EPIDEMIC:         The epidemic diseases to which the patient has
                       been exposed recently
```

EPIGLOTTITIS:	Whether epiglottitis is one of the diagnoses consistent with the patient's clinical history
EXPOSURE:	The contagious diseases to which the patient has been exposed recently
GU-MANIP:	Whether the patient has had a genito-urinary manipulative procedure
HEADINJDEF:	Whether the patient has a head injury defect
HEADTRAUMA:	Whether the patient has had a severe head trauma
JUNKIE:	Whether the patient has a history of illicit intravenous drug use
MUMPSYM:	Whether the patient has shown symptoms of mumps
NEUROSURGERY:	Whether the patient has had neurosurgery
NOSOCOMIAL:	Whether the infection was acquired while the patient was hospitalized
OCNERVE:	Whether the patient has evidence of ocular nerve dysfunction
OTITIS-MEDIA:	Whether otitis-media is one of the diagnoses consistent with the patient's clinical history
RACE:	The race of the patient
RASHES:	Whether the patient has a rash or cutaneous lesions
SICKLECELL:	Whether the patient has sickle cell disease
SKININFECT:	Whether the patient has evidence of serious skin or soft tissue infection
SPLENECTOMY:	Whether the patient has had a splenectomy
STEROIDS:	Whether the patient has received corticosteroids just prior to or during this infection
TBRISK:	Whether the patient has a tb risk factor
WBC:	The white count from the patient's peripheral CBC (in thousands)

Appendix D

Introductory Orientation for New Students

As a form of introduction for a new student, GUIDON prints the goal rule of the expert system and a sketch of subgoals at the start of the dialogue (following t-rule1.04). The text shown below was extracted from the sample protocol (Section 3). See Section 6.2.3.1 for discussion.

Presentation of the Goal Rule

```
In order to determine the therapeutic regimen of J.Smith we will want to
consider:
   1) A: organisms isolated from J.Smith, and
      B: organisms noted on smears of J.Smith, and
      C: negative cultures of J.Smith, and
      D: suspected infections without microbiological evidence, and
      E: current drugs of J.Smith, and
      F: prior drugs of J.Smith, and
   2) the organisms which require therapy, and
   3) A: the organisms (other than those seen on cultures or smears) which
      might be causing the infection, and
      B: whether the organisms isolated from the pending csf culture
      (CULTURE-1) should be considered for therapy, and
      C: the associated infections for which therapy should cover, and
      D: the organisms that will be considered to be causing the meningitis
      (INFECTION-1) for the purposes of therapy selection
```

Presentation of Major Subgoals

Sketch of the tree of subgoals for determining the therapeutic regimen of J.Smith:

7a. The therapeutic regimen of J.Smith
- 7b. The organisms (other than those seen on cultures or smears) which might be causing the infection
- 7c. Whether J.Smith has a head injury defect
- 7d. The infection which requires therapy
- 7e. The type of the infection
- 7f. The diagnoses of J.Smith
- 7g. Whether organisms were seen on the stain of the pending csf culture (CULTURE-1)
- 7h. Whether the organisms isolated from the pending csf culture (CULTURE-1) should be considered for therapy
- 7i. The organisms that will be considered to be causing the meningitis (INFECTION-1) for the purposes of therapy selection

(You can produce a full tree of subgoals for any topic by the SUBGOALS option.)

Appendix E

Representation of Discourse Procedures

One purpose of GUIDON is to provide a framework for testing teaching methods. Therefore, it is advantageous to have a representation that makes it convenient to display and vary the strategies that the tutor uses for guiding the dialogue. Using methods similar to those used in knowledge-based programs, we have formalized the tutorial program in rules and procedures that codify tutorial expertise. Here we will consider how the INTERLISP representation of t-rules satisfies the design criteria of transparency and flexibility.

Example of Internal Representation

Figure E.1 illustrates the internal form of a t-rule for presenting a (successful) d-rule to the student. The domain rule has already been selected and variables have been set that describe it (e.g., FACTORSTODO, the number of case data factors mentioned in the rule's premise that remain to be asked by the student). SUBGOALPC is a pair of the form (<parameter> . <context>) that was set when the rule was analyzed; CURGOALPC is the topic currently being discussed. Notice that substep ii. sets a flag when the GOAL procedure is invoked; this flag tells GOAL not to reintroduce the new topic (see Figure 4.9). The description of syntax below will be general at first; details will gradually accomodate this specific example.

```
[RULE 31.04 (IF (AND (ZEROP FACTORSTODO)
                     (EQP SGSTODO 1))
                THEN (PROGN (SUBSTEP\\ i. (SAY\\ (CLISP <"Try to determine"
                                                        ! (CPTRANS! SUBGOALPC)
                                                        '%. EOL >)
                                                 'SUGGESTSG))
                            (SUBSTEP\\ ii. (DISCUSS\\ GOAL.PROC001 SUBGOALPC
                                                        CURGOALPC T))
                            (SUBSTEP\\ iii. (DISCUSS\\ ENDOFRULE.PROC017
                                                        RULEFOCUS CURGOALPC)))
                                       (* Subgoals are obvious, he must not
                                          know about this one)]
```

Figure E.1. INTERLISP Form of a Sample T-rule

Syntax of Discourse Procedures

As shown in Figure 6.1, the action of a t-rule is a discourse procedure. Figure E.2 shows the general syntax of a discourse procedure.

```
(PROG ( <local variables> )
      (STEP\\ 1   ...)
      (STEP\\ 2   ...)
         .
         .
         .
      (STEP\\ <n> ...))
```

Figure E.2. Syntax of a Discourse Procedure

The steps (shown as "..." above) are compiler macros listed in Figure E.3.

Symbolic Form	Operation
(DISCUSS\\ <discourse procedure>)	invokes another procedure
(STRATEGY\\ <rule packet>)	invokes a packet of t-rules
(GO\\ <label>)	"go to"
(RETURN\\ <value>)	return from procedure
(COND\\ <if-part> <then-part> <else-part>)	simple conditional nested within a t-rule
(FOR\\ <set> <primitive function> <free variable name>)	iterative: for all members of the set, do the function
(SAY\\ <remark> <tag>)	prints remark for student
(ASK\\ <question> <tag> <yes/no-flag>)	asks student a question

Figure E.3. Primitive Functions of Discourse Procedures

In addition, one of two special operators can appear in place of a procedure step: a debugging statement (DEBUG\\ <*message*>) prints information on the terminal and/or a trace file and an initialization statement (INIT\\ <arbitrary INTERLISP code>) is used to set flags and counters used by the t-rules. Notice that every print statement (SAY\\) is uniquely tagged; this is useful for debugging.

Syntax of T-rule Packets

T-rule packets are also INTERLISP procedures, however they generally consist of a single step that has an ordered list of t-rules of the form (DO-ONE\\ <list of t-rules>) or (DO-ANY\\ <list of t-rules>). The first stops trying t-rules as soon as one succeeds; the second will apply any t-rule whose premise is satisfied. T-rule packets always accomplish a single task, e.g., presenting a predetermined d-rule. A t-rule is an INTERLISP record of the form:

```
(RULE    <number>
         (IF <if-part> THEN <then-part> ELSE  <else-part>)
         <comment> )
```

In practice, it is often convenient for a t-rule packet to return a value, so they are written with two steps, one with DO-ONE\\ or DO-ANY\\ and a second with RETURN\\. Moreover, in the current version of GUIDON there is frequently just one rule for a particular task, so the rule is simply placed "inline" in the discourse procedure that invokes it. Figure E.4 shows the second step of the procedure for discussing a goal. Notice that a t-rule's action does not bind local variables, so a PROGN is used (this makes it possible to simply use RETURN\\ to guarantee return from the surrounding discourse procedure).

```
[STEP\\ 2
   (RULE 1.02 (IF (KNOWN.BY.DEFN? CURGOALPC)
                  THEN (PROGN (SUBSTEP\\ i. (STRATEGY\\ DEFNRULF RULEPK042
                                                  CURGOALPC DEFINITION.RULE))
                              (SUBSTEP\\ ii. (DISCUSS\\ RELATEDRULES.PROC011
                                                  CURGOALPC DEFINITION.RULE))
                              (SUBSTEP\\ iii. (DISCUSS\\ NOTDISCUBRULES.PROC007
                                                              CURGOALPC))
                              (SUBSTEP\\ iv. (ANSWERBLKINFO NIL CURGOALPC))
                              (SUBSTEP\\ v. (FORCE.UPDATE CURGOALPC))
                              (SUBSTEP\\ vi. (RETURN\\]
```

Figure E.4. A Single T-rule as a Procedure Step

Translation of Rules and Procedures

GUIDON's rules and procedures are "translated" from INTERLISP form to more or less readable English using an extension of the techniques developed for translating MYCIN's rules. A special procedure (MACPROSE) is used to format translations (print them in indented form). Forms that are not handled by this procedure are converted to a lists of strings by the MYCIN function PROSE. Translation mainly consists of recursively substituting translations of arguments in the string template that describes translation of a function.

For example, the code (ZEROP FACTORSTODO) is translated by looking up the template for ZEROP, ((1) is "zero"), and substituting the translation of its first argument for "(1)." Templates can include arbitrary LISP code that computes the appropriate template forms or output strings, e.g., to translate EQUAL as "equal to" when numbers are involved and "the same as" otherwise. In addition, calls to STRATEGY\\ or DISCUSS\\ are printed to indicate that certain flags are being set. This is done by checking to see which actual parameters have a corresponding formal parameter with the FLAG property. [1]

It is possible to randomly access any t-rule and translate it. The algorithm is as follows:

1) Analyze the t-rule name (T-RULExxx.nnn) to determine the number of the procedure in which it appears (xxx); nnn is the index of the rule in that procedure.

2) Analyze the list of discourse procedure names (<name>.DISCPROCxxx or <name>.RULEPACKETxxx) to find which contains the t-rule.

3) Retrieve the filemap of the disk file that contains this function to determine the byte pointers that indicate where the function begins and ends.

4) Search the file between the byte pointer locations for the string "(RULE xxx.nnn" or "[RULE xxx.nnn" and return the LISP expression at this point. [2]

5) Use MACPROSE to translate the record (RULE xxx.nnn ...).

Thus, it is possible to index, access, and annotate t-rules as modular units (just like MYCIN rules), while they are organized into compiled procedures that control their evaluation.

---

[1] However, when the argument in the call to STRATEGY\\ or DISCUSS\\ is itself an argument to the function that contains the call, it is not possible to say which value is being passed on.

[2] This code was written by Larry Masinter and Bill van Melle.

Appendix F

Discourse Procedures and Tutorial Rules

Discourse Procedures (PROC)

ADVICEAFTERHYP.PROC039 391
ALLRULESFAILPARM.PROC035 391
BLOCKDATA.PROC034 391
COMPLETEDPARM.PROC005 392
CONSULT.PROC003 393
DEEPERSG.PROC019 393
ENDOFRULE.PROC017 394
FINISHGOAL.PROC006 395
FREEGOAL.PROC002 395
GIVENPARMDATA.PROC012 396
GOAL.PROC001 398
HELPFORGOAL.PROC016 402
HELPFORHYP.PROC018 403
HYPOTHESIS.PROC014 403
NEVERASKEDPARM.PROC015 406
NOTDISCUBRULES.PROC007 407
ORIENTATION.PROC036 408
PLEASEDISCUSS.PROC022 408
QUIZRULE.PROC021 409
RECENTFAILS.PROC010 410
RECENTSUCC.PROC043 410
RELATEDRULES.PROC011 411
REMAININGTASK.PROC047 412
RULE.PROC004 413
SPECIFICHELP.PROC020 414
TELLMEOPTION.PROC045 415
UNKNOWNRULES.PROC046 416

Rule Packets (RULEPK)

BACKGROUND.RULEPK040 416
CHOSENRULEINTRO.RULEPK038 417
DEFNRULE.RULEPK042 417
EXPERTDIDNTASK.RULEPK033 418
FAILRULE.RULEPK023 419
FOCUSRULES.RULEPK037 421
GENQTYPE.RULEPK024 422
HYPVALRESPONSE.RULEPK025 424
IKNOWOPTION.RULEPK044 426
NONEEDTOASK.RULEPK032 427
PRESENTVALS.RULEPK026 428
RULEINTEREST.RULEPK027 428
RULEUSED.RULEPK028 430
RULEUSEDSREF.RULEPK029 432
SAPPLIEDRULE.RULEPK041 433
SUGGESTHYP.RULEPK030 434
SUGGESTRULE.RULEPK031 435

ADVICEAFTERHYP.PROC039

Purpose: Offer a summary of what is known so far and the opportunity to terminate discussion of the hypothesis topic

Step1: T-RULE39.01
If: The expert has not applied all relevant rules for the hypothesis topic
Then: Offer an appraisal of what more needs to be done for this goal, suggesting that the topic be terminated, if appropriate [Proc047]
Else: Substep a. Say: oktoask
Substep b. Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate
Substep c. Record that discussion of the hypothesis topic is complete

ALLRULESFAILPARM.PROC035

Purpose: Discuss the fact that all rules fail for the goal currently being discussed and consider whether it is appropriate to request the reported value

Step1: Say: no-successful-rules
Step2: T-RULE35.01
If: 1) The goal currently being discussed has not been discussed before,
2) The student has not requested the reported value already, and
3) It is possible for the expert to request the value of the goal currently being discussed
Then: Say: norules-so-prompt

Step3: Present all relevant case data for this goal (the ALLDATA option)
Step4: T-RULE35.02
If: It is appropriate now to request the reported value for the goal currently being discussed
Then: Substep i. Say: case-data
Substep ii. Tell the student the reported value for the goal currently being discussed

BLOCKDATA.PROC034

Purpose: Present the related factors that form a block of data

Step1: T-RULE34.01

If: Related factors that form a block of data has not been discussed before
Then: Substep i. Give a fixed-text definition for related factors that form a block of data
Substep ii. Record that related factors that form a block of data has been mentioned in the dialogue

Step2: Give the values of the related factors

COMPLETEDPARM.PROC005

Purpose: Discuss final conclusion for the goal currently being discussed by giving the final answer or discussing an hypothesis
Flag1: The student has requested help for forming an hypothesis
Flag2: No summary of evidence should be offered for this goal
Flag3: A prompt should not be given before starting discussion of this goal

Step1: T-RULE5.01

If: 1) There are rules having a bearing on this goal that have succeeded and have not been discussed, and
2) The set of the rules having a bearing on the goal currently being discussed that succeeded has one member
Then: No summary of evidence should be offered for this goal

Step2: Apply the first rule that is appropriate:

T-RULE5.02

If: 1) There are rules having a bearing on this goal that have succeeded and have not been discussed,
2) The set of rules having a bearing on this goal that have succeeded and have not been discussed has one member, and
3) There is strong evidence that the student has applied the rule
Then: Affirm the conclusions made by the rule by simply stating the key factors and values to be concluded

T-RULE5.03

If: You have examined rules having a bearing on this goal that have succeeded and have not been discussed, and have found a the rule under consideration for which there is not strong evidence that the student has applied the rule under consideration
Then: Substep i. If: 1) A prompt should be given before starting discussion of this goal, and
2) The student has not requested help for forming an hypothesis
Then: Say: hyp-ready
Substep ii. Discuss the student's hypothesis [Proc014]
-> indicate whether The student has requested help for forming an hypothesis

```
Step3: T-RULE5.04
       ----------
       If: 1) No summary of evidence should be offered for this goal, or
           2) There are not rules having a bearing on this goal that have
           succeeded
       Then: Record that discussion of the goal currently being discussed is
             complete
       Else: Discuss the final value of the goal currently being discussed
             and wrap-up the discussion before returning to previous topics
             [Proc006]
              -> A remark should be made that discussion of this goal is now
             complete


CONSULT.PROC003
---------------

Purpose:  Discuss the current case

Step1: Initialize the student profile
Step2: ***missing*** case-selection
Step3: Read in the appropriate files for this case
Step4: Print out initial data, organized by objects in the domain
Step5: Record how the expert used this introductory information
Step6: T-RULE3.01
       ----------
       If: Discourse-mode is goal-directed
       Then: Substep i. Say: goal-directed-intro
             Substep ii. Discuss the goal with the student in a goal-directed
                         mode [Proc001]
       Else: Substep i. Say: freeform-intro
             Substep ii. Discuss the goal of the consultation in a free-form
                         mode [Proc002]

Step7: ***missing*** printcritique
Step8: ***missing*** eval.userther
Step9: ***missing*** special-remarks
Step10: ***missing*** laterstudy-info
Step11: ***missing*** follow-upcases
Step12: Save the student model and discussion record

DEEPERSG.PROC019
----------------

Purpose:  Present the path from the goal currently being discussed to the
          factor, and discuss the deepest subgoal for which the factor is
          relevant

