The Epistemology of a Rule-Based Expert System
—a Framework for Explanation

William J. Clancey

Heuristic Programming Project, Computer Science Department, Stanford University, Stanford, CA 94305, U.S.A.

Recommended by John Seely Brown

ABSTRACT

Production rules are a popular representation for encoding heuristic knowledge in programs for scientific and medical problem solving. However, experience with one of these programs, MYCIN, indicates that the representation has serious limitations: people other than the original rule authors find it difficult to modify the rule set, and the rules are unsuitable for use in other settings, such as for application to teaching. These problems are rooted in fundamental limitations in MYCIN's original rule representation: the view that expert knowledge can be encoded as a uniform, weakly structured set of if/then associations is found to be wanting.

To illustrate these problems, this paper examines MYCIN's rules from the perspective of a teacher trying to justify them and to convey a problem-solving approach. We discover that individual rules play different roles, have different kinds of justifications, and are constructed using different rationales for the ordering and choice of premise clauses. This design knowledge, consisting of structural and strategic concepts which lie outside the representation, is shown to be procedurally embedded in the rules. Moreover, because the data/hypothesis associations are themselves a proceduralized form of underlying disease models, they can only be supported by appealing to this deeper level of knowledge. Making explicit this structural, strategic and support knowledge enhances the ability to understand and modify the system.

1. Introduction

Production rules are a popular representation for capturing heuristics, 'rules of thumb,' in *expert systems*. These artificial intelligence programs are designed to provide expert-level consultative advice in scientific (CRYSALIS [18], RI [22]) and medical problem solving (EXPERT [23], VM [19]). MYCIN [15, 24] is generally acknowledged to be one of the forerunners of this research. Shortliffe's and Buchanan's original design goal was to use a simple, uniform formalism, which was easy to modify, to encode heuristics [16]. They cited the importance of

providing reasonable explanations, as well as good advice, for the program to be acceptable to users. This approach was in stark contrast to the common Bayesian programs which did not seek to capture an expert's reasoning steps, and so were not easily understood by clients [25]. MYCIN set the standards for much subsequent research.

The success of MYCIN as a problem solver in infectious disease diagnosis, as shown in several formal evaluations [33, 34], suggested that the program's knowledge base might be a suitable source of subject material for teaching students. This use of MYCIN was consistent with the design goals that the program's explanations be educational to naive users, and that the representation be flexible enough to allow for use of the rules outside the consultative setting. In theory, the rules acquired from human experts would be understandable and useful to students. The GUIDON program [6, 7, 8] was developed to push these assumptions by using the rules in a tutorial interaction with medical students.

In attempting to 'transfer back' the experts' knowledge through GUIDON, we find that the experts' diagnostic approach and understanding of rules have not been explicitly represented. GUIDON cannot justify the rules because MYCIN does not have an encoding of how the concepts in a rule fit together. GUIDON cannot fully articulate MYCIN's problem solving approach because the structure of the search space and the strategy for traversing it are implicit in the ordering of rule concepts. Thus, the seemingly straightforward task of converting a know-ledge based-system into a computer-aided instruction program has led to a detailed re-examination of the rule base and the foundations upon which rules are constructed, an epistemological study.

In building MYCIN, rule authors did not recognize a need to record the structured way in which they were fitting rule parts together. The rules are more than simple associations between data and hypotheses. Sometimes clause order counts for everything (and the order can mean different things), and some rules are present for effect, to control the invocation of others. The uniformity of the representation obscures these various functions of clauses and rules. In looking beyond the surface of the rule representation to make explicit the intent of the rule authors, this paper has a purpose similar to Wood's "What's in a link?" [32] and Brachman's "What's in a concept?" [2]. We ask, "What's in a rule?"

The demands of tutoring provide a 'forcing function' for articulating the structure of the rule base and the limitations of the program's explanation behavior. These insights have implications for generating and explaining con-sultative advice, and modifying the system. For the moment, consider that many, if not most or all, of the issues of tutoring must be addressed in providing explanations to naive users. Consider how an expert can violate a rule in difficult, 'non-standard' situations, because he can reason about the rule's justification. Finally, consider the difficulties of changing a program whose design is not fully documented.

In building GUIDON, we thought that we were simply being 'applications engineers' by making use of MYCIN's explanation facility for a tutorial setting. It was surprising to find out how little the explanation facility could accomplish for a student. Without a crisp characterization of what we expected an 'explanation' to convey, the program was of questionable value. On the positive side, the study of these shortcomings led to a radical change in our conception of MYCIN's rules, and supplied a new epistemological framework for building expert systems, the subject of this paper.