Step1: T-RULE19.01
       -----------
       If: The expert has reached a final conclusion about the new subgoal
       Then: Substep i. Say: finish-deeper-sg
             Substep ii. Summarize the reasoning path from the factor to the
                         goal currently being discussed
             Substep iii. Discuss final conclusion for the new subgoal by
                          giving the final answer or discussing an hypothesis
```

```
                    [Proc005]
          Substep iv. Say that the dialogue is returning to discussion of
                      the goal currently being discussed
    Else: Substep i. Say: discuss-deeper-sg
          Substep ii. Summarize the reasoning path from the factor to the
                      goal currently being discussed
          Substep iii. Discuss the goal with the student in a goal-
                      directed mode [Proc001]
          Substep iv. Say that the dialogue is returning to discussion of
                      the goal currently being discussed


ENDOFRULE.PROC017
-----------------

Purpose:  Wrap up the discussion of the rule currently under consideration

Step1: T-RULE17.01
       -----------
       If: The rule currently under consideration is the goal rule of the
           task
       Then: Substep i. Say: give-goalaction
             Substep ii. Record that the rule currently under consideration
                         has been discussed
             Substep iii. Say: end-rule
             Substep iv. Print a design of a short line
             Substep v. Exit this procedure

Step2: T-RULE17.02
       -----------
       If: It is not true that the student model is strong for the rule
           currently under consideration
       Then: Substep i. Say: mentionendof-rule
             Substep ii. Record that the rule currently under consideration
                         has been discussed
       Else: Generate a question about the goal currently being discussed,
             using the "facts of the rule's premise" format in the premise
             part and the "correct conclusion" format in the conclusion

Step3: T-RULE17.03
       -----------
       If: This is a self-referencing rule
       Then: Say: sref-conclusion

Step4: T-RULE17.04
       -----------
       If: This is not the first rule discussed for the current topic
       Then: Say: cum-evidence

Step5: T-RULE17.05
       -----------
       If: The conclusion made by this rule is insignificant
       Then: Say: insig-evid

Step6: Apply the first rule that is appropriate:
```

T-RULE17.06

If: The expert has not applied all relevant rules for the goal
currently being discussed
Then: Say: more-rules

T-RULE17.07

If: Discussion of the goal currently being discussed is not complete
Then: Say: not-disc

Step7: Discuss rules that are related to the rule which was just discussed
[Proc011]
Step8: Say: end-rule
Step9: Print a design of a short line

FINISHGOAL.PROC006

Purpose: Discuss the final value of the goal currently being discussed and
wrap-up the discussion before returning to previous topics
Flag1: A summary should be given of this evidence
Flag2: A remark should be made that discussion of this goal is now complete

Step1: Say: printval
Step2: T-RULE6.01

If: 1) There are rules that the expert used to determine this goal,
and
2) A: A summary should be given of this evidence, or
B: Ask: want-final-summary
Then: Present the conclusion and a summary of evidence that supports
it

Step3: ***missing*** mention-doneparm-topics
Step4: T-RULE6.02

If: There are rules that the expert used to determine this goal
Then: Print a design of asterisks

Step5: Record that discussion of the goal currently being discussed is
complete
Step6: T-RULE6.03

If: A remark should be made that discussion of this goal is now
complete
Then: Say: done-topic

FREEGOAL.PROC002

Purpose: Discuss the goal currently being discussed in a free-form mode

```
Step1: If: The student has requested to be told when he appears to be
           changing the subject
       Then: ***missing*** announce-goal-for-student
Step2: If: The expert has reached a final conclusion about the goal currently
           being discussed
       Then: Exit this procedure
Step3: Accept a question, hypothesis, or special option from the student
Step4: If: The student's current input is a hypothesis
       Then: Substep i. If: 1) The hypothesis topic is not a subgoal of the
                            goal currently being discussed, and
                            2) The student has requested to be told when he
                            appears to be changing the subject
                         Then: ***missing*** irrelevant-hyp
             Substep ii. Discuss the student's hypothesis [Proc014]
       Else: Substep i. Tell the student the reported value for the factor
                        the student has inquired about
             Substep ii. If: The factor the student has inquired about is not
                             a subgoal of (or the same as) the goal currently
                             being discussed
                         Then: ***missing*** irrelevant-ques
Step5: Go to step2

GIVENPARMDATA.PROC012
---------------------

Purpose:  Supply requested data to the student, with introduction and related
          information, as appropriate
Flag1: The request for data has already been determined to be appropriate
Flag2: The student has requested this data explicitly

Step1: T-RULE12.01
       -----------
       If: 1) The student has not requested this data explicitly, and
           2) The factor has not been discussed before
       Then: Substep i. If: This factor should be asked about before an
                            attempt is made to deduce it from rules
                         Then: Say: this is a 'labdata' parameter
                         Else: Say: this is a parameter that should be asked
                               about when it can't be deduced from other
                               evidence
             Substep ii. Give a fixed-text definition for the factor
             Substep iii. Record that the factor has been mentioned in the
                          dialogue
       Else: If: The student has not requested this data explicitly
             Then: Say: case-data

Step2: Apply the first rule that is appropriate:

       T-RULE12.02
       -----------
       If: The request for data has already been determined to be appropriate
       Then: The question is valid
```

T-RULE12.03

If: It is not appropriate now to request the reported value for the factor
Then: Substep i. Inform the student that it is not necessary to request the reported value for the factor in this case. [Rulepk032]
Substep ii. Exit this procedure

T-RULE12.04

If: The factor is the goal currently being discussed
Then: The question is valid

T-RULE12.05

If: 1) The factor is not a subgoal of (or the same as) the goal currently being discussed, and
2) The factor is not relevant to a higher subgoal
Then: Say: irrelevant-question

T-RULE12.06

If: 1) The factor is relevant to a higher subgoal, and
2) The factor is not a subgoal of the goal currently being discussed
Then: Say: this factor is relevant to a previous goal, but not the current topic

T-RULE12.07

If: 1) There are not related factors that form a block of data, and
2) There are not rules that mention the factor that need this information in order to be applied
Then: Substep a. Say: rules-evaled-already
Substep b. If: The student wants details
Then: Describe how the factor is relevant to the goal currently being discussed (the USE option)
Substep c. Exit this procedure
Else: The question is valid

Step3: Tell the student the reported value for the factor
Step4: If this factor is screened, mark the screening parameter as known; otherwise, if this question is a screening parameter, give the value of the screened factor
Step5: T-RULE12.08

If: There are related factors that form a block of data
Then: Present the related factors that form a block of data [Proc034]

GOAL.PROC001

Purpose: Discuss the goal with the student in a goal-directed mode
Flag1: A prompt should not be given before starting discussion of this goal

Step1: T-RULE1.01

If: There are no rules for concluding about this goal which apply to this case
Then: Substep i. If: It is appropriate now to request the reported value for the goal currently being discussed
Then: Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate
Substep ii. Say: no-rules-apply
Substep iii. Record that discussion of the goal currently being discussed is complete
Substep iv. Exit this procedure

Step2: T-RULE1.02

If: The expert deduced the goal currently being discussed from a definition
Then: Substep i. Present the definition rule [Rulepk042]
Substep ii. Discuss rules that are related to the rule which was just discussed [Proc011]
Substep iii. Discuss the rules having a bearing on the goal currently being discussed that have not yet been discussed [Proc007]
Substep iv. Give the values of the related factors
Substep v. Record that discussion of the goal currently being discussed is complete
Substep vi. Exit this procedure

Step3: T-RULE1.03

If: All of the rules for determining the goal currently being discussed have failed
Then: Substep i. Discuss the fact that all rules fail for the goal currently being discussed and consider whether it is appropriate to request the reported value [Proc035]
Substep ii. Record that discussion of the goal currently being discussed is complete
Substep iii. Exit this procedure

Step4: T-RULE1.04

If: 1) The goal currently being discussed is the goal of the consultation, and
2) A: The student's estimation of his sophistication (on a scale from 0 to 4) is less than 3, or
B: The goal of the consultation has not been discussed before
Then: Substep i. Say: desc-goalrule
Substep ii. Show a sketch of the subgoal tree for the goal currently being discussed
Substep iii. Say: tell-optionname

```
            Substep iv. Discuss the goal rule of the task clause-by-clause
                        [Proc004]
            Substep v. Say: goodbye
            Substep vi. Record that the goal of the consultation has been
                        mentioned in the dialogue
            Substep vii. Exit this procedure

Step5: For each of the antecedent rules that have made conclusions about the
       goal currently being discussed
       Do: T-RULE1.05
           ----------
           If: The student model is negative for the rule under consideration
           Then: Substep i. Select a question type suitable for the rule
                            under consideration [Rulepk024]
                 Substep ii. Generate a question based upon the rule under
                             consideration and evaluate the student's
                             response
                 Substep iii. Record that the rule under consideration has
                              been discussed
                 Substep iv. Discuss rules that are related to the rule which
                             was just discussed [Proc011]

Step6: T-RULE1.06
       ----------
       If: It is appropriate now to request the reported value for the goal
           currently being discussed
       Then: Supply requested data to the student, with introduction and
             related information, as appropriate [Proc012]
              -> The request for data has already been determined to be
             appropriate

Step7: T-RULE1.07
       ----------
       If: 1) The student profile indicates that rule conclusions should be
           mentioned, and
           2) There are rules that have just been applied by the expert which
           pertain to the goal currently being discussed
       Then: Substep i. Say: completedrules
             Substep ii. For each of the rules.to.mention
                         Do: Present the conclusion and a summary of evidence
                             that supports it

Step8: T-RULE1.08
       ----------
       If: Discussion of the goal currently being discussed is complete
       Then: Exit this procedure

Step9: T-RULE1.09
       ----------
       If: The expert has reached a final conclusion about the goal currently
           being discussed
       Then: Substep i. Discuss final conclusion for the goal currently being
                        discussed by giving the final answer or discussing an
                        hypothesis [Proc005]
             Substep ii. Exit this procedure
```

```
Step10:
        T-RULE1.10
        ----------
        If: 1) The goal currently being discussed is the goal of the
            consultation, and
            2) The expert has applied all relevant rules for the goal of the
            consultation
        Then: Substep i. Wrap up the discussion of the goal rule of the task
                         [Proc017]
              Substep ii. Record that discussion of the goal currently being
                          discussed is complete
              Substep iii. Exit this procedure

Step11:
        T-RULE1.11
        ----------
        If: 1) You have not asked the student if he would like to end
            discussion of the current topic,
            2) Discussion of the goal currently being discussed is essentially
            complete, and
            3) Ask: s-wants-hyp
        Then: Substep i. Discuss the student's hypothesis [Proc014]
              Substep ii. If: Discussion of the goal currently being discussed
                              is complete
                          Then: Substep a. Discuss the final value of the goal
                                           currently being discussed and wrap-
                                           up the discussion before returning
                                           to previous topics [Proc006]
                                            -> A remark should be made that
                                           discussion of this goal is now
                                           complete
                                Substep b. Exit this procedure

Step12:
        T-RULE1.12
        ----------
        If: This is the first time through this discussion loop
        Then: Substep i. Provide initial orientation for determining the goal
                         currently being discussed [Proc036]
                          -> indicate whether A prompt should not be given
                         before starting discussion of this goal
              Substep ii. Define this is the first time through this
                          discussion loop to be nothing

Step13:
        T-RULE1.13
        ----------
        If: There are rules for the goal currently being discussed that have
            recently failed
        Then: Discuss interesting rules which are known to fail for the goal
              currently being discussed, based on data now available to the
              student [Proc010]
```

```
Step14:
       T-RULE1.14
       ----------

       If: There are rules for the goal currently being discussed that have
           recently succeeded
       Then: Discuss interesting rules which are known to succeed for the
             goal currently being discussed, based on data now available to
             the student [Proc043]

Step15: Accept a question, hypothesis, or special option from the student
Step16: Process the student's input as follows:
       HYPOTHESIS: Substep i. T-RULE1.15
                             ----------
                             If: The hypothesis topic is not a subgoal of
                                 the goal currently being discussed
                             Then: Say: irrelevant-hyp

                   Substep ii. Discuss the student's hypothesis [Proc014]
                   Substep iii. T-RULE1.16
                                ----------
                                If: The expert has reached a final conclusion
                                    about the hypothesis topic
                                Then: Substep a. Discuss the final value of
                                                 the hypothesis topic and
                                                 wrap-up the discussion
                                                 before returning to previous
                                                 topics [Proc006]
                                                  -> A remark should be made
                                                 that discussion of this goal
                                                 is now complete
                                      Substep b. If: The hypothesis topic is
                                                     the goal currently being
                                                     discussed
                                                 Then: Exit this procedure
                                                 Else: Say that the dialogue
                                                       is returning to
                                                       discussion of the goal
                                                       currently being
                                                       discussed

                   Substep iv. T-RULE1.17
                               ----------
                               If: 1) The expert has not reached a final
                                   conclusion about the hypothesis topic, and
                                   2) The hypothesis topic is not the goal
                                   currently being discussed
                               Then: Substep a. Discuss the goal with the
                                                student in a goal-directed
                                                mode [Proc001]
                                     Substep b. Say that the dialogue is
                                                returning to discussion of
                                                the goal currently being
                                                discussed

       DATA: Substep i. Supply requested data to the student, with
                        introduction and related information, as appropriate
                        [Proc012]
                         -> The student has requested this data explicitly
             Substep ii. If: The question is not valid
```

```
                    Then: Go to step15
          Substep iii. If: The factor the student has inquired about is
                           not a factor of the goal currently being
                           discussed
                       Then: Present the path from the goal currently
                             being discussed to the factor the student has
                             inquired about, and discuss the deepest
                             subgoal for which the factor the student has
                             inquired about is relevant [Proc819]
OTHERWISE: Substep i. Invoke the special option procedure for the
                      student's input
           Substep ii. If: Nodatagivenflg
                       Then: Go to step15
Step17: Go to step6

HELPFORGOAL.PROC816
-------------------

Purpose:  Provide help for determining the goal currently being discussed

Step1: T-RULE16.01
       -----------
       If: 1) It is not true that what has been mentioned in the recent
           context of the interaction, and
           2) The student has not seen the factors for this goal
       Then: Substep i. List the factors that occur in the rules that have a
                        bearing on this goal
             Substep ii. Say: mention-optionname
             Substep iii. Exit this procedure

Step2: T-RULE16.02
       -----------
       If: 1) There are not successful rules for determining this goal that
           have not been discussed, or
           2) There are self-referencing rules that are waiting for other
           rules to be applied
       Then: For each of the the rules having a bearing on the goal currently
             being discussed that failed
             Do: If: The rule has not fired
                 Then: Present the rule under consideration [Rulepk823]
                       -> Some remark should be made, even if no
                       presentation is possible

Step3: T-RULE16.03
       -----------
       If: There are not successful rules for determining this goal that have
           not been discussed
       Then: Exit this procedure

Step4: To determine the rules under contention, compute the following:
       Select the rules that might be presented to the student from among
       successful rules for determining this goal that have not been
       discussed [Rulepk837]
Step5: To determine the rule being considered, compute the following:  for
       each of the the rules under contention, determine how "interesting" it
       is to present the rule at this time, returning the element with the
```

highest score
Step6: Present the rule being considered [Rulepk031]
Step7: Modify the student knowledge model for the rule being considered by -400

HELPFORHYP.PROC018

Purpose: Select a rule and present it to the student as a form of help for reaching an hypothesis

Step1: To determine the rules under contention, compute the following: Select the rules that might be presented to the student from among rules having a bearing on this goal that have succeeded and have not been discussed [Rulepk037]
Step2: To determine the rule being considered, compute the following: for each of the the rules under contention, determine how "interesting" it is to present the rule at this time, returning the element with the highest score
Step3: Suggest a rule to the student in the form of help for reaching an hypothesis [Rulepk030]

HYPOTHESIS.PROC014

Purpose: Discuss the student's hypothesis
Flag1: The student has requested help for forming an hypothesis

Step1: T-RULE14.01

If: It is appropriate now to request the reported value for the hypothesis topic
Then: Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate

Step2: Apply the first rule that is appropriate:

T-RULE14.02

If: There are values mentioned by the student when he asked to make an hypothesis
Then: Define values hypothesized by the student to be values mentioned by the student when he asked to make an hypothesis

T-RULE14.03

If: The student has not requested help for forming an hypothesis
Then: Ask the student for his hypothesis for the value of the hypothesis topic

```
Step3: T-RULE14.04
       -----------

       If: 1) The student has requested help for forming an hypothesis, and
           2) There are rules having a bearing on this goal that have
           succeeded and have not been discussed
       Then: Substep i. Select a rule and present it to the student as a form
                        of help for reaching an hypothesis [Proc018]
             Substep ii. If: There are rules having a bearing on this goal
                             that have succeeded and have not been discussed
                         Then: Substep a. Say: hyp-ready
                               Substep b. Go to step2
                         Else: Exit this procedure
       Else: If: 1) The student has requested help for forming an hypothesis,
                 and
                 2) It is not true that Ask: no-help
             Then: Exit this procedure

Step4: Compare the student's hypothesis to the expert's current hypothesis
       for this goal
Step5: Mark the rules used by the expert to reflect their consistency with
       the student's hypothesis
Step6: For each of the rules applied by the expert
       Do: Apply modelling t-rules to determine the belief that the rule
           under consideration was considered by the student [Rulepk028]
Step7: For each of the rules applied by the expert
       Do: In view of the model for all other rules, apply  self-referencing
           t-rules to determine whether the rule under consideration was
           considered by the student [Rulepk029]
Step8: T-RULE14.05
       -----------
       If: There are correct hypothesized values for which there is strong
           evidence that the student considered the appropriate rules
       Then: Say: right-hyp

       Comment: Strategy for strong right val is to say that S is correct
                and state the evidence.

Step9: Apply the first rule that is appropriate:

       T-RULE14.06
       -----------
       If: The set of domain rules that the tutor strongly believes were used
           by the student to make his hypothesis has one member
       Then: Affirm the conclusions made by the rule by simply stating the
             key factors and values to be concluded


       T-RULE14.07
       -----------
       If: There are domain rules that the tutor strongly believes were used
           by the student to make his hypothesis
       Then: Substep i. Say: strongrules-intrc
             Substep ii. Present a summary of evidence that the hypothesis
                         topic is correct hypothesized values for which there
                         is strong evidence that the student considered the
                         appropriate rules
```

Step10:

T-RULE14.08

If: There are correct hypothesized values for which the student may have missed some relevant evidence
Then: For each of the correct hypothesized values for which the student may have missed some relevant evidence
Do: Respond to the hypothesis that the hypothesis topic is this . value [Rulepk025]

Comment: If there are several strong rules and 1 weak one, rule this rulepk will state the strong evidence and quiz about the weak rule (mixed right val strategy).

Step11:

T-RULE14.09

If: There are correct values for which there is both positive and negative evidence
Then: Generate a question about relevant rules, using "fill-in of known factors" format in the premise part and the "correct conclusion" format in the conclusion

Comment: Strategy for ambig right val is to see if he knows evidence contrary to his hypothesis.

Step12:

T-RULE14.1

If: There are wrong hypothesized values
Then: For each of the wrong hypothesized values
Do: Respond to the hypothesis that the hypothesis topic is this value [Rulepk025]

Comment: Rulepks 25 and 26 will present the unknown evidence

Step13:

T-RULE14.11

If: There are values missing from the student's hypothesis
Then: Generate a question about relevant rules, using "fill-in of known fact rs" format in the premise part and the "correct conclusion" format in the conclusion

Step14:

T-RULE14.12

If: There are values that the expert explicitly rejected
Then: Substep i. Say: intro-rejections
Substep ii. For each of the values that the expert explicitly rejected
Do: Present the conclusion that the hypothesis topic is (not) this value [Rulepk026]

Comment: Strategy for rejected vals is to quiz about the evidence that the expert is considering

```
Step15:
        T-RULE14.13
        -----------
        If: There are hypothesized values for which there is only
            insignificant evidence
        Then: For each of the hypothesized values for which there is only
              insignificant evidence
              Do: Substep i. Ask: request-certainty
                  Substep ii. Say: cf-reply
                  Substep iii. Say: insig-concls
                  Substep iv. Present a summary of evidence that the
                              hypothesis topic is this value

Step16:
        T-RULE14.14
        -----------
        If: 1) There are hypothesized values for which there is no evidence at
            all, and
            2) The hypothesis topic is not numerically valued
        Then: Generate a question about the hypothesis topic, using
              invalidfactors format in the premise part and the "correct
              conclusion" format in the conclusion

Step17:
        T-RULE14.15
        -----------
        If: There are insignificant values that the student did not mention
        Then: ***missing*** mention-insigvals

Step18:
        T-RULE14.16
        -----------
        If: The expert has not reached a final conclusion about the hypothesis
            topic
        Then: Offer a summary of what is known so far and the opportunity to
              terminate discussion of the hypothesis topic [Proc039]
        Else: Record that discussion of the hypothesis topic is complete


NEVERASKEDPARM.PROC015
----------------------

Purpose:  Inform the student that the expert never needed to ask about the
          factor and offer details

Step1: Say: the expert never needed to inquire about this factor
Step2: T-RULE15.01
        -----------
        If: There are rules having a bearing on this goal that have succeeded
        Then: Say: rules-tellus

Step3: T-RULE15.02
        -----------
        If: 1) The factor is not the goal currently being discussed,
            2) There are rules having a bearing on this goal that have
            succeeded,
            3) There are rules having a bearing on this goal that have failed,
```

```
        and
        4) Ask: wants-details
    Then: Substep i. Discuss the goal with the student in a goal-directed
                     mode [Proc001]
          Substep ii. Say that the dialogue is returning to discussion of
                      the goal currently being discussed

Step4: T-RULE15.03
       -----------
       If: The factor is not the goal currently being discussed
       Then: Describe how the factor is relevant to the goal currently being
             discussed (the USE option)

       Comment: In theory, this will print a message about relevance and exit


NOTDISCUBRULES.PROC007
----------------------

Purpose:  Discuss the rules having a bearing on the goal currently being
          discussed that have not yet been discussed

Step1: For each of the the rules having a bearing on the goal currently being
       discussed that succeeded
       Do: T-RULE7.01
           ----------
           If: 1) The rule has not fired,
               2) The rule has not been discussed, and
               3) The rule under consideration is on the lesson plan
           Then: Substep i. Make a ">>>" design with:  Considering rules that
                            haven't been discussed
                 Substep ii. Discuss the rule under consideration clause-by-
                             clause [Proc004]

Step2: For each of the the rules having a bearing on the goal currently being
       discussed that failed
       Do: T-RULE7.02
           ----------
           If: 1) The rule has not been discussed, and
               2) A: The rule has not fired, or
                  B: The rule under consideration is on the lesson plan
           Then: Substep i. Make a ">>>" design with:  Considering rules that
                            haven't been discussed
                 Substep ii. Present the rule under consideration [Rulepk023]

Step3: T-RULE7.03
       ----------
       If: There are rules not used by the expert to determine this goal
       Then: ***missing*** unusedrules

Step4: End the ">>>" design with:  End of discussion of rules for this factor
```

ORIENTATION.PROC036

Purpose: Provide initial orientation for determining the goal currently being discussed
Flag1: A prompt should not be given before starting discussion of this goal

Step1: T-RULE36.01
If: A prompt should be given before starting discussion of this goal
Then: Say: announce-goal-for-student

Step2: T-RULE36.02
If: 1) The number of rules that conclude about the current goal is greater than 5, and
2) The goal currently being discussed has not been discussed before
Then: Substep i. Give a fixed-text definition for the goal currently being discussed
Substep ii. Provide orientation for finding out about the goal currently being discussed by listing the subgoals most commonly pursued in this context
Substep iii. Say: orient-by-rule
Substep iv. List (typical) values for the goal currently being discussed
Substep v. Record that the goal currently being discussed has been mentioned in the dialogue

Step3: T-RULE36.03
If: Values hypothesized for this goal which have already been confirmed in discussion
Then: Say: discussed-concl

PLEASEDISCUSS.PROC022

Purpose: Change the dialogue from the goal currently being discussed to the new topic, as the student has requested

Step1: T-RULE22.01
If: Discussion of the new topic is complete
Then: Substep i. Say: donetopic
Substep ii. Discuss the final value of the new topic and wrap-up the discussion before returning to previous topics [Proc006]
Substep iii. Say that the dialogue is returning to discussion of the goal currently being discussed
Substep iv. Exit this procedure

Step2: T-RULE22.02
If: The new topic is not a subgoal of (or the same as) the goal currently being discussed
Then: Say: will-proceed

Step3: Discuss the goal with the student in a goal-directed mode [Proc001]
Step4: Say that the dialogue is returning to discussion of the goal currently being discussed

QUIZRULE.PROC021

Purpose: Quiz the student about a rule for determining the goal currently being discussed
Flag1: The student has requested a hint

Step1: T-RULE21.01

If: The student has requested a hint
Then: To determine the rules under contention, compute the following: the rules that succeed for the goal currently being discussed that have not been discussed
Else: To determine the rules under contention, compute the following: the subset of rules which are relevant for the goal currently being discussed and can now be applied, but have not been discussed for which the rule under consideration is not a rule that failed inappropriately for quizzing (i.e., it has factors whose values are unknown to the expert or has a tabular factor with a null value)

Step2: T-RULE21.02

If: There are not the rules under contention
Then: Substep i. Say: no-quiz
Substep ii. Exit this procedure

Step3: To determine the rule being considered, compute the following: for each of the the rules under contention, determine how "interesting" it is to present the rule at this time, returning the element with the highest score
Step4: T-RULE21.03

If: 1) The student has requested a hint, and
2) A hint value can be suggested for this rule, based on its effect on the current hypothesis
Then: Substep i. Say: valu-hint
Substep ii. Define the rule hinted about to be the rule being considered
Substep iii. Define what has been mentioned in the recent context of the interaction to be a hint about a rule
Substep iv. Exit this procedure

Step5: T-RULE21.04

If: The student has requested a hint
Then: Substep i. Say: no-hint
Substep ii. Go to step1

Step6: Select a question type suitable for the rule being considered [Rulepk024]

Step7: Generate a question based upon the rule being considered and evaluate the student's response
Step8: Record that the rule being considered has been discussed

RECENTFAILS.PROC010

Purpose: Discuss interesting rules which are known to fail for the goal currently being discussed, based on data now available to the student

Step1: To determine the rules under contention, compute the following: the subset of recentfails for which
1) The rule has not been discussed,
2) The rule under consideration is not a rule that failed inappropriately for quizzing (i.e., it has factors whose values are unknown to the expert or has a tabular factor with a null value), and
3) The rule under consideration is on the lesson plan
Step2: If: There are not the rules under contention
Then: Exit this procedure
Step3: To determine the rule being considered, compute the following: for each of the the rules under contention, determine how "interesting" it is to present the rule at this time, returning the element with the highest score
Step4: T-RULE10.01

If: The number of subgoals remaining to be determined before this rule can be applied is not zero
Then: Exit this procedure

Step5: T-RULE10.02

If: The number of factors appearing in this rule which need to be asked by the student is not zero
Then: Substep i. Say: intro-data
Substep ii. For each of the factors whose reported value must be given to the student before he can apply the rule under consideration
Do: Tell the student the reported value for the factor

Step6: Select a question type suitable for the rule being considered [Rulepk024]
Step7: Generate a question based upon the rule being considered and evaluate the student's response
Step8: Record that the rule being considered has been discussed

RECENTSUCC.PROC043

Purpose: Discuss interesting rules which are known to succeed for the goal currently being discussed, based on data now available to the student

Step1: For each of the rules that have recently succeeded

```
Do: Apply the first rule that is appropriate:

    T-RULE43.01
    -----------
    If: 1) The rule has not been discussed,
        2) There is not strong evidence that the student has applied
        the rule under consideration, and
        3) There is a subgoal that the student probably does not know
    Then: Substep i. Print a design of a line of dashes
          Substep ii. Say: intro-recentsucc-tree
          Substep iii. For each of the rules in this subtree that the
                       student probably cannot apply because  he is
                       unable to determine one or more subgoals
                       Do: Present missing rules to the student so
                           that he can determine the missing subgoal,
                           then quiz about rules that use these
                           subgoals [Proc046]
          Substep iv. Print a design of a line of dashes


    T-RULE43.02
    -----------
    If: 1) The rule has not been discussed,
        2) There is not a subgoal that the student probably does not
        know,
        3) There is not strong evidence that the student has applied
        the rule under consideration, and
        4) The rule under consideration is the only rule for the goal
        currently being discussed that mentions certain case data
    Then: Substep i. Select a question type suitable for the rule
                     under consideration [Rulepk024]
          Substep ii. Generate a question based upon the rule under
                      consideration and evaluate the student's
                      response
          Substep iii. Record that the rule under consideration has
                       been discussed

RELATEDRULES.PROC011
--------------------

Purpose:  Discuss rules that are related to the rule which was just discussed

Step1: T-RULE11.01
       -----------
       If: The goal currently being discussed is a true/false parameter
       Then: Exit this procedure

       Comment: Restrict related rule quizzes to parms with real values so
                the relation is clear

Step2: For each of the randomly reordered set of rules which are relevant for
       the goal currently being discussed and can now be applied, but have
       not been discussed
```

```
        Do: T-RULE11.02
            ----------
            If: 1) The number of questions asked so far is less than  3,
                2) It is not true that the premise of the rule under
                consideration includes an iteration,
                3) The rule under consideration is not a rule that failed
                inappropriately for quizzing (i.e., it has factors whose
                values are unknown to the expert or has a tabular factor with
                a null value), and
                4) The rule under consideration does conclude about some of
                the same values as the rule just discussed
            Then: Substep i. Make a ">>>" design with:  Considering related
                             rules
                  Substep ii. Select a question type suitable for the rule
                              under consideration [Rulepk024]
                  Substep iii. Generate a question based upon the rule under
                               consideration and evaluate the student's
                               response
                  Substep iv. Record that the rule under consideration has
                              been discussed

Step3: End the ">>>" design with:  End of our discussion of related rules

REMAININGTASK.PROC047
---------------------

Purpose:  Offer an appraisal of what more needs to be done for this goal,
          suggesting that the topic be terminated, if appropriate.

Step1: Compare the student's hypothesis to the expert's current hypothesis
       for this goal
Step2: T-RULE47.01
       -----------
       If: There are values missing from the student's hypothesis
       Then: Say: missingfinalvals
       Else: Say: hyparmrulesnotdone

Step3: T-RULE47.02
       -----------
       If: 1) There are not values missing from the student's hypothesis, and
           2) Discussion of the hypothesis topic is essentially complete
       Then: Say: goal-nearly-done

Step4: T-RULE47.03
       -----------
       If: Ask: want-shortcut
       Then: Substep a. If: 1) The expert did find it necessary to inquire
                            about the hypothesis topic, and
                            2) The student has not requested the reported
                            value already
                       Then: Supply requested data to the student, with
                             introduction and related information, as
                             appropriate [Proc012]
                              -> The request for data has already been
                             determined to be appropriate
             Substep b. Present all relevant case data for this goal (the
                        ALLDATA option)
```

```
            Substep c. Record that discussion of the hypothesis topic is
                       complete
       Else: If: 1) There is a curval, and
                 2) Ask: s-wants-hyp
             Then: Present the conclusion and a summary of evidence that
                   supports it


RULE.PROC004
------------

Purpose:  Discuss the rule currently under consideration clause-by-clause

Step1: Print a design of a short line
Step2: Say: announce-rule
Step3: T-RULE4.01
       ----------
       If: 1) The rule currently under consideration is not the goal rule of
           the task,
           2) It is not true that the premise of the rule currently under
           consideration includes an iteration,
           3) The student model is uncertain for the rule currently under
           consideration, and
           4) The rule currently under consideration is the only rule that
           concludes this value for the goal currently being discussed
       Then: Generate a question about the goal currently being discussed,
             using newsubgoals format in the premise part and the "correct
             conclusion" format in the conclusion

Step4: If: There is a factor that remains to be considered before this rule
           can be applied
       Then: Substep i. If: A new clause in the rule will now be considered
                        Then: Substep a. If: 1) The number of factors in this
                                             clause is equal to 1, and
                                             2) The factor under
                                             consideration is the goal
                                             currently being discussed
                                      Then: Go to step4
                              Substep b. If: All factors in this clause have
                                             not been discussed
                                      Then: Say: mustdoclause
                                      Else: Substep i. Say: alldoneclause
                                            Substep ii. Go to step4
             Substep ii. If: 1) There are no rules for concluding about this
                             goal which apply to this case, or
                             2) Discussion of the factor under consideration
                             is not complete
                         Then: Substep i. If: It is not true that a new
                                              clause in the rule will now be
                                              considered
                                       Then: Say: mustdo-factor
                               Substep ii. Discuss the goal with the student
                                           in a goal-directed mode [Proc001]
                               Substep iii. Say: rule-return
             Substep iii. Go to step4
Step5: Wrap up the discussion of the rule currently under consideration