Subsequent sections provide a review of MYCIN's explanatory capability (Section 2), and an overview of an epistemological framework for enhancing that capability (Section 3). The following two sections examine in detail the problems of *justifying a rule* and *explaining an approach*, thereby elucidating the *support* and *strategic* aspects of the epistemological framework. In Section 6 implications for performance of a consultation system and modifiability are considered briefly. Finally, in Section 7 the framework is used to analyze a variety of other expert systems.

2. MYCIN's Explanation Capability

As already mentioned, MYCIN was designed so that it could explain its reasoning to the user. This means that it can describe its reasoning steps: how a request for data is related to a goal, how one goal leads to another, and how a goal is achieved. Fig. 1 shows a consultation excerpt in which the program demonstrates a line of reasoning (chained goals) that lead backwards from the current question: Is the patient a compromised host (rule 343) →[1] is the patient immunosuppressed? (rule 395) → is the patient receiving steroids? (rule 543) → is e.coli causing the meningitis? Fig. 2 illustrates how, in the questioning session after the consultation, one can inquire further about the program's intermediate reasoning steps, including why it didn't ask about something. These are the explanation capabilities that we sought to exploit in a teaching program.

MYCIN's explanations are entirely in terms of its rules and goals. The question "WHY?" means "How is this information useful?", translated internally as "In what rule does this goal appear, and what goal does the rule conclude about?" Davis, who developed the explanation facility (Fig. 1), pointed out that MYCIN did not have the knowledge to respond to other interpretations of a "WHY" question [11]. He mentioned specifically the lack of rule justifications and planning knowledge which we have studied and address in this paper.

In order to illustrate other meanings for the question "Why?" in MYCIN, we illustrate the rule set as a network of goals, rules and hypotheses (Fig. 3). At the top level are all of the system's goals that it might want to pursue to solve a problem (diagnostic and therapeutic decisions). Examples of goals, stated as

---

[1] Read: "...compromised host (by rule 343) answers the question..."

23) Is J. Smith a compromised host?
**WHY*

[i.e. WHY is it important to determine whether J. Smith is a compromised host?]
[3.0] This will aid in determining whether J. Smith is immunosuppressed.

    If

       [3.1] *J. Smith is not a compromised host*
    then
       it is definite (1.0) that J. Smith is not immunosuppressed
    [RULE343]

  [back to question 23 . . .]
**WHY*

[i.e. WHY is it important to determine whether J. Smith is immunosuppressed?]
[4.0] . . . in order to determine whether J. Smith has received corticosteroids.

    If

       [4.1] *J. Smith is not immunosuppressed*
    then
       it is definite (1.0) that J. Smith has not received corticosteroids.
    [RULE395]

  [back to question 23 . . .]
**WHY*

[i.e. WHY is it important to determine whether J. Smith has received corticosteroids?]
[5.0] . . . in order to determine the organisms causing the infection.

    It has already been established that

       [5.1] *the infection which requires therapy is meningitis, and*
       [5.2] *only circumstantial evidence is available for this case, and*
       [5.3] *the type of meningitis is bacterial.*
    Therefore, if
       [5.4] *J. Smith is receiving corticosteroids*
    then
       there is evidence that the organisms causing the infection are:
       e.coli (.4), klebsiella-pneumoniae (.2), or pseudomonas-aeruginosa (.1)
    [RULE543]

  [back to question 23 . . .]
**NO*

FIG. 1. Consultation excerpt showing explanation capability. (User input in bold italics; all other text is printed by MYCIN.)

**DID YOU CONSIDER WHETHER THE PATIENT WAS RECEIVING STEROIDS?*
RULE543 could have used:
  whether J. Smith has received corticosteroids.
Clause 4 of RULE543 {the patient has received corticosteroids} was already known to be false.

**HOW DID YOU KNOW THAT THE PATIENT HAD NOT RECEIVED STEROIDS?*
RULE395 was used to conclude that J. Smith has not received corticosteroids. The last question asked before the conclusion was made was 23.

**WHAT WAS QUESTION 23?*
Question 23 was asked in order to find out whether J. Smith is a compromised host in an effort to execute RULE343.