       [Proc017]
```

SPECIFICHELP.PROC020

Purpose: Provide assistance for using the subtopic mentioned in the request for help to determine the goal currently being discussed

Step1: T-RULE20.01

If: It is appropriate now to request the reported value for the subtopic mentioned in the request for help
Then: Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate

Step2: T-RULE20.02

If: The subtopic mentioned in the request for help is a subgoal of (or the same as) the goal currently being discussed
Then: Describe how the subtopic mentioned in the request for help is relevant to the goal currently being discussed (the USE option)
Else: Say: irrelevant-ques

Step3: T-RULE20.03

If: 1) The expert has reached a final conclusion about the subtopic mentioned in the request for help, and
2) Discussion of the subtopic mentioned in the request for help is complete
Then: Exit this procedure

Step4: T-RULE20.04

If: 1) The expert has not reached a final conclusion about the subtopic mentioned in the request for help, and
2) Discussion of the subtopic mentioned in the request for help is not complete
Then: Substep i. Say: discuss-help-topic
Substep ii. Discuss the goal with the student in a goal-directed mode [Proc001]

Step5: T-RULE20.05

If: Discussion of the subtopic mentioned in the request for help is not complete
Then: Substep i. Say: intro-hyp
Substep ii. Discuss final conclusion for the subtopic mentioned in the request for help by giving the final answer or discussing an hypothesis [Proc005]
-> The student has requested help for forming an hypothesis

Step6: Say that the dialogue is returning to discussion of the goal currently being discussed

TELLMEOPTION.PROC045

Purpose: Terminate discussion of this goal, as the student has requested
Flag1: A summary should be given of this evidence

Step1: T-RULE45.01

If: Discussion of this goal is complete
Then: Substep i. Say: give-val
Substep ii. Exit this procedure

Step2: Apply the first rule that is appropriate:

T-RULE45.02

If: All of the rules for determining this goal have failed
Then: Discuss the fact that all rules fail for this goal and consider whether it is appropriate to request the reported value [Proc035]

Apply all rules that are appropriate:

T-RULE45.03

If: 1) The expert did find it necessary to inquire about this goal, and
2) The student has not requested the reported value already
Then: Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate

T-RULE45.04

If: There are rules that the expert used to determine this goal
Then: Discuss the final value of this goal and wrap-up the discussion before returning to previous topics [Proc006]
-> indicate whether A summary should be given of this evidence

Step3: T-RULE45.05

If: 1) The goal currently being discussed is not this goal, and
2) There are rules that the expert used to determine this goal
Then: Substep i. Say: topic-done
Substep ii. Say that the dialogue is returning to discussion of the goal currently being discussed

Step4: Record that discussion of this goal is complete

UNKNOWNRULES.PROC046

Purpose: Present missing rules to the student so that he can determine the missing subgoal, then quiz about rules that use these subgoals

Step1: Say: intro-newvals
Step2: Affirm the conclusions made by rules probably not known by the student by simply stating the key factors and values to be concluded
Step3: Discuss final conclusion for the subgoal that these rules conclude about by giving the final answer or discussing an hypothesis [Proc085]
-> No summary of evidence should be offered for this goal
Step4: T-RULE46.01

If: There are other unknown subgoals that are used in conjunction with this subgoal that the student does not know
Then: Exit this procedure

Step5: Select a question type suitable for the rule that uses this subgoal [Rulepk024]
Step6: Generate a question based upon the rule that uses this subgoal and evaluate the student's response
Step7: Record that the rule that uses this subgoal has been discussed

BACKGROUND.RULEPK040

Purpose: Determine the a priori belief that the student knows the rule

Step1: Apply all rules that are appropriate:

T-RULE40.01

If: The sophistication level of the rule is not greater than the student's estimation of his sophistication (on a scale from 0 to 4)
Then: Modify the cumulative "background component" of the student knowledge model by if the sophistication level of the rule minus the student's estimation of his sophistication (on a scale from 0 to 4) is zero then 400, else 800

T-RULE40.02

If: 1) The sophistication level of the rule is not greater than the student's estimation of his sophistication (on a scale from 0 to 4),
2) The rule is definitional, and
3) It is not true that the rule mentions tables or lists in its premise
Then: Modify the cumulative "background component" of the student knowledge model by .9

T-RULE40.03

If: The premise of the rule includes an iteration
Then: Modify the cumulative "background component" of the student

knowledge model by -.9

Step2: Modify the student knowledge model for the rule by the cumulative "background component" of the student knowledge model
Step3: Exit this procedure, returning as a result: the cumulative "background component" of the student knowledge model

CHOSENRULEINTRO.RULEPK038

Purpose: Select an introduction for beginning discussion of the rule being considered

Step1: Apply the first rule that is appropriate:

T-RULE38.01

If: 1) What has been mentioned in the recent context of the interaction is either a deeper subgoal or a subtopic relevant to the current goal, and
2) There is a factor being focussed upon
Then: Say: focus-intro

T-RULE38.02

If: A hint value can be suggested for this rule, based on its effect on the current hypothesis
Then: Say: valu-intro

T-RULE38.03

If: True
Then: Say: default-rule-intro

DEFNRULE.RULEPK042

Purpose: Present the definition rule

Step1: Apply the first rule that is appropriate:

T-RULE42.01

If: 1) The rule has fired,
2) The student model is strong for the definition rule, or
3) The definition rule is one of self-referencing rules that are waiting for other rules to be applied
Then: Present the conclusion about the goal currently being discussed made by the definition rule [Rulepk026]

T-RULE42.02

If: 1) The number of subgoals remaining to be determined before this rule can be applied is zero,
2) This topic is *not* a true/false parameter, and
3) The definition rule is the only rule that concludes this value for the goal currently being discussed
Then: Generate a question about the goal currently being discussed, using "given new facts" format in the premise part and "fill-in" format in the conclusion

T-RULE42.03

If: The number of subgoals remaining to be determined before this rule can be applied is zero
Then: Generate a question about the goal currently being discussed, using "fill-in undiscussed facts" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE42.04

If: True
Then: Substep i. Say: intro-defn-rule
Substep ii. Discuss the definition rule clause-by-clause [Proc004]

EXPERTDIDNTASK.RULEPK033

Purpose: Explain why the expert did not ask about the factor

Step1: Apply the first rule that is appropriate:

T-RULE33.01

If: There is a definitional rule that states that this factor should not be asked for this case
Then: Substep i. Say: dontask-rule
Substep ii. Record that the definition rule has been discussed

T-RULE33.02

If: The expert deduced the factor from a definition
Then: Present the definition rule [Rulepk042]

T-RULE33.03

If: A screening question is asked before considering the factor
Then: Substep i. Define the factor the student has inquired about to be the screening parameter

```
            Substep ii. Supply requested data to the student, with
                        introduction and related information, as appropriate
                        [Proc012]
                         -> The student has requested this data explicitly

        Comment: See if expert asked the screen parm


        T-RULE33.04
        -----------
        If: True
        Then: Inform the student that the expert never needed to ask about the
              factor and offer details [Proc015]


FAILRULE.RULEPK023
------------------

Purpose:  Present the rule currently under consideration
Flag1: Some remark should be made, even if no presentation is possible

Step1: Apply the first rule that is appropriate:

        T-RULE23.01
        -----------
        If: 1) The rule has fired, and
            2) The premise of the rule currently under consideration includes
            an iteration
        Then: Generate a question about the goal currently being discussed,
              using the "facts of the rule's premise" format in the premise
              part and the "correct conclusion" format in the conclusion


        T-RULE23.02
        -----------
        If: 1) The rule has fired, and
            2) This is not a rule that failed and has factors whose values are
            unknown to the expert or a tabular factor with a null value
        Then: Substep i. Select a question type suitable for the rule
                         currently under consideration [Rulepk024]
              Substep ii. Generate a question based upon the rule currently
                          under consideration and evaluate the student's
                          response


        T-RULE23.03
        -----------
        If: 1) The number of subgoals remaining to be determined before this
            rule can be applied is zero, and
            2) A: The premise of the rule currently under consideration
               includes an iteration, or
               B: This is a rule that failed and has factors whose values are
               unknown to the expert or a tabular factor with a null value
        Then: Say: complex-fail
```

T-RULE23.04

If: 1) The number of subgoals remaining to be determined before this rule can be applied is zero,
2) The number of factors appearing in this rule which need to be asked by the student is equal to 1, and
3) A value is not known for factors whose reported value must be given to the student before he can apply the rule under consideration
Then: Substep i. Say: intro-factor
Substep ii. Tell the student the reported value for the factor
Substep iii. If: The factor is a true/false parameter
Then: Generate a question about the goal currently being discussed, using "right premise hypothesis" format in the premise part and the "correct conclusion" format in the conclusion
Else: Generate a question about the goal currently being discussed, using "multiple choice" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE23.05

If: The number of subgoals remaining to be determined before this rule can be applied is zero
Then: Substep i. Say: intro-newdata
Substep ii. For each of the factors whose reported value must be given to the student before he can apply the rule under consideration
Do: Tell the student the reported value for the factor
Substep iii. Select a question type suitable for the rule currently under consideration [Rulepk024]
Substep iv. Generate a question based upon the rule currently under consideration and evaluate the student's response

T-RULE23.06

If: There is a subgoal appearing in this rule which must be determined for one or more rules that succeed
Then: Substep i. Say: consider-newgoal
Substep ii. Discuss the goal with the student in a goal-directed mode [Proc001]
Substep iii. Say that the dialogue is returning to discussion of the goal currently being discussed

T-RULE23.07

If: 1) The number of subgoals remaining to be determined before this rule can be applied is zero,
2) The number of factors appearing in this rule which need to be

```
    asked by the student is not zero, and
    3) This is not a rule that failed and has factors whose values are
    unknown to the expert or a tabular factor with a null value
Then: Substep i. For each of the factors whose reported value must be
                 given to the student before he can apply the rule
                 under consideration
                 Do: Tell the student the reported value for the
                     factor
      Substep ii. Say: motiv-topic


T-RULE23.08
-----------
If: 1) This is not a rule that failed and has factors whose values are
    unknown to the expert or a tabular factor with a null value,
    2) The number of subgoals remaining to be determined before this
    rule can be applied is equal to 1, and
    3) The number of factors appearing in this rule which need to be
    asked by the student is zero
Then: Substep i. Say: newgoal-motiv
      Substep ii. Discuss the goal with the student in a goal-directed
                  mode [Proc881]
      Substep iii. Say: fail-reminder
      Substep iv. Say that the dialogue is returning to discussion of
                  the goal currently being discussed


T-RULE23.09
-----------
If: The rule has not fired
Then: Substep i. Force this rule to fire without discussing it
      Substep ii. If: Some remark should be made, even if no
                      presentation is possible
                  Then: Say: no-quiz-possible

Step2: Record that the rule currently under consideration has been discussed

FOCUSRULES.RULEPK037
--------------------

Purpose:  Select the rules that might be presented to the student from among
          successful rules for determining this goal that have not been
          discussed

Step1: Apply the first rule that is appropriate:

       T-RULE37.01
       -----------
       If: 1) What has been mentioned in the recent context of the
           interaction is a hint about a rule, and
           2) The rule hinted about is one of successful rules for
           determining this goal that have not been discussed
       Then: Exit this procedure, returning as a result: the rule hinted
             about
```

T-RULE37.02

If: 1) What has been mentioned in the recent context of the interaction is a suggestion to pursue a particular hypothesis, and
2) To determine the rules under contention, compute the following: the intersection of the set of rules implicitly referred to when the tutor suggested that the student pursue a particular hypothesis and the set of successful rules for determining this goal that have not been discussed
Then: Exit this procedure, returning as a result: the rules under contention

T-RULE37.03

If: 1) What has been mentioned in the recent context of the interaction is either a deeper subgoal or a subtopic relevant to the current goal, and
2) To determine the rules under contention, compute the following: those successful rules for determining this goal that have not been discussed that mention the factor being focussed upon in their premise
Then: Exit this procedure, returning as a result: the rules under contention

T-RULE37.04

If: True
Then: Substep i. Define what has been mentioned in the recent context of the interaction to be nothing
Substep ii. Exit this procedure, returning as a result: successful rules for determining this goal that have not been discussed

GENQTYPE.RULEPK024

Purpose: Select a question type suitable for the rule under discussion

Step1: Apply all rules that are appropriate:

T-RULE24.01

If: True
Then: Define the valid conclusion formats for a question to be the "correct conclusion"

T-RULE24.02

If: 1) The rule being presented does succeed, or
2) This topic is not a true/false parameter
Then: Include the "wrong conclusion" in the list of the valid

conclusion formats for a question

Comment: If yes/no rule failed, there is the danger that it will be stated as another, correct rule -- e.g., when X => Y and not X => not Y are valid rules

T-RULE24.03

If: 1) This rule does not mention all valid hypotheses in its conclusion, and
2) This topic is not a true/false parameter
Then: Include "multiple choice" in the list of the valid conclusion formats for a question

Comment: Can't make multiple choice when all possible vals are in the conclusion

T-RULE24.04

If: 1) This is a quiz about two related rules,
2) There are related hypotheses in the two rules, and
3) There are not contrary hypotheses made by the two related rules
Then: Include "confirmation of related evidence" in the list of the valid conclusion formats for a question

Comment: Confirmation question requires that there be a second rule with some of the same values and same sign cfs, but no contrary evidence

T-RULE24.05

If: True
Then: To determine the format for the conclusion part of the question, compute the following: a random selection from among the valid conclusion formats for a question

Step2: Apply all rules that are appropriate:

T-RULE24.06

If: 1) The format for the conclusion part of the question is not either "multiple choice" or the "wrong conclusion", or
2) The rule being presented does succeed
Then: Include the "facts of the rule's premise" in the list of the valid premise formats for a question

Comment: If rule fails can't give facts and a wrong conclusion or multiple choice

T-RULE24.07

If: 1) The format for the conclusion part of the question is not either "multiple choice" or the "wrong conclusion", and
2) There are factors in the rule's premise that are not true/false parameters
Then: Include "multiple choice" in the list of the valid premise formats for a question

Comment: Use multiple choice premise when action is not multi or wrong and there is a non-yes/no premise factor

T-RULE24.08

If: The rule being presented does not succeed
Then: Include "right premise hypothesis" in the list of the valid premise formats for a question

Comment: Use right hypothesis only when rule fails

T-RULE24.09

If: 1) There are factors in the rule's premise that are not true/false parameters, and
2) A: The format for the conclusion part of the question is not either "multiple choice" or the "wrong conclusion", or
B: The rule under discussion is a tabular rule
Then: Include "wrong premise hypothesis" in the list of the valid premise formats for a question

Comment: Use wrong hypothesis when there are non-yes/no parms to smash, the rule is unique, and the action is either a table or not presented in multiple choice form. (Uniqueness not currently checked.)

T-RULE24.1

If: True
Then: To determine the format for the premise part of the question, compute the following: a random selection from among the valid premise formats for a question

HYPVALRESPONSE.RULEPK025

Purpose: Respond to the hypothesis that the goal currently being discussed is this value

Step1: Apply the first rule that is appropriate:

T-RULE25.01

If: 1) This hypothesized value is wrong,

2) The number of domain rules having values that were not mentioned by the student is equal to 1, and
3) It is not true that
A: Domain rules for which the tutor has weak evidence that the student considered them to make his hypothesis,
B: Domain rules that the tutor believes the student did not consider when making his hypothesis, or
C: Domain rules that support a wrong hypothesis

Then: Substep i. If: There are domain rules that the tutor strongly believes were used by the student to make his hypothesis
Then: Tell the student that domain rules that the tutor strongly believes were used by the student to make his hypothesis confirm his hypothesis

Substep ii. Generate a question about relevant rules, using "key factors" format in the premise part and "fill-in" format in the conclusion

T-RULE25.02

If: This hypothesized value is wrong
Then: Present the conclusion that the goal currently being discussed is (not) this value [Rulepk026]

Comment: NB: There can't be just one wrongrule and no missingrules; if there is a wrong rule, then there must be contrary evidence the student did not consider, by definition. Here we quiz S to see if he knows of evidence that is contrary to his hypothesis.

T-RULE25.03

If: 1) There are domain rules that the tutor strongly believes were used by the student to make his hypothesis,
2) There are not domain rules for which the tutor has weak evidence that the student considered them to make his hypothesis, and
3) There are not domain rules that the tutor believes the student did not consider when making his hypothesis
Then: Tell the student that domain rules that the tutor strongly believes were used by the student to make his hypothesis confirm his hypothesis

Comment: Don't check for wrong or missingrules here; that will be taken care of when discussing wrong or missing values.

T-RULE25.04

If: 1) It is not true that
A: There are domain rules having values that were not mentioned by the student,
B: There are domain rules that support a wrong hypothesis, or

C: There are domain rules that the tutor believes the student did not consider when making his hypothesis,
2) The number of domain rules for which the tutor has weak evidence that the student considered them to make his hypothesis is equal to 1, and
3) The number of values hypothesized by the student is greater than 1
Then: Substep i. If: There are domain rules that the tutor strongly believes were used by the student to make his hypothesis
Then: Tell the student that domain rules that the tutor strongly believes were used by the student to make his hypothesis confirm his hypothesis
Substep ii. Generate a question about the goal currently being discussed, using "key factors" format in the premise part and "fill-in" format in the conclusion

T-RULE25.05

If: True
Then: Generate a question about the goal currently being discussed, using "fill-in of known factors" format in the premise part and the "correct conclusion" format in the conclusion

IKNOWOPTION.RULEPK044

Purpose: Respond to the student's claim that he knows the goal currently being discussed

Step1: Apply the first rule that is appropriate:

T-RULE44.01

If: Discussion of the factor is complete
Then: Say: already-discussed

T-RULE44.02

If: It is appropriate now to request the reported value for the factor
Then: Substep i. Say: sneverasked
Substep ii. Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate

T-RULE44.03

If: The expert has not reached a final conclusion about the factor

Then: Substep i. Discuss the student's hypothesis [Proc014]
Substep iii. Say that the dialogue is returning to discussion of the goal currently being discussed
Substep iv. If: Discussion of the factor is not complete
Then: Say: mention-option

T-RULE44.04

If: True
Then: Substep i. Say: agree-possibility
Substep ii. Discuss final conclusion for the factor by giving the final answer or discussing an hypothesis [Proc005]
-> A prompt should not be given before starting discussion of this goal
Substep iii. Say that the dialogue is returning to discussion of the goal currently being discussed

NONEEDTOASK.RULEPK032

Purpose: Inform the student that it is not necessary to request the reported value for the factor in this case.

Step1: Apply the first rule that is appropriate:

T-RULE32.01

If: The student has requested the reported value already
Then: Substep i. Say: already-told
Substep ii. Give the student the reported value of the factor

T-RULE32.02

If: This factor is not one that a consultant normally asks
Then: Say: not-askable

T-RULE32.03

If: It is not true that the expert did find it necessary to inquire about the factor
Then: Explain why the expert did not ask about the factor [Rulepk033]

T-RULE32.04

If: 1) A screening question is asked before considering the factor, and
2) It is appropriate now to request the reported value for the screening parameter
Then: Substep i. Define the factor the student has inquired about to

be the screening parameter
Substep ii. Supply requested data to the student, with introduction and related information, as appropriate [Proc012]
-> The request for data has already been determined to be appropriate
-> The student has requested this data explicitly

T-RULE32.05

If: True
Then: Say: premature-question

PRESENTVALS.RULEPK026

Purpose: Present the conclusion that the goal currently being discussed is (not) this value

Step1: Apply the first rule that is appropriate:

T-RULE26.01

If: The number of rules that conclude this value is not equal to 1
Then: Generate a question about the goal currently being discussed, using "fill-in of known factors" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE26.02

If: 1) These rules is not on the lesson plan, or
2) The student model is strong for the rule
Then: Affirm the conclusions made by the rule by simply stating the key factors and values to be concluded

T-RULE26.03

If: The goal currently being discussed is a true/false parameter
Then: Generate a question about the goal currently being discussed, using the "facts of the rule's premise" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE26.04

If: True
Then: Generate a question about the goal currently being discussed, using "key factors" format in the premise part and "fill-in" format in the conclusion

RULEINTEREST.RULEPK027

Purpose: Determine how "interesting" it is to present the rule at this time

Step1: Apply all rules that are appropriate:

T-RULE27.01

If: 1) The rule succeeded in this context, and
2) The following value has been determined: a hint value can be suggested for this rule, based on its effect on the current hypothesis
Then: Modify the cumulative belief that this rule should be presented by the sum over the computed value as follows:
If the evidence contributed is still insignificant then .1
If a new insignificant value is contributed then .1
If "confirmation of related evidence" then .3
If a new significant value is contributed then .5
If a new strongly significant value is contributed then .75
If an insignificant value becomes significant then .8
If an old value is now insignificant then .85
If belief in an old value is strongly contradicted then .95

T-RULE27.02

If: 1) The cumulative belief that this rule should be presented is greater than .7, and
2) The student model is strong for the rule under discussion
Then: Modify the cumulative belief that this rule should be presented by .3

T-RULE27.03

If: The rule under discussion is not on the lesson plan
Then: Modify the cumulative belief that this rule should be presented by -.5

T-RULE27.04

If: The following value has been determined: how close the rule under discussion is to firing
Then: Modify the cumulative belief that this rule should be presented by the computed value as follows:
If the rule has fired then .5
If only case data remains to be requested for the rule to fire then .3
If one or more subgoals remain to be determined, but some preconditions can be evaluated then .2
If otherwise then 0.0

T-RULE27.05

If: The rule under discussion mentions tables or lists in its premise
Then: Modify the cumulative belief that this rule should be presented by .5

Step2: Exit this procedure, returning as a result: the cumulative belief that this rule should be presented

RULEUSED.RULEPK028

Purpose: Apply modelling t-rules to determine the belief that the rule was considered by the student

Step1: Apply the first rule that is appropriate:

T-RULE28.01

If: 1) The hypothesis does include values that can only be concluded by this d-rule,
2) The hypothesis does not include values that can be concluded by this d-rule, as well as other rules, and
3) There are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be .8

T-RULE28.02

If: 1) The hypothesis does include values that can only be concluded by this d-rule,
2) The hypothesis does not include values that can be concluded by this d-rule, as well as other rules, and
3) It is not true that there are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be 1.0

T-RULE28.03

If: 1) The hypothesis does include values that can only be concluded by this d-rule,
2) The hypothesis does include values that can be concluded by this d-rule, as well as other rules, and
3) There are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be .6

T-RULE28.04

If: 1) The hypothesis does include values that can only be concluded by this d-rule,
2) The hypothesis does include values that can be concluded by

this d-rule, as well as other rules, and
3) It is not true that there are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be .7

T-RULE28.05

If: 1) The hypothesis does include values that can be concluded by this d-rule, as well as other rules,
2) The hypothesis does not include values that can only be concluded by this d-rule, and
3) There are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be -.7

T-RULE28.06

If: 1) The hypothesis does include values that can be concluded by this d-rule, as well as other rules,
2) The hypothesis does not include values that can only be concluded by this d-rule, and
3) It is not true that there are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be .4

T-RULE28.07

If: 1) The hypothesis does not include values that can only be concluded by this d-rule,
2) The hypothesis does not include values that can be concluded by this d-rule, as well as other rules, and
3) There are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be -1.0

T-RULE28.08

If: 1) The hypothesis does not include values that can only be concluded by this d-rule,
2) The hypothesis does not include values that can be concluded by this d-rule, as well as other rules, and
3) It is not true that there are values concluded by the d-rule that are missing from the hypothesis
Then: Define the belief that this rule was considered to be 0

Step2: Record that the tutor's belief in the student's knowledge of the rule is the belief that this rule was considered
Step3: Exit this procedure, returning as a result: the belief that this rule was considered

RULEUSEDSREF.RULEPK029

Purpose: In view of the model for all other rules, apply self-referencing t-rules to determine whether the rule was considered by the student

Step1: If: The belief that this rule was considered is equal to 1000
Then: Exit this procedure
Step2: Apply all rules that are appropriate:

T-RULE29.01

If: You believe that this rule was considered, it concludes a value present in the student's hypothesis, and no other rule that mentions this value is believed to have been considered
Then: Modify the cumulative belief that this rule was considered by .4

T-RULE29.02

If: 1) The belief that the student knows this rule is not zero, and
2) The belief that this rule was considered and the belief that the student knows this rule are of the same sign