FIG. 2. Excerpt from question/answer session. (User input appears in bold italics.)
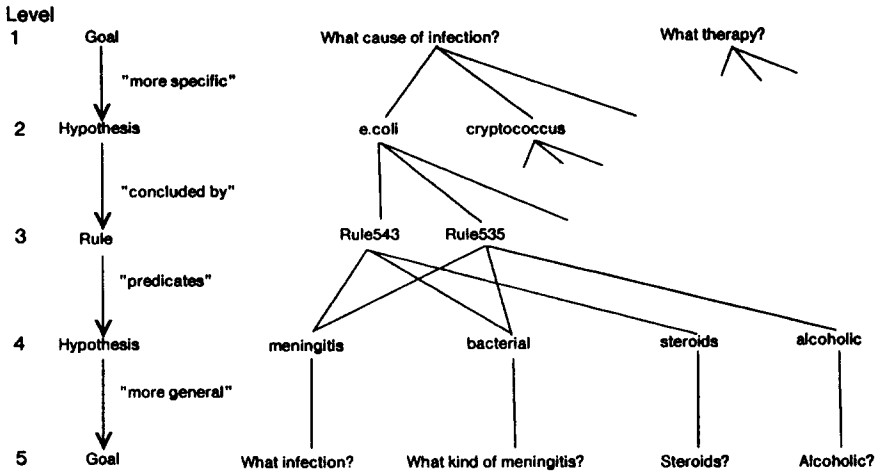
FIG. 3. Rule set shown as a network linking hypotheses and goals.

questions to answer, are: "What is the shape of the organism?" "What organism is causing the meningitis?" At the second level are hypotheses or possible choices for each of the goals. Examples of hypotheses are: "The organism is a rod." "E.coli is causing the meningitis." At the third level are the rules that support each hypothesis (this is a graph because a rule might support more than one hypothesis). At the fourth level appear the premises of these rules—specific hypotheses that must be believed for the rule to apply (multiple rules might predicate the same hypothesis). For example, for rule 543 to apply, it must be the case that the infection is meningitis, that the meningitis be caused by bacteria, that the patient be receiving steroids, and so on.

A key aspect of MYCIN's interpreter is that, when confronted with a hypothesis in a rule premise that it needs to confirm, it considers all related hypotheses by pursuing the more general goal. For example, attempting to apply rule 543, the program will consider all rules that conclude about the infection, rather than just those that conclude that the infection is meningitis. Similarly, it will consider all rules that conclude about the kind of meningitis (viral? fungal? tb? bacterial?), rather than just those that hypothesize that the meningitis is bacterial.[2] These new goals deriving from rules can now be seen conceptually as level 1 goals and the process recurs.

---

[2] This is not inefficient, given the program's exhaustive search strategy and the fact that the other hypotheses will be referenced by other rules. Note also that some hypotheses, such as "the patient is receiving steroids", are not generalized, but represented as goals directly. Whether or not a hypothesis is represented as a 'yes/no parameter' or as a 'value' of 'multi-valued parameter' (such as 'kind of meningitis') is a rule-author decision, deriving from a pattern of hypotheses that he wishes to collapse for clarity into a more general goal. By this process of abstraction, a single 'multi-valued parameter' dealing with kinds of surgery would replace individual 'yes/no parameters' that specified 'cardiac surgery', 'neurosurgery', etc. These organizational decisions have no bearing on system performance, so the knowledge base is somewhat inconsistent in how these choices are made.

The links in Fig. 3 and their ordering are points of flexibility in the rule representation. For example, the rule author defines each goal and its specific hypotheses (levels 1–2 and 4–5). Less trivially, it is the rule author's choice to define rules that link hypotheses to one another (rules on level 3 link levels 2 and 4). We call the rationale behind this link the *justification of the rule*. GUIDON cannot teach rule justifications because they are not represented in MYCIN. Section 4 examines the nature of rule justifications and how a tutoring system can provide them.

Next, by the rule author's ordering of hypotheses in a rule's premise, he will affect the order in which goals are pursued (level 5). The rationale for this choice again lies outside of the rule network. Thus, the program cannot explain why it pursues meningitis (goal 5.1 in Fig. 1) before determining that the infection is bacterial (goal 5.3). Section 5 examines how this ordering constitutes a strategy and how it can be made explicit.

The order in which rules for a goal are tried (level 3) also affects the order in which hypotheses and hence subgoals are pursued (level 5). For example, rule 535 considers whether the patient is an alcoholic, so if this rule is tried before rule 543, alcoholism will be considered before steroids. As these goals lead to asking questions of the user, it is evident that the ordering of questions, when it derives from rule order as opposed to clause order, is also determined by the ordering of rules.