Then: Modify the cumulative belief that this rule was considered by the belief that the student knows this rule
Else: Record on the agenda that there is an inconsistency in the knowledge model for this rule

T-RULE29.03

If: 1) The rule has been discussed, and
2) The belief that this rule was considered is not negative
Then: Modify the cumulative belief that this rule was considered by .7

T-RULE29.04

If: 1) The belief that this rule was considered is less than the threshold for strong evidence, and
2) You have examined the the factors associated with this rule, selecting every factor for which Discussion of the factor is not complete
Then: Substep i. Modify the cumulative belief that this rule was considered by -.6
Substep ii. Record on the agenda that has subgoals that need to be discussed

T-RULE29.05

If: This domain rule contains a factor that appears in several rules, none of which are believed to have been considered to make the hypothesis

Then: Substep i. Modify the cumulative belief that this rule was considered by -.3
Substep ii. Record on the agenda that this domain rule contains a factor that appears in several rules, none of which are believed to have been considered to make the hypothesis

Step3: Record that the tutor's belief in the student's knowledge of the rule is the cumulative belief that this rule was considered
Step4: Exit this procedure, returning as a result: the cumulative belief that this rule was considered

SAPPLIEDRULE.RULEPK041

Purpose: Determine whether the student knows enough to use the rule

Step1: Apply all rules that are appropriate:

T-RULE41.01

If: The record from past interactions with the student that he knows this rule is zero
Then: Modify the cumulative belief that the student applied this rule by Determine the a priori belief that the student knows the rule [Rulepk040]
Else: Modify the cumulative belief that the student applied this rule by the record from past interactions with the student that he knows this rule

T-RULE41.02

If: Mention-this-session of the rule
Then: Modify the cumulative belief that the student applied this rule by the threshold for strong evidence

T-RULE41.03

If: The rule has been discussed before
Then: Modify the cumulative belief that the student applied this rule by .2

T-RULE41.04

If: 1) The rule has not been discussed, and
2) It is not true that the student knows enough about relevant subgoals so that he can evaluate the premise of the rule
Then: Modify the cumulative belief that the student applied this rule by -1.0

Comment: This sets the SCANTAPPLY property of the rule if one or more clauses can't be applied by the student due to missing

subgoals in his knowledge model

Step2: Record that the belief that the student applied this rule of the rule is the cumulative belief that the student applied this rule
Step3: Exit this procedure, returning as a result: the cumulative belief that the student applied this rule

SUGGESTHYP.RULEPK030

Purpose: Suggest a rule to the student in the form of help for reaching an hypothesis

Step1: Apply the first rule that is appropriate:

T-RULE30.01

If: 1) The premise of the rule being considered includes an iteration, or
2) A: The student model is negative for the rule being considered, and
B: The rule being considered is not on the lesson plan
Then: Affirm the conclusions made by the rule being considered by simply stating the key factors and values to be concluded

T-RULE30.02

If: The hypothesis topic is a true/false parameter
Then: Generate a question about the hypothesis topic, using the "facts of the rule's premise" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE30.03

If: 1) The rule being considered is not the only rule that concludes this value for the hypothesis topic, or
2) The factor being focussed upon
Then: Generate a question about the hypothesis topic, using "key factors" format in the premise part and "fill-in" format in the conclusion

T-RULE30.04

If: True
Then: Generate a question about the hypothesis topic, using "fill-in of known factors" format in the premise part and the "correct conclusion" format in the conclusion

SUGGESTRULE.RULEPK031

Purpose: Present the rule being considered

Step1: Apply the first rule that is appropriate:

T-RULE31.01

If: 1) The rule has fired, and
    2) The premise of the rule being considered includes an iteration
Then: Generate a question about the goal currently being discussed, using the "facts of the rule's premise" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE31.02

If: The rule has fired
Then: Substep i. Select a question type suitable for the rule being considered [Rulepk024]
      Substep ii. Generate a question based upon the rule being considered and evaluate the student's response
      Substep iii. Record that the rule being considered has been discussed

T-RULE31.03

If: 1) What has been mentioned in the recent context of the interaction is either a deeper subgoal or a subtopic relevant to the current goal, and
    2) Discussion of the factor being focussed upon is not complete
Then: Substep i. Say: sgintro
      Substep ii. Discuss the goal with the student in a goal-directed mode [Proc001]
      Substep iii. Say: focusruleintro
      Substep iv. Discuss the rule being considered clause-by-clause [Proc004]

T-RULE31.04

If: 1) The number of factors appearing in this rule which need to be asked by the student is zero, and
    2) The number of subgoals remaining to be determined before this rule can be applied is equal to 1
Then: Substep i. Say: suggestsg
      Substep ii. Discuss the goal with the student in a goal-directed mode [Proc001]
                  -> A prompt should not be given before starting discussion of this goal
      Substep iii. Wrap up the discussion of the rule being considered [Proc017]

T-RULE31.05

If: 1) The premise of the rule being considered includes an iteration,
2) The number of subgoals remaining to be determined before this rule can be applied is greater than 1,
3) A: What has been mentioned in the recent context of the interaction is a hint about a rule,
B: The rule hinted about is the rule being considered, and
C: The number of subgoals remaining to be determined before this rule can be applied is not zero, or
4) The student model is negative for the rule being considered
Then: Substep i. Select an introduction for beginning discussion of the rule being considered [Rulepk038]
Substep ii. Discuss the rule being considered clause-by-clause [Proc004]

T-RULE31.06

If: 1) A hint value can be suggested for this rule, based on its effect on the current hypothesis, and
2) A: What has been mentioned in the recent context of the interaction is not a hint about a rule, or
B: The rule hinted about is not the rule being considered
Then: Substep i. Say: valu-hint
Substep ii. Define the rule hinted about to be the rule being considered
Substep iii. Define what has been mentioned in the recent context of the interaction to be a hint about a rule

T-RULE31.07

If: 1) The number of subgoals remaining to be determined before this rule can be applied is zero,
2) A: 1) What has been mentioned in the recent context of the interaction is a hint about a rule, and
2) The rule hinted about is the rule being considered,
B: The rule being considered is on the lesson plan, or
C: It is not true that the student model is strong for the rule being considered,
3) This topic is not a true/false parameter, and
4) The rule being considered is the only rule that concludes this value for the goal currently being discussed
Then: Generate a question about the goal currently being discussed, using "given new facts" format in the premise part and "fill-in" format in the conclusion

T-RULE31.08

If: 1) The number of subgoals remaining to be determined before this rule can be applied is zero, and
2) A: 1) What has been mentioned in the recent context of the interaction is a hint about a rule, and
2) The rule hinted about is the rule being considered,
B: The rule being considered is on the lesson plan, or

C: It is not true that the student model is strong for the rule being considered
Then: Generate a question about the goal currently being discussed, using "fill-in undiscussed facts" format in the premise part and the "correct conclusion" format in the conclusion

T-RULE31.09

If: True
Then: Substep i. ***missing*** suggestinfoclasses
Substep ii. Say: hintconcls
Substep iii. Define what has been mentioned in the recent context of the interaction to be a hint about a rule
Substep iv. Define the rule hinted about to be the rule being considered

References

The following abbreviations are used in the Reference section.

BBN Bolt, Beranek, & Newman. Cambridge, MA.

TINLAP-1 Theoretical Issues in Natural Language Processing, R. C. Schank & B. L. Nash-Webber (Eds.), Cambridge, MA, 1975.

TINLAP-2 Theoretical Issues in Natural Language Processing, D. L. Waltz (Ed.), University of Illinois at Urbana-Champaign, Illinois, 1978.

IJCAI3 Proceedings of the Third International Joint Conference on Artificial Intelligence, Stanford, Calif., August 1975 (available from SRI International Publications, 330 Ravenswood Ave., Menlo Park, CA 94025).

IJCAI4 Proceedings of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 1975 (available from MIT AI Lab, 545 Technology Square, Cambridge, MA 02138).

IJCAI5 Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Cambridge, Mass., August 1977 (available from MIT AI Lab, 545 Technology Square, Cambridge, MA 02138).

IJCAI6 Proceedings of the Sixth International Joint Conference on Artificial Intelligence, Tokyo, Japan, August 1979 (available from Computer Science Department, Stanford University, Stanford, CA 94305).

[Atkinson, 1972]
Atkinson, R. C. Ingredients for a theory of instruction. *American Psychologist*, 1972, *27*, 921-931.

[Atkinson & Wilson, 1969]
Atkinson, R. C., & Wilson, H. A. (Eds.) *Computer-assisted instruction*. New York: Academic Press, 1969.

[Barr & Atkinson, 1975]
Barr, A., & Atkinson, R. C. Adaptive instructional strategies. Paper presented at the *IPN Symposium 7: Formalized Theories of Thinking and Learning and their Implications for Science Instruction*, Kiel, September 1975.

[Barr, Beard, & Atkinson, 1976]
Barr, A., Beard, M., & Atkinson, R. C. The computer as a tutorial laboratory: The Stanford BIP project. *International Journal of Man-Machine Studies*, 1976, *8*, 567-596.

[Barr, 1979]
Barr, A. Meta-Knowledge and cognition. *IJCAI6*, 1979, 31-33.

[Barr, Bennett, & Clancey, 1979]
Barr, A., Bennett, J., & Clancey, W. *Transfer of expertise: a theme for AI research.* HPP-79-11, Stanford University, April 1979.

[Bennett, et al., 1978]
Bennett, J. S., Creary, L. A., Engelmore, R. M., & Melosh, R. E. *SACON: A knowledge-based consultant in structural analysis*, STAN-CS-78-699 (HPP 78-23). Stanford University, 1978.

[Bennett & Engelmore, 1979]
Bennett, J. S., & Engelmore, R. M. SACON: A knowledge-based consultant in structural analysis, *IJCAI6*, 1979, 47-49.

[Bobrow, et al., 1977]
Bobrow, D. G., Kaplan, R. M., Kay, M., et al. GUS, A frame-driven dialog system. *Artificial Intelligence*, 1977, *8*(2), 155-173.

[Bobrow & Winograd, 1977]
Bobrow, D. .G., & Winograd, T. An overview of KRL, a knowledge representation language. *Cognitive Science*, 1977, *1*(1), 1-46.

[Brown, Burton, & Zydbel, 1973]
Brown, J. S., Burton, R. .R., & Zydbel, F. *A model driven question-answering system for a CAI environment.* AFHRL-TR-72-39. Lowry AFB, CO: Technical Training Division, Air Force Human Resources Laboratory, March 1973.

[Brown, Burton, & Bell, 1974b]
Brown, J. S., Burton, R. R., & Bell, A. G. *Sophie: A Sophisticated Instructional Environment for teaching electronic troubleshooting (An example of AI in CAI)* (BBN Report No. 2790). 1974.

[Brown & Burton, 1975]
Brown, J. S., & Burton, R. Multiple representations of knowledge for tutorial reasoning. In D. G. Bobrow & A. Collins (Eds.), *Representation and Understanding.* New York: Academic Press, 1975, pp. 311-349.

[Brown, et al., 1975]
Brown, J.S., Burton R., Miller M., deKleer, J., Purcell, S., Hausmann, C., & Bobrow, R. *Steps toward a theoretic foundation for complex, knowledge-based CAI* (BBN Report No. 3135). 1975.

[Brown, Rubinstein, & Burton, 1976]
Brown, J. S., Rubinstein, R., & Burton, R. *Reactive learning environment for computer assisted electronics instruction* (BBN Report No. 3314). 1976.

[Brown, 1977a]
Brown, J. S. Remarks on building expert systems (Report on a panel on Applications of Artificial Intelligence). *IJCAI5*, 1977, 1003-1005. (a)

[Brown, 1977b]
Brown, J. S. Uses of artificial intelligence and advanced computer technology in education. In R. J. Seidel & M. Rubin (Eds.), *Computers and Communications: Implications for Education*. New York: Academic Press, 1977. (b)

[Brown, Collins, & Harris, 1977]
Brown, J. S., Collins, A., & Harris, G. Artificial intelligence and learning strategies. In O'Neill (Ed.), *Learning Strategies*. New York: Academic Press, 1977.

[Brown & Goldstein, 1977]
Brown, J. S., & Goldstein, I. P. *Computers in a learning society*, Testimony for the House Science & Technology Subcommittee on Domestic and International Planning, Analysis, & Cooperation, October 1977.

[Brown & Burton, 1978]
Brown, J. S., & Burton, R. R. Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science*, 1978, 2(2), 155-192.

[Bruce, 1975]
Bruce, B. C. Generation as a social action. *TINLAP-I*, 1975, 74-77.

[Bruner, 1966]
Bruner, J. S. *Toward a theory of instruction*. Cambridge: Harvard University Press, 1966.

[Burton, 1976]
Burton, R. R. *Semantic grammar: An engineering technique for constructing natural language understanding systems* (BBN Report No. 3453). 1976.

[Burton & Brown, 1976]
Burton, R. R., & Brown, J. S. A tutoring and student modelling paradigm for gaming environments. *Proceedings for the Symposium on Computer Science and Education*, Irvine, CA, February 1976. (Also, *SIGCSE Bulletin*, 1976, *8*, 236-246.)

[Burton, 1979]
Burton, R. R. An investigation of computer coaching for informal learning activities. *The International Journal of Man-Machine Studies*, 1979, *11*, 5-24.

[Carbonell, 1970]
Carbonell, J. R. *Mixed-initiative man-computer instructional dialogues* (BBN Report No. 1971). 1970.

[Carbonell & Collins, 1973]
Carbonell, J. R., & Collins, A. Natural semantics in Artificial Intelligence. *IJCAI3*, 1973, 344-351.

[Carbonell, 1978]
Carbonell, J. R. Intentionality and human conversations. *TINLAP-2*, 1978, 141-148.

[Carr & Goldstein, 1977]
Carr, B., & Goldstein, I. *Overlays: A theory of modeling for computer aided instruction*, AI Memo 406, Massachusetts Institute of Technology, 1977.

[Clancey, 1978]
Clancey, W. J. *An Antibiotic Therapy Selector which Provides for Explanations*. HPP-78-26, Stanford University, December 1978.

[Collins, et al., 1975]
Collins, A., Warnock, E. H., Aiello, N., & Miller, M. L. Reasoning from incomplete knowledge. In D. G. Bobrow & A. Collins, *Representation and understanding*. New York: Academic Press, 1975, pp. 383-415.

[Collins, 1976]
Collins, A. Processes in acquiring knowledge. In R. C. Anderson, R. J. Spiro, & W. E. Montague (Eds.) *Schooling and the acquisition of knowledge*. Hillsdale, NJ: Erlbaum Assoc., 1976, pp. 339-363.

[Collins, 1978]
Collins, A. Fragments of a theory of human plausible reasoning. *TINLAP-2*, 1978, 194-201.

[Crowder, 1962]
Crowder, N. A. Intrinsic and extrinsic programming. In J. E. Coulson (Ed.), *Proceedings of the conference on application of digital computers to automated instruction*, New York: Wiley, 1962, 58-55.

[Davis, 1976]
Davis, R. *Applications of meta-level knowledge to the construction, maintenance and use of large knowledge bases* (STAN-CS-76-552, HPP-76-7). Stanford University, July 1976.

[Davis, Buchanan, & Shortliffe, 1977]
Davis, R., Buchanan, B., & Shortliffe, E. H. Production rules as a representation for a knowledge-base consultation program. *Journal of Artificial Intelligence*, 1977, *8*(1), 15-45.

[Davis & Buchanan, 1977]
Davis, R., & Buchanan, B. Meta-level knowledge: Overview and applications *IJCAI5*, 1977, 920-927.

[Davis & King, 1977]
Davis, R., & King, J. J. An overview of production systems. In E. W. Elcock & D. Michie (Eds.), *Machine intelligence 8*. New York: Wylie & Sons, 1977, pp. 300-332.

[de Kleer, 1979]
de Kleer, J. The origin and resolution of ambiguities in causal arguments. *IJCAI6*, 1979, 197-203.

[Deutsch, 1974]
Deutsch, B. G. The structure of task-oriented dialogs. *Proceedings of the IEEE Symposium for Speech Recognition*, Carnegie-Mellon University, April, 1974, pp. 250-253.

[Elstein, Shulman, & Sprafka, 1978]
Elstein, A. S., Shulman, L. S., & Sprafka, S. A. *Medical problem-solving: An analysis of clinical reasoning*. Cambridge: Harvard University Press, 1978.

[Entwisle & Entwisle, 1963]
Entwisle, G., & Entwisle D. R. The use of a digitial computer as a teaching machine. *Journal of Medical Education*, 1963, *38*, 803-812.

[Fagan, et al., 1979]