Here there is no implicit author rationale, for rule order lies outside of his choice; it is fixed, but determined only by the order that rules were entered into the system. As pointed out above, MYCIN does not decide to pursue the hypothesis 'bacterial meningitis' before 'viral meningitis'—it simply picks up the bag of rules that make some conclusion about 'kind of meningitis' and tries them in numeric order. Hence, rule order is the answer to the question "Why is one hypothesis (for a given goal) considered before another?" And rule order is often the answer to "Why is one question asked before another?" Focusing on a hypothesis and choosing a question to confirm a hypothesis are not necessarily arbitrary in human reasoning, raising serious questions about using MYCIN for interpreting a student's behavior and teaching him how to reason, also discussed in Section 5.

To summarize, we have used a rule network as a device for illustrating aspects of MYCIN's behavior which it cannot explain. We are especially interested in making explicit the knowledge that lies behind the behavior that is not arbitrary, but which cannot be explained because it is implicit in rule design, particularly the nature of the rule link between hypotheses and the ordering of hypotheses in rule premises. To do this, we will need some sort of framework for characterizing the knowledge involved, since the rule link itself is not sufficient. An epistemological framework for understanding MYCIN's rules is presented in the next section.

3. An Epistemological Framework for Rule-based Systems

The framework presented in this section stems from an extensive study of MYCIN's rules. It is the basic framework that we have used for understanding physicians' explanations of their reasoning, as well as being a foundation for re-representing the knowledge in MYCIN's rules. As an illustration, we will consider in detail the 'steroids rule' (Fig. 4).[3]

Fig. 5 shows how this diagnostic heuristic is justified and incorporated in a problem-solving approach by relating it to strategic, structural, and support knowledge. Recalling Fig. 3, we use the term *strategy* to refer to a plan by which goals and hypotheses are ordered in problem solving. A decision to determine 'cause of the infection' before 'therapy to administer' is a strategic decision. Similarly, a decision to pursue the hypothesis "e.coli is causing meningitis" before "cryptococcus is causing meningitis" is strategic. And recalling an earlier example, deliberately deciding to ask the user about steroids before alcoholism would be a strategic decision. These decisions all lie above the plane of goals and hypotheses, and as shown in Section 5, they can often be stated in domain-independent terms. "Consider differential-broadening factors" is an example of a domain-independent statement of strategy.

In order to make contact with the knowledge of the domain, a level of *structural knowledge* is necessary. Structural knowledge consists of abstractions that are used to index the domain knowledge. For example, one can classify causes of disease into common and unusual causes, as there are common and unusual causes of bacterial meningitis. These concepts provide a *handle* by which a strategy can be applied, a means of referencing the domain-specific knowledge. For example, a strategy might specify to consider common causes of a disease; the structural knowledge about bacterial meningitis allows this strategy to be instantiated in that context. This conception of structural

RULE543

If   (1) the infection which requires therapy is meningitis,
       (2) only circumstantial evidence is available for this case,
       (3) the type of the infection is bacterial,
       (4) the patient is receiving corticosteroids,
then there is evidence that the organisms which might be causing the infection are e.coli (.4), klebsiella-pneumoniae (.2), or pseudomonas-aeruginosa (.1)

FIG. 4. The steroids rule.

---

[3] The English form of rules stated in this paper has been simplified for readability. Sometimes clauses are omitted. Medical examples are for purposes of illustration only.

knowledge follows directly from Davis' [12] technique of *content-directed invocation* of knowledge sources. A handle is a means of indirect reference and is the key to abstracting reasoning in domain-independent terms. The discussion here (particularly Section 7) elaborates upon the nature of handles and their role in the explanation of reasoning.

The structural knowledge we will be considering is used to index two kinds of hypotheses (as the term was used in Section 2): *problem features* which describe the problem at hand (for example, whether or not the patient is receiving steroids is a problem feature) and *diagnoses* which characterize the cause (disorder or disease) of the observed problem features (for example, acute meningitis is a diagnosis). In general, problem features appear in the premise of diagnostic rules and diagnoses appear in the conclusion. Thus, organizations of problem features and diagnoses provide two ways of indexing rule associations: one can use a strategy that brings certain diagnoses to mind, and consider rules that support those hypotheses; or one can use a strategy that brings certain problem features to mind, gather that information, and draw conclusions (apply rules) in a data-directed way.
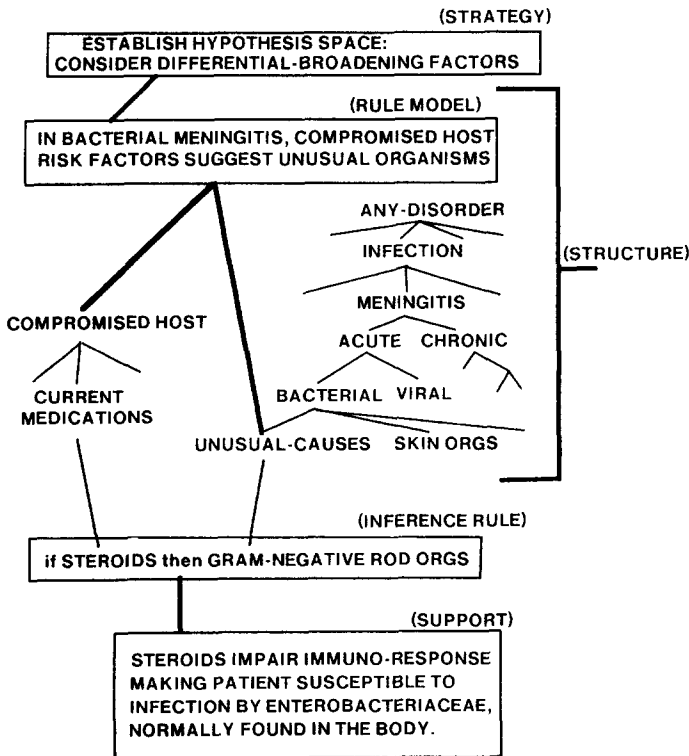

FIG. 5. Knowledge for indexing, justifying, and invoking a MYCIN rule.

Fig. 5 shows how a rule model or *generalized rule*[4], as a form of structural knowledge, enables both data-directed consideration of the steroids rule (via compromised host risk factors) or hypothesis-directed consideration (via unusual causes of meningitis). Illustrated are partial hierarchies of problem features (compromised host factors) and diagnoses (kinds of infections, meningitis, etc.)—a typical form of structural knowledge. The specific organisms of the steroids rule are replaced by the set 'gram-negative rods', a key hierarchical concept we use for understanding this rule.

Finally, the justification of the steroids rule, a link between the problem feature hypothesis "patient is receiving steroids" and diagnostic hypothesis "gram-negative rod organisms are causing acute bacterial infectious meningitis", is based on a causal argument about steroids impairing the body's ability to control organisms that normally reside in the body. While this *support knowledge* is characteristically low-level or narrow in contrast with the strategical justification for considering compromised host risk factors, it still makes interesting contact with structural terms, such as the mention of enterobacteriaceae (kinds of gram-negative rod organisms). In the next section, we will consider the nature of rule justifications in more detail, illustrating how structural knowledge enables us to make sense of a rule by tying it to the underlying causal process.

4. Justifying a Rule

Here we consider the logical bases for rules: What kinds of arguments justify the rules and what is their relation to a mechanistic model of the domain? We use the terms 'explain' and 'justify' synonymously, though the sense of 'making clear what is not understood' (explain) is intended more than 'vindicating, showing to be right or lawful' (justify).

4.1. Different kinds of justifications

There are four kinds of justifications for MYCIN's rules: identification, causal, world fact, and domain fact. In order to explain a rule, it is first necessary to know what kind of justification it is based upon.

Rules that use identifying properties of an object to classify it are called *identification rules*. Most of MYCIN's rules that use laboratory observations of an unknown are like this: "If the organism is a gram-negative, anaerobic rod, its genus may be bacteroides (.6)." Thus, an identification rule is based on the properties of a class.

Rules whose premise and action are related by a causal argument are called *causal rules*. The causality can go in either direction in MYCIN rules: 'symptom

---

[4] Davis' rule models [13], generated automatically, capture patterns, but they do not restate rules more abstractly as we intend here.

caused by disease' or (more commonly) 'prior problem causes disease'. Szolovits [29] suggests that it is possible to subdivide causal rules according to the scientific understanding of the causal link: *empirical association* (a correlation for which the process is not understood), *complication* (direction of causality is known, but the conditions of the process are not understood), and *mechanistic* (process is well-modeled). Most of MYCIN's causal rules represent medical complications that are not easily expressed as anatomical relations and physiological processes.

The certainty factors in MYCIN's causal rules generally represent a mixture of probabilistic and cost/benefit judgment. Rather than simply encoding the strength of association between symptom and cause, a certainty factor also captures how important it is that a diagnosis be considered in therapy selection. Thus, a frequently fatal cause of a disease would be assigned a higher certainty factor than a competing cause that is more likely, but rarely fatal. A later paper will deal with MYCIN's certainty factors in more detail.