Fagan, L. M., Kunz, J. C., Feigenbaum, E. A., & Osborn, J. J. Representation of dynamic clinical knowledge: measurement interpretation in the intensive care unit. *IJCAI6*, 1979, 260-262.

[Faught, 1977]
Faught, W. S. *Motivation and Intensionality in a Computer Simulation Model*. STAN-CS-77-633 (AIM-305). Stanford University, September 1977.

[Feigenbaum, 1977]
Feigenbaum, E. A. The art of Artificial Intelligence: I. Themes and case studies of knowledge engineering. *IJCAI5*, 1977, 1014-1029.

[Feurzeig, et al., 1964]
Feurzeig, W., Munter, P., Swets, J., & Breen, M. Computer-aided teaching in medical diagnosis. *Journal of Medical Education*, 1964, *39*, 746-755.

[Flavell, 1977]
Flavell, J. *Metacognitive development*. Presented at the NATO Advanced Study Institute on Structural and Process Theories of Complex Human Behavior, Banff, June, 1977.

[Flavell & Wellman, 1977]
Flavell, J., & Wellman, H. Metamemory. In R. Kail & J. Hagen (Eds.), *Perspectives on the development of memory and cognition*. Hillsdale, NJ: Erlbaum Associates, 1977.

[Fletcher, 1975]
Fletcher, J. D. Modeling the learner in computer-assisted instruction. *Journal of Computer-Based Instruction*, 1975, *1*, 118-126.

[Goldberg, 1973]
Goldberg, A. *Computer-assisted instruction: The application of theorem-proving to adaptive response analysis* (Tech. Rep. 203). Institute for Mathematical Studies in the Social Sciences, Stanford University, 1973.

[Goldstein, 1977]
Goldstein, I. *The computer as coach: An athletic paradigm for intellectual education*, AI Memo 389, Massachusetts Institute of Technology, 1977.

[Goldstein & Papert, 1977]
Goldstein, I., & Papert, S. Artificial intelligence, language, and the study of knowledge. *Cognitive Science*, 1977, *1*(1), 84-123.

[Goldstein & Roberts, 1977]
Goldstein, I., & Roberts, B. *NUDGE, a Knowledge-based scheduling program*, AI Memo 405, Massachusetts Institute of Technology, 1977.

[Goldstein, 1978]
Goldstein, I. *Developing a computational representation for problem solving skills*. Proceedings of the Carnegie-Mellon Conference on Problem Solving and Education: Issues in Teaching and Research, October 9-10, 1978.

[Goldstein, 1979]
Goldstein, I. The genetic epistemology of rule systems. *The International Journal of Man-Machine Studies*, 1979, *11*, 51-77.

[Gordon, 1971]
Gordon D., & Lakoff, G. Conversational postulates. Papers from *Seventh Regional Meeting, Chicago Linguistic Society*. Chicago: University of Chicago Linguistics Department, 1971.

[Greeno, 1976]
Greeno, James G. Cognitive objectives of instruction: Theory of knowledge for solving problems and answering questions. In Klahr (Ed.), *Cognition and Instruction*. Hillsdale, NJ: Erlbaum Associates.

[Grice, 1975]
Grice, H. P. Logic and conversation. In P. Cole & J. L. Morgan (Eds.), *Studies in syntax* (Vol. 3). New York: Seminar Press, 1975.

[Grignetti, Hausman, & Gould, 1975]
Grignetti, M. C., Hausmann, C., & Gould, L. An *intelligent* on-line assistant and tutor--NLS-SCHOLAR. *Proceedings of the National Computer Conference*, San Diego, Calif., 1975, pp. 775-781.

[Grosz, 1977]
Grosz, Barbara J. The representation and use of focus in a system for understanding dialogues. *IJCAI5*, 1977, 67-76.

[Harrison's, 1974]
Harrison's Principles of Internal Medicine, (7th ed.). M. M. Wintrobe, G. W. Thorn, R. D. Adams, E. Braunwald, K. J. Isselbacher, & R. G. Petersdorf (Eds.). New York: McGraw-Hill, 1974.

[Harless, et al., 1971]
Harless, W. .G., Crennon, G. .G., Marxer, J. J., Root, J. A., Miller, G. E. CASE: A computer-aided simulation of the clinical encounter. *Journal of Medical Education*, 1971, *46*, 443-448.

[Hart, 1975]
Hart, Peter E. Progress on a computer based consultant. *IJCAI4*, 1975, 831-841.

[Hart & Koffman, 1975]
Hart, R. O., & Koffman, E. B. A Student Oriented Natural Language Environment for Learning LISP. *IJCAI4*, 1975, 391-396.

[Heiser, Brooks, & Ballard, 1978]
Heiser, J. F., Brooks, R. E., & Ballard, J. P. Progress report: A computerized psychopharmacology advisor. *Proceedings of the 11th Collegium Internationale Neuro-Psychopharmacologicum*, Vienna, Austria. 1978.

[Hoffer, et al., 1975]
Hoffer, E. P., Barnett, O. G., Farquhar, B. B., Prather, P. A. CAI in medicine. *Annual Review of Biophysical Engineering*, 1975, *4*, 103-118.

[Howe, 1973]
Howe, J. A. M. Individualizing computer-assisted instruction. In A. Elithorn & D. Jones (Eds.), *Artificial and human thinking*. Amsterdam: Elsevier, 1973, pp. 94-101.

[Kimball, 1973]
Kimball, R. B. *Self-optimizing computer-assisted tutoring: theory and practice* (Tech. Rep. 206). Stanford, Calif.: Stanford University, Institute for Mathematical Studies in the Social Sciences, 1973.

[Koffman & Blount, 1973]
Koffman, E. B., & Blount, S. E. Artificial Intelligence and automatic programming in CAI. *IJCAI3*, 1973, 86-94.

[Kunz, et al., 1978]
Kunz, J. C., Fallan, R. J., McClung, D. H., Osborn, J. J., Votteri, B. A., Nii, H. P., Aikins, J. S., Fagan, L. M., & Feigenbaum, E. A. *A physiological rule-based system for interpreting pulmonary function test results.* HPP-78-19, Stanford University, November 1978.

[Laubsch, 1975]
Laubsch, J. H. Some thoughts about representing knowledge in instructional systems. *IJCAI4*, 1975, 122-125.

[Lenat, 1976]
Lenat, D. B. *AM: An Artificial Intelligence approach to discovery in mathematics as heuristic search.* CS-STAN-76-570 (AIM-286), Stanford University, July 1976.

[Minsky, 1975]
Minsky M. A framework for representing knowledge. In P. Winston (Ed.), *The psychology of computer vision.* New York: McGraw-Hill, 1975.

[Levin & Moore, 1977]
Levin, J. A., & Moore, J. A. Dialogue-games: Metacommunication structures for natural language interaction. *Cognitive Science*, 1977, *1*(4), 395-420.

[Nelson, 1974]
Nelson, T. H. *Computer lib* (Copies available from Hugo's Book Service, Box 2622, Chicago, Illinois 60690.) 1974.

[Nii & Feigenbaum, 1977]
Nii, P. H., & Feigenbaum, E. A. *Rule-based understanding of signals.* STAN-CS-77-612 (HPP-77-7), Stanford University, 1977.

[Nilsson, 1971]
Nilsson, N. J. *Problem-solving methods in Artificial Intelligence.* New York: McGraw Hill, 1971.

[Norman, Gentner, & Stevens, 1976]
Norman D. A., Gentner, D. R., & Stevens, A. L. Comments on learning schemata and memory representation. In Klahr (Ed.), *Cognition and instruction.* Hillsdale, NJ: Erlbaum Associates.

[O'Shea, 1979]
O'Shea, T. A self-improving quadratic tutor. *The International Journal of Man-Machine Studies*, 1979, *11*, 97-124.

[Papert, 1970]
Papert, S. Teaching children programming. *IFIP Conference on Computer Education.* Amsterdam: North Holland, 1970.

[Pople, 1975]
Pople, H. E. DIALOG: A model of diagnostic logic for internal medicine. *IJCAI4*, 1975, 848-855.

[Pople, 1977]
Pople, H. E. The formation of composite hypotheses in diagnostic problem solving--an exercise in synthetic reasoning. *IJCAI5*, 1977, 1030-1037.

[Reither, 1978]
Reither, R. On reasoning by default. *TINLAP-2*, 1978, 210-218.

[Rieger & Grinberg, 1977]
Rieger, C., & Grinberg, M. The declarative representation and procedural simulation of causality in physical mechanisms. *IJCAI5*, 1977, 250-255.

[Schank & Abelson, 1975]
Schank, R.C., & Abelson, R. P. Scripts, plans, and knowledge. *IJCAI4*, 1975, 151-157.

[Scott, et al., 1977]
Scott, A. C., Clancey, W. J., Davis, R., Shortliffe, E. H. Explanation capabilities of production-based consultation systems. *American Journal of Computational Linguistics*, 1977, Microfiche 62.

[Self, 1974]
Self, J. A. Student models in computer-aided instruction. *International Journal of Man-Machine Studies*, 1974, *6*, 261-276.

[Shortliffe, 1976]
Shortliffe, E. H. *MYCIN: A rule-based computer program for advising physicians regarding antimicrobial therapy selection*. Ph.D. dissertation in Medical Information Sciences, Stanford University, 1974. (Also, *Computer-based medical consultations: MYCIN*. New York: Elsevier, 1976.)

[Shortliffe & Buchanan, 1975]
Shortliffe, E. H., & Buchanan, B. G. A model of inexact reasoning in medicine. *Mathematical Biosciences*, 1975, *23*, 351-379.

[Sleeman, 1977]
Sleeman, D. A system which allows students to explore algorithms. *IJCAI5*, 1977, 780-786.

[Sprosty, 1963]
Sprosty, P. J. The use of questions in the diagnostic problem solving process. In Jacquez (Ed.), *The diagnostic process* (Proceedings of a conference held at the University of Michigan), 1963, pp. 281-308.

[Stansfield, Carr, & Goldstein, 1976]
Stansfield, J. L., Carr, B. P., & Goldstein, I. P. *WUMPUS Advisor I, a first implementation of a program that tutors logical and probabilistic reasoning skills*. AI Memo 381, Massachusetts Institute of Technology, October 1976.

[Steele, et al., 1978]
Steele, A. A., Davis, P. J., Hoffer, E. P., & Famiglietti, K. T. A computer-assisted instruction (CAI) program in diseases of the thyroid gland (THYROID). *Computers and Biomedical Research*, 1978, *11*, 133-146.

[Stefik, 1979]
Stefik, M. An examination of a frame-structured representation system. *IJCAI6*, 1979, 845-852.

[Stevens & Collins, 1977]
Stevens, A. L., & Collins, A. *The goal structure of a Socratic tutor* (BBN Report No. 3518). 1977.

[Stevens, Collins, & Goldin, 1978]
Stevens, A. L., Collins, A., & Goldin, S. *Diagnosing student's misconceptions in causal models* (BBN Report No. 3786). 1978.

[Stevens & Collins, 1978]
Stevens, A. L., & Collins, A. Multiple conceptual models of a complex system (BBN Report No. 3923). 1978. To appear in R. Snow, P. Federico and W. Mantague (Eds.), *Aptitude, learning and instruction: Cognitive process analysis*.

[Suppes & Morningstar, 1972]
Suppes, P., & Morningstar, M. *Computer-assisted instruction at Stanford, 1966-68: Data, models, and evaluation of the arithmetic programs*. New York: Academic Press, 1972.

[Swartout, 1977]
Swartout, W. A digitalis therapy advisor with explanations. *IJCAI5*, 1977, 819-825.

[Swets & Feurzeig, 1965]
Swets, J. A., & Feurzeig, W. Computer-aided instruction. *Science*, 1965, *150*, 572-576.

[Szolovits & Pauker, 1978]
Szolovits, P., & Pauker, S. G. Categorical and probabilistic reasoning in medical diagnosis. *Artificial Intelligence*, 1978, *11*(1), 115-144.

[van Melle, 1979]
van Melle, W. A domain-independent production-rule system for consultation programs. *IJCAI6*, 1979, 923-925.

[Weber & Hagamen, 1972]
Weber, J. C., & Hagamen, W. D. ATS: A new system for computer-mediated tutorials in medical education. *Journal of Medical Education*, 1972, *47*, 637-644.

[Weiss, Kulikowski, & Safir, 1977]
Weiss, S., Kulikowski, C., & Safir, A. A model-based consultation system for the long-term management of glaucoma. *IJCAI5*, 1977, 826-832.

[Wescourt, Beard, & Gould, 1977]
Wescourt, K. T., Beard, M., & Gould, M. *Knowledge-based adaptive curriculum sequences for CAI: Application of a network representation* (Tech. Rep. 288). Institute for Mathematical Studies in the Social Sciences, Stanford University, 1977.

[Wescourt, et al., 1977]
Wescourt, K. T., Beard, M., Gould, M., & Barr, A. (1977). *Knowledge-based CAI: CINS for individualized curriculum sequencing* (Tech. Rep. 290). Institute for Mathematical Studies in the Social Sciences, Stanford University, 1977.

[Wescourt, et al., 1978]
Wescourt, K. T., & Hemphill, L. (1978). *Representing and teaching knowledge for troubleshooting/debugging* (Tech. Rep. 292). Institute for Mathematical Studies in the Social Sciences, Stanford University, 1978.

[Wexler, 1970]
Wexler, J. D. Information networks in generative computer-assisted instruction. *IEEE Transactions on Man-machine Systems*, 1970, *MMS-11*(4), 181-190.

[Winograd, 1977]
Winograd, T. A framework for understanding discourse. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension*. Hillsdale, NJ: Erlbaum Associates, 1977, pp. 63-88.

[Wittgenstein, 1958]
Wittgenstein, L. *Philosophical investigations*. Oxford: Basil, Blackwell, & Mott Ltd., 1958.

[Yu, et al., 1978]
Yu, V. L., Buchanan, B. G., Shortliffe, E. H., Wraith, S. M., Davis, R., Scott, A. C., & Cohen, S. N. Evaluating the performance of a computer-based consultant. *Computer Programs in Biomedicine*, 1978, *9*, 1978, 95-102.

Author Index

[Atkinson & Wilson, 1969] 23
[Atkinson, 1972] 24

[Barr & Atkinson, 1975] 27
[Barr, 1979] 157, 187, 189
[Barr, Beard, & Atkinson, 1976] 28
[Barr, Bennett, & Clancey, 1979] 268, 359
[Bennett & Engelmore, 1979] 138
[Bennett, et al., 1978] 95, 316
[Bobrow & Winograd, 1977] 177, 189
[Bobrow, et al., 1977] 112, 198, 348
[Brown & Burton, 1975] 102
[Brown & Burton, 1978] 34
[Brown & Goldstein, 1977] 27
[Brown, 1977a] 356
[Brown, 1977b] 14, 26, 33, 34
[Brown, Burton, & Bell, 1974b] 13, 26, 28
[Brown, Burton, & Zydbel, 1973] 13
[Brown, Collins, & Harris, 1977] 96, 182, 304
[Brown, et al., 1975] 33, 209, 347
[Brown, Rubinstein, & Burton, 1976] 13, 25, 27, 102, 347
[Bruce, 1975] 198, 199
[Bruner, 1966] 211
[Burton & Brown, 1976] 27, 347
[Burton, 1976] 347
[Burton, 1979] 33, 187, 197, 298, 344, 349, 355

[Carbonell & Collins, 1973] 28
[Carbonell, 1970] 13, 23, 24, 26, 347
[Carbonell, 1978] 344, 348
[Carr & Goldstein, 1977] 33, 102, 111, 280, 355
[Clancey, 1978] 84
[Collins, 1976] 13, 33, 34, 209, 344, 347
[Collins, 1978] 187, 189
[Collins, et al., 1975] 187
[Crowder, 1962] 23

[Davis & Buchanan, 1977] 19, 96, 98, 187
[Davis & King, 1977] 94
[Davis, 1976] 10, 11, 98, 119, 133, 182, 263
[Davis, Buchanan, & Shortliffe, 1977] 9
[de Kleer, 1979] 102, 175
[Deutsch, 1974] 198, 349

[Elstein, Shulman, & Sprafka, 1978] 354
[Entwisle & Entwisle, 1963] 36

[Fagan, et al., 1979] 152
[Faught, 1977] 199, 348
[Feigenbaum, 1977] 98
[Feurzeig, et al., 1964] 85, 344
[Flavell & Wellman, 1977] 189
[Flavell, 1977] 187, 226
[Fletcher, 1975] 24

[Goldberg, 1973] 26
[Goldstein & Papert, 1977] 29
[Goldstein & Roberts, 1977] 189
[Goldstein, 1977] 27, 28, 29, 35, 111, 197, 249, 347, 357
[Goldstein, 1978] 137
[Goldstein, 1979] 32
[Gordon, 1971] 197
[Greeno, 1976] 183
[Grice, 1975] 197, 211
[Grosz, 1977] 112, 348, 349

[Harless, et al., 1971] 36, 85, 89
[Harrison's, 1974] 138, 151, 156
[Hart & Koffman, 1975] 347
[Hart, 1975] 98, 150, 347
[Heiser, Brooks, & Ballard, 1978] 95
[Hoffer, et al., 1975] 36
[Howe, 1973] 22

[Kimball, 1973] 24
[Koffman & Blount, 1973] 25, 29, 33
[Kunz, et al., 1978] 95, 316

[Laubsch, 1975] 26, 28, 30
[Lenat, 1976] 2
[Levin & Moore, 1977] 198

[Minsky, 1975] 24

[Nelson, 1974] 273
[Nii & Feigenbaum, 1977] 98
[Nilsson, 1971] 365
[Norman, Gentner, & Stevens, 1976] 33, 155

[O'Shea, 1979] 357

[Papert, 1970] 22
[Pople, 1975] 2
[Pople, 1977] 98, 356

[Reither, 1978] 172
[Rieger & Grinberg, 1977] 175

[Schank & Abelson, 1975] 168
[Scott, et al., 1977] 98
[Self, 1974] 27
[Shortliffe & Buchanan, 1975] 201
[Shortliffe, 1976] 2, 9, 366
[Sleeman, 1977] 349
[Sprosty, 1963] 102, 354
[Stansfield, Carr, & Goldstein, 1976] 29
[Steele, et al., 1978] 36
[Stefik, 1979] 177, 189
[Stevens & Collins, 1977] 196, 209, 347, 374
[Stevens & Collins, 1978] 28, 167, 168
[Stevens, Collins, & Goldin, 1978] 34, 175, 314, 347
[Suppes & Morningstar, 1972] 26
[Swartout, 1977] 99
[Swets & Feurzeig, 1965] 36
[Szolovits & Pauker, 1978] 103

[van Melle, 1979] 16, 95

[Weber & Hagamen, 1972] 36
[Weiss, Kulikowski, & Safir, 1977] 98, 102, 356
[Wescourt, Beard, & Gould, 1977] 308
[Wescourt, et al., 1977] 308
[Wescourt, et al., 1978] 34, 187, 356
[Wexler, 1970] 25
[Winograd, 1977] 112, 198, 277, 349
[Wittgenstein, 1958] 181

[Yu, et al., 1978] 101, 138

Navy

1 Dr. Ed Aiken
Navy Personnel R&D Center
San Diego, CA 92152

1 Dr. Jack R. Borsting
Provost & Academic Dean
U.S. Naval Postgraduate School
Monterey, CA 93940

1 Dr. Robert Breaux
Code N-71
NAVTRAEQUIPCEN
Orlando, FL 32813

1 DR. JOHN F. BROCK
NAVY PERSONNEL R& D CENTER
SAN DIEGO, CA 92152

1 MR. MAURICE CALLAHAN
Pers 23a
Bureau of Naval Personnel
Washington, DC 20370

1 CAPT. H. J. CONNERY
NAVY MEDICAL R&D COMMAND
NNMC
BETHESDA, MD 20014

1 Mr. James S. Duva
Chief, Human Factors Laboratory
Naval Training Equipment Center
(Code N-215)
Orlando, Florida 32813

1 Dr. Richard Elster
Department of Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940

1 DR. PAT FEDERICO
NAVY PERSONNEL R&D CENTER
SAN DIEGO, CA 92152

1 CDR John Ferguson, MSC, USN
Naval Medical R&D Command (Code 44)
National Naval Medical Center
Bethesda, MD 20014

Navy

1 Dr. John Ford
Navy Personnel R&D Center
San Diego, CA 92152

1 CAPT. D.M. GRAGG, MC, USN
HEAD, SECTION ON MEDICAL EDUCATION
UNIFORMED SERVICES UNIV. OF THE
HEALTH SCIENCES
6917 ARLINGTON ROAD
BETHESDA, MD 20014

1 MR. GEORGE N. GRAINE
NAVAL SEA SYSTEMS COMMAND
SEA 05L1C2
WASHINGTON, DC 20362

1 LT Steven D. Harris, MSC, USN
Code 6021
Naval Air Development Center
Warminster, Pennsylvania 18974

1 Dr. Norman J. Kerr
Chief of Naval Technical Training
Naval Air Station Memphis (75)
Millington, TN 38054

1 Dr. Leonard Kroeker
Navy Personnel R&D Center
San Diego, CA 92152

1 CHAIRMAN, LEADERSHIP & LAW DEPT.
DIV. OF PROFESSIONAL DEVELOPMMENT
U.S. NAVAL ACADEMYY
ANNAPOLIS, MD 21402

1 Dr. William L. Maloy