Rules that are based on empirical, common sense knowledge about the world are called *world fact rules*. An example is, "if the patient is male, then the patient is not pregnant or breast feeding". Other examples are based on social patterns of behavior, such as the fact that a young male might be a military recruit (and so living in a crowded environment where disease spreads readily).

Domain fact rules link hypotheses on the basis of domain definitions. An example is "if a drug was administered orally and it is poorly absorbed in the GI tract, then the drug was not administered adequately". (By definition, to be administered adequately a drug must be present in the body at high enough dosage levels.) By using domain fact rules, the program can relate problem features to one another, reducing the amount of information it has to request from the user.

In summary, a rule link captures class properties, social and domain facts, and probabilistic and cost/benefit judgments. When a definition, property or world fact is involved, simply saying this provides a reasonable explanation. But causal rules, with their connection to an underlying process of disease, require much more, so we will concentrate on them.

4.2. Levels of explanation—what's not in a rule?

In this section we will consider the problem of justifying a causal rule, the tetracycline rule, "If the patient is less than 8 years old, don't prescribe tetracycline". This rule simply states one of the things that MYCIN needs to know to properly prescribe drugs for youngsters. The rule does not mention the underlying causal process (chelation—drug deposition in developing bones) and the social ramifications (blackened permanent teeth) upon which it is based. From this example, it should be clear that the justifications of MYCIN's rules lie outside of the rule base. In other words, the record of inference steps

that ties premise to action has been left out. A few issues need to be raised here: Did the expert really leave out steps of reasoning? What is a justification for? And what is a good justification?

Frequently we refer to rules like MYCIN's as 'compiled knowledge'. However, when we ask physicians to justify rules that they believe and follow, they very often can't explain why they are correct, or rationales are so long in coming and so tentative, it is clear that we are not being told reasoning steps that are consciously followed. Leaps from data to conclusion are justified because the intermediate steps (like the process of chelation and the social ramifications) generally remain the same from problem to problem. There is no need to step through this knowledge—to express it conditionally in rules. Thus, for the most part, MYCIN's rules are not compiled in the sense that they represent a deliberate composition of reasoning steps by the rule authors. They are compiled in the sense that they are optimizations that leave out unnecessary steps—evolved patterns of reasoning that cope with the demands of ordinary problems.

If an expert does not think about the reasoning steps that justify a rule, why does a student need to be told about them? One simple reason why a student needs a justification for a rule is so that he can remember the rule. A justification can even serve as memory aid (mnemonic) without being an accurate description of the underlying phenomena. For example, medical students have long been told to think in terms of "bacteria eating glucose" from which they can remember that low CSF glucose is a sign of a bacterial (as opposed to fungal or viral) meningitis. The interpretative rule is learned by analogy to a familiar association (glucose is a food and bacteria are analogous to large organisms that eat food). This explanation has been discredited by biological research, but it is still a useful mnemonic.

Given that an accurate causal argument is usually expected, how is a satisfying explanation constructed? To see the difficulty here, observe that, in expanding a rule, there is seemingly no limit to the details that might be included. Imagine expanding the tetracycline rule by introducing three intermediate concepts, as shown in Fig. 6.

The choice of 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? There are levels of detail in a causal explanation. To explain a rule, we not only need to know the intermediate steps, we need to

tetracycline in youngster
⇒ chelation of the drug in growing bones
⇒ teeth discoloration
⇒ undesirable body change
⇒ don't administer tetracycline

FIG. 6. Expansion of tetracycline rule.

decide which steps in the reasoning need to be explained. Purpose (how deep an understanding is desirable) and prior knowledge are obviously important.

Conceptually, the support knowledge for a causal rule is a tree of rules, where each node is a reasoning step that can theoretically be justified in terms of finer-grained steps. The important thing to remember is that MYCIN is a flat system of rules. It can only state its immediate reasoning steps, and cannot explain them on *any* level of detail.

4.3. Problem features, the hypothesis taxonomy, and rule generalizations

A tree of rules seems unwieldy. Surely most teachers cannot expand upon every reasoning step down to the level of the most detailed physical knowledge known to man. (The explanation tree for the tetracycline rule quickly gets into chemical bonding theory.) Explaining a rule (or understanding one) does not require that every detail of causality be considered. Instead, a relatively high level of explanation is generally satisfying—you probably felt satisfied by the explanation that tetracycline causes teeth discoloration. This level of 'satisfaction' has something to do with the student's prior knowledge.