Principal Civilian Advisor for
Education and Training
Naval Training Command, Code 00A
Pensacola, FL 32508

1 CAPT Richard L. Martin
USS Francis Marion (LPA-249)
FPO New York, NY 09501

1 Dr. Sylvia R. Mayer (MCIT)
HQ Electronic Systems Div.
Hanscom AFB
Bedford, MA 01731

1 Dr. James McBride
Code 301
Navy Personnel R&D Center
San Diego, CA 92152

1 DR. WILLIAM MONTAGUE
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213

1 Commanding Officer
U.S. Naval Amphibious School
Coronado, CA 92155

1 Commanding Officer
Naval Health Research
Center
Attn: Library
San Diego, CA 92152

1 Naval Medical R&D Command
Code 44
National Naval Medical Center
Bethesda, MD 20014

1 CAPT Paul Nelson, USN
Chief, Medical Service Corps
Code 7
Bureau of Medicine & Surgery
U. S. Department of the Navy
Washington, DC 20372

1 Library
Navy Personnel R&D Center
San Diego, CA 92152

6 Commanding Officer
Naval Research Laboratory
Code 2627
Washington, DC 20390

1 JOHN OLSEN
CHIEF OF NAVAL EDUCATION &
TRAINING SUPPORT
PENSACOLA, FL 32509

1 Psychologist
ONR Branch Office
495 Summer Street
Boston, MA 02210

1 Psychologist
ONR Branch Office
536 S. Clark Street
Chicago, IL 60605

1 Office of Naval Research
Code 200
Arlington, VA 22217

1 Office of Naval Research
Code 437
800 N. Quincy SStreet
Arlington, VA 22217

5 Personnel & Training Research Programs
(Code 458)
Office of Naval Research
Arlington, VA 22217

1 Psychologist
OFFICE OF NAVAL RESEARCH BRANCH
223 OLD MARYLEBONE ROAD
LONDON, NW, 15TH ENGLAND

1 Psychologist
ONR Branch Office
1030 East Green Street
Pasadena, CA 91101

1 Scientific Director
Office of Naval Research
Scientific Liaison Group/Tokyo
American Embassy
APO San Francisco, CA 96503

1 Scientific Advisor to the Chief of
Naval Personnel (Pers-Or)
Naval Bureau of Personnel
Room 4410, Arlington Annex
Washington, DC 20370

Navy

1 LT Frank C. Fetho, MSC, USNR (Ph.D)
Code L51
Naval Aerospace Medical Research Laborat
Pensacola, FL 32508

1 DR. RICHARD A. POLLAK
ACADEMIC COMPUTING CENTER
U.S. NAVAL ACADEMY
ANNAPOLIS, MD 21402

1 Roger W. Remington, Ph.D
Code L52
NAMRL
Pensacola, FL 32508

1 Dr. Bernard Rimland
Navy Personnel R&D Center
San Diego, CA 92152

1 Mr. Arnold Rubenstein
Naval Personnel Support Technology
Naval Material Command (08T244)
Room 1044, Crystal Plaza #5
2221 Jefferson Davis Highway
Arlington, VA 20360

1 Dr. Worth Scanland
Chief of Naval Education and Training
Code N-5
NAS, Pensacola, FL 32508

1 A. A. SJOHOLM
TECH. SUPPORT, CODE 201
NAVY PERSONNEL R& D CENTER
SAN DIEGO, CA 92152

1 Mr. Robert Smith
Office of Chief of Naval Operations
OP-987E
Washington, DC 20350

1 Dr. Alfred F. Smode
Training Analysis & Evaluation Group
(TAEG)
Dept. of the Navy
Orlando, FL 32813

Navy

1 Dr. Richard Sorensen
Navy Personnel R&D Center
San Diego, CA 92152

1 CDR Charles J. Theisen, JR. MSC, USN
Head Human Factors Engineering Div.
Naval Air Development Center
Warminster, PA 18974

1 W. Gary Thomson
Naval Ocean Systems Center
Code 7132
San Diego, CA 92152

1 DR. H.M. WEST III
DEPUTY ADCNO FOR CIVILIAN PLANNING
AND PROGRAMMING
RM. 2625, ARLINGTON ANNEX
WASHINGTON, DC 20370

Army

1 HQ USAREUE & 7th Army
ODCSOPS
USAAREUE Director of GED
APO New York 09403

1 LCOL Gary Bloedorn
Training Effectiveness Analysis Division
US Army TRADOC Systems Analysis Activity
White Sands Missile Range, NM 88002

1 DR. RALPH DUSEK
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333

1 DR. FRANK J. HARRIS
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333

1 Col. Frank Hart, Director
Training Development Institute
ATTNG-TDI
Ft. Eustis, VA 23604

1 Dr. Ed Johnson
Army Research Institute
5001 Eisenhower Blvd.
Alexandria, VA 22333

1 Dr. Michael Kaplan
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333

1 Dr. Milton S. Katz
Individual Training & Skill
Evaluation Technical Area
U.S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

1 Dr. Beatrice J. Farr
Army Research Institute (PERI-OK)
5001 Eisenhower Avenue
Alexandria, VA 22333

Army

1 Dr. Harold F. O'Neil, Jr.
ATTN: PERI-OK
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333

1 LTCOL Michael T. Plummer
Organizational Effectiveness Division
Office of the Deputy Chief of Staff
for Personnel
Department of the Army
Washington, DC 20301

1 Dr. Robert Sasmor
U. S. Army Research Institute for the
Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

2 Faculty Development Division
Headquarters Sheppard Technical
Training Center (ATC)
Sheppard AFB, TX 76311

1 Director, Training Development
U.S. Army Administration Center
ATTN: Dr. Sherrill
Ft. Benjamin Harrison, IN 46218

1 Dr. Joseph Ward
U.S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Air Force

1 Air University Library
AUL/LSE 76/443
Maxwell AFB, AL 36112

1 DR. T. E. COTTERMAN
AFHRL/ASR
WRIGHT PATTERSON AFB
OHIO 45433

1 DR. G. A. ECKSTRAND
AFHRL/AS
WRIGHT-PATTERSON AFB, OH 45433

1 Dr. Genevieve Haddad
Program Manager
Life Sciences Directorate
AFOSR
Bolling AFB, DC 20332

1 CDR. MERCER
CNET LIAISON OFFICER
AFHRL/FLYING TRAINING DIV.
WILLIAMS AFB, AZ 85224

1 Dr. Donald E. Meyer
U.S. Air Force
ATC/XPTD
Randolph AFB, TX 78148

1 Dr. Ross L. Morgan (AFHRL/ASR)
Wright -Patterson AFB
Ohio 45433

1 Research Branch
AFMPC/DPMYP
Randolph AFB, TX 78148

1 Dr. Marty Rockway (AFHRL/TT)
Lowry AFB
Colorado 80230

1 Jack A. Thorpe, Capt, USAF
Program Manager
Life Sciences Directorate
AFOSR
Bolling AFB, DC 20332

Air Force

1 Brian K. Waters, LCOL, USAF
Air University
Maxwell AFB
Montgomery, AL 36112

Marines

1 H. William Greenup
Education Advisor (E031)
Education Center, MCDEC
Quantico, VA 22134

1 Director, Office of Manpower Utilization
HQ, Marine Corps (MPU)
BCB, Bldg. 2009
Quantico, VA 22134

1 DR. A.L. SLAFKOSKY
SCIENTIFIC ADVISOR (CODE RD-1)
HQ, U.S. MARINE CORPS
WASHINGTON, DC 20380

1 Major Jack Wallace
Headquarters, Marine Corps
OTTI 31
Arlington Annex
Columbia Pike at Arlington Ridge Rd.
Arlington, VA 20380

CoastGuard

1 Mr. Richard Lanterman
PSYCHOLOGICAL RESEARCH (G-P-1/62)
U.S. COAST GUARD HQ
WASHINGTON, DC 20590

Other DoD

1 Dr. Stephen Andriole
ADVANCED RESEARCH PROJECTS AGENCY
1400 WILSON BLVD.
ARLINGTON, VA 22209

12 Defense Documentation Center
Cameron Station, Bldg. 5
Alexandria, VA 22314
Attn: TC

1 Dr. Dexter Fletcher
ADVANCED RESEARCH PROJECTS AGENCY
1400 WILSON BLVD.
ARLINGTON, VA 22209

1 Military Assistant for Training and
Personnel Technology
Office of the Under Secretary of Defense
for Research & Engineering
Room 3D129, The Pentagon
Washington, DC 20301

Civil Govt

1 Dr. Susan Chipman
Basic Skills Program
National Institute of Education
1200 19th Street NW
Washington, DC 20208

11 Mr. James M. Ferstl
Bureau of Training
U.S. Civil Service Commission
Washington, D.C. 20415

1 Dr. Richards J. Heuer
ORPA/AMERS
Washington, DC 20505

1 Dr. Joseph I. Lipson
Division of Science Education
Room W-638
National Science Foundation
Washington, DC 20550

1 Dr. John Mays
National Institute of Education
1200 19th Street NW
Washington, DC 20208

1 William J. McLaurin
Rm. 301, Internal Revenue Service
2221 Jefferson Davis Highway
Arlington, VA 22202

1 Dr. Arthur Melmed
National Intitute of Education
1200 19th Street NW
Washington, DC 20208

1 Dr. Andrew R. Molnar
Science Education Dev.
and Research
National Science Foundation
Washington, DC 20550

1 Dr. H. Wallace Sinaiko
Program Director
Manpower Research and Advisory Services
Smithsonian Institution
801 North Pitt Street
Alexandria, VA 22314

Civil Govt

1 Dr. Thomas G. Sticht
Basic Skills Program
National Institute of Education
1200 19th Street NW
Washington, DC 20208

1 Dr. Frank Withrow
U. S. Office of Education
400 6th Street SW
Washington, DC 20202

1 Dr. Joseph L. Young, Director
Memory & Cognitive Processes
National Science Foundation
Washington, DC 20550

Non Govt

1 Dr. Earl A. Alluisi
HQ, AFHRL (AFSC)
Brooks AFB, TX 78235

1 Dr. John R. Anderson
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

1 Dr. John Annett
Department of Psychology
University of Warwick
Coventry CV4 7AL
ENGLAND

1 DR. MICHAEL ATWOOD
SCIENCE APPLICATIONS INSTITUTE
40 DENVER TECH. CENTER WEST
7935 E. PRENTICE AVENUE
ENGLEWOOD, CO 80110

1 1 psychological research unit
Dept. of Defense (Army Office)
Campbell Park Offices
Canberra ACT 2600, Australia

1 Dr. R. A. Avner
University of Illinois
Computer-Based Educational Research Lab
Urbana, IL 61801

1 Dr. Alan Baddeley
Medical Research Council
Applied Psychology Unit
15 Chaucer Road
Cambridge CB2 2EF
ENGLAND

1 Dr. Patricia Baggett
Department of Psychology
University of Denver
University Park
Denver, CO 80208

1 Ms. Carole A. Bagley
Minnesota Educational Computing
Consortium
2520 Broadway Drive
St. Paul, MN 55113

Non Govt

1 Mr Avron Barr
Department of Computer Science
Stanford University
Stanford, CA 94305

1 Dr. John Bergan
School of Education
University of Arizona
Tuscon AZ 85721

1 Dr. Nicholas A. Bond
Dept. of Psychology
Sacramento State College
600 Jay Street
Sacramento, CA 95819

1 Dr. Lyle Bourne
Department of Psychology
University of Colorado
Boulder, CO 80302

1 Dr. Kenneth Bowles
Institute for Information Sciences
University of California at San Diego
La Jolla, CA 92037

1 Dr. John Brackett
SofTech
460 Totten Pond Road
Waltham, MA 02154

1 Dr. John S. Brown
XEROX Palo Alto Research Center
3333 Coyote Road
Palo Alto, CA 94304

1 DR. C. VICTOR BUNDERSON
WICAT INC.
UNIVERSITY PLAZA, SUITE 10
1160 SO. STATE ST.
OREM, UT 84057

1 Dr. Anthony Cancelli
School of Education
University of Arizona
Tuscon, AZ 85721

Non Govt

1 Charles Myers Library
Livingstone House
Livingstone Road
Stratford
London E15 2LJ
ENGLAND

1 Dr. William Chase
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

1 Dr. Micheline Chi
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

1 Dr. John Chiorini
Litton-Mellonics
Box 1286
Springfield, VA 22151

1 Dr. Kenneth E. Clark
College of Arts & Sciences
University of Rochester
River Campus Station
Rochester, NY 14627

1 Dr. Allan M. Collins
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, Ma 02138

1 Dr. Meredith Crawford
Department of Engineering Administration
George Washington University
Suite 805
2101 L Street N. W.
Washington, DC 20037

1 Dr. Hubert Dreyfus
Department of Philosophy
University of California
Berkely, CA 94720

1 ERIC Facility-Acquisitions
4833 Rugby Avenue
Bethesda, MD 20014

Non Govt

1 Dr. A. J. Eschenbrenner
Dept. E422, Bldg. 101
McDonnell Douglas Astronautics Co.
P.O.Box 516
St. Louis, MO 63166

1 MAJOR I. N. EVONIC
CANADIAN FORCES PERS. APPLIED RESEARCH
1107 AVENUE ROAD
TORONTO, ONTARIO, CANADA

1 Dr. Ed Feigenbaum
Department of Computer Science
Stanford University
Stanford, CA 94305

1 Dr. Richard L. Ferguson
The American College Testing Program
P.O. Box 168
Iowa City, IA 52240

1 Mr. Wallace Feurzeig
Bolt Beranek & Newman, Inc.
50 Moulton St.
Cambridge, MA 02138

1 Dr. Victor Fields
Dept. of Psychology
Montgomery College
Rockville, MD 20850

1 Dr. Edwin A. Fleishman
Advanced Research Resources Organ.
Suite 900
4330 East West Highway
Washington, DC 20014

1 Dr. John R. Frederiksen
Bolt Beranek & Newman
50 Moulton Street
Cambridge, MA 02138

1 Dr. Vernon S. Gerlach
College of Education
146 Payne Bldg. B
Arizona State University
Tempe, AZ 85281

Non Govt

1 DR. ROBERT GLASER
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213

1 Dr. Ira Goldstein
XEROX Palo Alto Research Center
3333 Coyote Road
Palo Alto, CA 94304

1 DR. JAMES G. GREENO
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213

1 Dr. Ron Hambleton
School of Education
University of Massechusetts
Amherst, MA 01002

1 Dr. Barbara Hayes-Roth
The Rand Corporation
1700 Main Street
Santa Monica, CA 90406

1 Dr. Frederick Hayes-Roth
The Rand Corporation
1700 Main Street
Santa Monica, CA 90406

1 Dr. Dustin H. Heuston
Wicat, Inc.
Box 986
Orem, UT 84057

1 Library
HumRRO/Western Division
27857 Berwick Drive
Carmel, CA 93921

1 Dr. Earl Hunt
Dept. of Psychology
University of Washington
Seattle, WA 98105

stration

Non Govt

1 DR. LAWRENCE B. JOHNSON
LAWRENCE JOHNSON & ASSOC., INC.
SUITE 502
2001 S STREET NW
WASHINGTON, DC 20009

1 Dr. Wilson A. Judd
McDonnell-Douglas
Astronautics Co. East
Lowry AFB
Denver, CO 80230

1 Dr. Arnold F. Kanarick
Honeywell, Inc.
2600 Ridgeway Pkwy
Minneapolis, MN 55413

1 Dr. Roger A. Kaufman
203 Dodd Hall
Florida State Univ.
Tallahassee, FL 32306

1 Dr. Walter Kintsch
Department of Psychology
University of Colorado
Boulder, CO 80302

1 Dr. David Kieras
Department of Psychology
University of Arizona
Tuscon, AZ 85721

1 Mr. Marlin Kroger
1117 Via Goleta
Palos Verdes Estates, CA 90274

1 LCOL. C.R.J. LAFLEUR
PERSONNEL APPLIED RESEARCH
NATIONAL DEFENSE HQS
101 COLONEL BY DRIVE
OTTAWA, CANADA K1A 0K2

1 Dr. Jill Larkin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213

Non Govt

1 Dr. Alan Lesgold
Learning R&D Center
University of Pittsburgh
Pittsburgh, PA 15260

1 Dr. Robert A. Levit
Manager, Behavioral Sciences
The BDM Corporation
7915 Jones Branch Drive
McClean, VA 22101

1 Dr. Robert Linn
College of Education
University of Illinois
Urbana, IL 61801

1 Dr. Robert R. Mackie
Human Factors Research, Inc.
6780 Cortona Drive
Santa Barbara Research Pk.
Goleta, CA 93017

1 Dr. Mark Miller
Systems and Information Sciences Laborat
Central Research Laboratories
TEXAS INSTRUMENTS, INC.
Mail Station 5
Post Office Box 5936
Dallas, TX 75222

1 Dr. Richard B. Millward
Dept. of Psychology
Hunter Lab.
Brown University
Providence, RI 02912

1 Dr. Stuart Milner
Department of Education
George Mason University
4400 Fairfax Drive
Fairfax, VA 22030

1 Richard T. Mowday
College of Business Administration
University of Oregon
Eugene, OR 97403

Non Govt

1 Dr. Allen Munro
Univ. of So. California
Behavioral Technology Labs
3717 South Hope Street
Los Angeles, CA 90007

1 Dr. Donald A Norman
Dept. of Psychology C-009
Univ. of California, San Diego
La Jolla, CA 92093

1 Dr. Seymour A. Papert
Massachusetts Institute of Technology
Artificial Intelligence Lab
545 Technology Square
Cambridge, MA 02139

1 Dr. James A. Paulson
Portland State University
P.O. Box 751
Portland, OR 97207

1 MR. LUIGI PETRULLO
2431 N. EDGEWOOD STREET
ARLINGTON, VA 22207

1 Dr. Barbara Pflanz
Department of German
University of Redlands
Redlands, CA 92373

1 DR. PETER POLSON
DEPT. OF PSYCHOLOGY
UNIVERSITY OF COLORADO
BOULDER, CO 80302

1 DR. DIANE M. RAMSEY-KLEE
R-K RESEARCH & SYSTEM DESIGN
3947 RIDGEMONT DRIVE
MALIBU, CA 90265

1 Dr. Peter D. Read
Social Science Research Council
605 Third Avenue
New York, NY 10016

Non Govt

1 Dr. Mark D. Reckase
Educational Psychology Dept.
University of Missouri-Columbia
12 Hill Hall
Columbia, MO 65201

1 Dr. Fred Reif
SESAME
c/o Physics Department
University of California
Berkely, CA 94720

1 Dr. Leonard L. Rosenbaum, Chairman
Department of Psychology
Montgomery College
Rockville, MD 20850

1 Dr. Ernst Z. Rothkopf
Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974

1 Dr. David Rumelhart
Center for Human Information Processing
Univ. of California, San Diego
La Jolla, CA 92093

1 Dr. Allen Schoenfeld
Department of Mathematics
Hamilton College
Clinton, NY 13323

1 DR. ROBERT J. SEIDEL
INSTRUCTIONAL TECHNOLOGY GROUP
HUMRRO
300 N. WASHINGTON ST.
ALEXANDRIA, VA 22314

1 Dr. Robert Smith
Department of Computer Science
Rutgers University
New Brunswick, NJ 08903

1 Dr. Richard Snow
School of Education
Stanford University
Stanford, CA 94305

Non Govt

1 Dr. Robert Sternberg
Dept. of Psychology
Yale University
Box 11A, Yale Station
New Haven, CT 06520

1 DR. ALBERT STEVENS
BOLT BERANEK & NEWMAN, INC.
50 MOULTON STREET
CAMBRIDGE, MA 02138

1 Mr. William Stobie
McDonnell-Douglas
Astronautics Co.
P. O. Box 30204
Chico, CA 95926

1 Mr. D. J. Sullivan
c/o Canyon Research Group, Inc.
741 Lakefield Road
Westlake Village, CA 91361

1 DR. PATRICK SUPPES
INSTITUTE FOR MATHEMATICAL STUDIES IN
THE SOCIAL SCIENCES
STANFORD UNIVERSITY
STANFORD, CA 94305

1 Dr. Kikumi Tatsuoka
Computer Based Education Research
Laboratory
252 Engineering Research Laboratory
University of Illinois
Urbana, IL 61801

1 Dr. John Thomas
IBM Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

1 DR. PERRY THORNDYKE
THE RAND CORPORATION
1700 MAIN STREET
SANTA MONICA, CA 90406

Non Govt

1 Dr. Walt W. Tornow
Control Data Corporation
Corporate Personnel Research
P.O. Box O - HQN060
Minneapolis, MN 55440

1 Dr. Douglas Towne
Univ. of So. California
Behavioral Technology Labs
3717 South Hope Street
Los Angeles, CA 90007

1 Dr. J. Uhlaner
Perceptronics, Inc.
6271 Variel Avenue
Woodland Hills, CA 91364

1 Dr. Benton J. Underwood
Dept. of Psychology
Northwestern University
Evanston, IL 60201

1 Dr. Willard S. Vaughan, Jr.
Oceanautics, Inc.
422 Sixth Street
Annapolis, MD 21403

1 Dr. Phyllis Weaver
Graduate School of Education
Harvard University
200 Larsen Hall, Appian Way
Cambridge, MA 02138

1 Dr. David J. Weiss
N660 Elliott Hall
University of Minnesota
75 E. River Road
Minneapolis, MN 55455

1 Dr. Karl Zinn
Center for research on Learning
and Teaching
University of Michigan
Ann Arbor, MI 48104