For an explanation to be satisfying, it must make contact with already known concepts. We can characterize explanations by studying the kinds of intermediate concepts they use. For example, it is significant that most contraindication rules, reasons for not giving antibiotics, will refer to 'undesirable body changes'. This pattern is illustrated hierarchically in Fig. 7. (The first level gives types of undesirable changes; the second level gives causes of these types.) Notice that Fig. 7 contains the last step of the expanded tetracycline rule, and a leap from tetracycline to this step. The pattern connecting drugs to the idea of undesirable body changes forms the basis of an expectation for explanations: we will be satisfied if a particular explanation connects to this pattern. In other words, given an effect that we can interpret as an undesirable body change, we will understand why a drug causing that effect should not be given. We might want to know how the effect occurs, but here again, we will rest easy on islands of familiarity, just as we don't feel compelled to ask why people don't want black teeth.

To summarize, key concepts in rule explanations are abstractions that connect to a pattern of reasoning we have encountered before, premises that
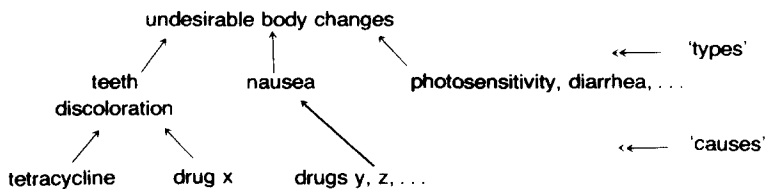

FIG. 7. Problem feature hierarchy for contraindication rules.

we readily accept. This suggests that one way to explain a rule, to make contact with a familiar reasoning pattern, is to generalize the rule. We can see this more clearly from the viewpoint of diagnosis, which makes rich use of hierarchical abstractions.

Consider the rule fragment, "If a complete blood count is available and the white blood count is less than 2.5 units, then the following bacteria might be causing infection: e.coli (.75), pseudomonas-aeruginosa (.5), klebsiella-pneumoniae (.5)". How can we explain this rule? First, we generalize the rule (Fig. 8).

The premise concepts in the rules on levels 1 through 3 are problem features (recall discussion in Section 3), organized hierarchically by different kinds of relations. Generally, a physician speaks loosely about the connections—referring to 'leukopenia' both as a cause of immunosuppression as well as a kind of immunosuppression—probably because the various causes are thought of hierarchically. The similarity of this part of the diagram to Fig. 7 should be evident.

The relationships among CBC, WBC and leukopenia reveal some interesting facts about how MYCIN's rules are constructed. WBC is one component of a 'complete blood count' (CBC). If the CBC is not available, it makes no sense to ask for any of the components. Thus, the CBC clause in the leukopenia rule is an example of a *screening clause*. Another example of a screening clause is the age clause in the rule fragment, "if . . . age is greater than 17 and the patient is an alcoholic, then . . .". Here the relation is a social fact; if the patent is not an adult, we assume that he is not an alcoholic. The third relation we observe is
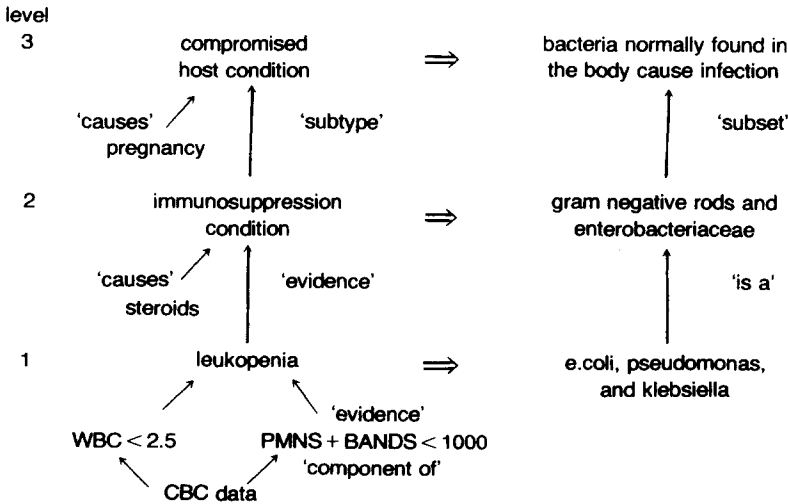

FIG. 8. Generalizations of the leukopenia rule.

subtype, as in "if . . . the patient has undergone surgery and the patient has undergone neurosurgery, then . . .". All screening relations can be expressed as rules, and some are, such as "if the patient has not undergone surgery, then he has not undergone cardiac surgery" (stated negatively, as is procedurally useful). MYCIN's rule set is inconsistent in this respect; to be economical and make the relationship between clauses explicit, all screening clauses should be expressed as world fact rules or hierarchies of parameters. Indeed, the age/alcoholic relation suggests that some of the relations are uncertain and should be modified by certainty factors.

Viewed as a semantic network representation, MYCIN's rules are links without names. Even when rules explicitly link problem features, (as a rule could state "if no CBC was taken, then WBC is not available"), the *kind of relation* is not represented (WBC is a *component of* a CBC) because MYCIN's rule language does not allow the link to be labeled. Finally, when one problem feature serves as a redefinition of another, such as the relation between leukopenia and WBC, the more abstract problem feature tends to be left out altogether ('leukopenia' is not a MYCIN parameter, the rule mentions WBC directly, another manifestation of knowledge compilation). For purposes of explanation, we argue that problem features, their relations, and the nature of the link should be explicit.

Returning to Fig. 8, the action concepts, *diagnostic hypotheses*, are part of a large hierarchy of causes that the problem solver will cite in the final diagnosis. The links in this diagnosis space generally specify refinement of cause, though in our example they strictly designate subclasses. Generally, problem features are abstractions of patient states indicated by the observable symptoms, while the diagnosis space is made up of abstractions of causal processes that produce the symptoms. Paralleling our observations about rule problem features, we note that the relations among diagnostic hypotheses are not represented in MYCIN—nowhere in the knowledge base does it explicitly state that e.coli is a bacterium.

Now suppose that the knowledge in Fig. 8 were available, how would this help us to explain the leukopenia rule? The idea is that we first restate the rule on a higher level. We point out that a low WBC indicates leukopenia, which is a form of immunosuppression, thus tying the rule to the familiar pattern that implicates gram-negative rods and enterobacteriaceae. This is directly analogous to pointing out that tetracycline causes teeth discoloration, which is a form of undesirable body change, suggesting that the drug should not be given.

By rerepresenting Fig. 8 linearly, we see that it is an expansion of the original rule:

WBC < 2.5 ⇒ leukopenia ⇒ immunosuppression ⇒ compromised host ⇒ infection by organisms found in body ⇒ gram-negative rods and enterobacteriaceae ⇒ e.coli, pseudomonas and klebsiella.

The expansion marches up the problem feature hierarchy and then back down the hierarchy of diagnoses. The links of this expansion involve causality composed with identification, subtype and subset relations. By the hierarchical relationships, a rule on one level 'explains' the rule below it. For example, the rule on level 3 provides the detail that links immunosuppression to the gram-negative rods. By generalizing we have made a connection to familiar concepts.

Tabular rules provide an interesting special case. The CSF protein rule (Fig. 9) appears to be quite formidable.

Graphing this rule (Fig. 10), we find a relatively simple relation that an expert stated as, "if the protein value is less than 40, I think of viral; if it is more than 100, I think of bacterial, fungal or tb". This is the first level of generalization, the principle that is implicit in the rule. The second level elicited from the expert is, "if the protein value is low, I think of an acute process; if it is high, I think of a severe or long term process".[5] Then, at the highest level, he states, "an infection in the meninges stimulates protein production". So in moving up abstraction hierarchies on both the premise and action side of the rule (acute and chronic are subtypes of infection), we arrive at a mnemonic, just like "bacteria eat glucose". Abstractions of both the observations and the conclusions are important for understanding the rule.

We might be surprised that explanations of rules provide levels of *detail* by referring to more *general* concepts. We are accustomed to the fact that principled theoretical explanations of, say, chemical phenomena, refer to atomic properties, finer-grained levels of causality. Why should a rule explana-

RULE500

If   (1) the infection which requires therapy is meningitis,
        (2) a lumbar puncture has been performed on the patient, and
        (3) the CSF protein is known
then the type of the infection is as follows:
        If the CSF protein is:
        (a) less than 41 then:
            not bacterial (.5), viral (.7), not fungal (.6), not tb (.5);
        (b) between 41 and 100 then:
            bacterial (.1), viral (.4), fungal (.1);
        (c) between 100 and 200 then:
            bacterial (.3), fungal (.3), tb (.3);
        (d) between 200 and 300 then:
            bacterial (.4), not viral (.5), fungal (.4), tb (.4);
        (e) greater or equal to 300 then:
            bacterial (.4), not viral (.6), fungal (.4), tb (.4).

FIG. 9. The CSF protein rule.

[5] Bacterial meningitis is a severe, acute (short term) problem, while fungal and tb meningitis are problems of long (chronic) duration.
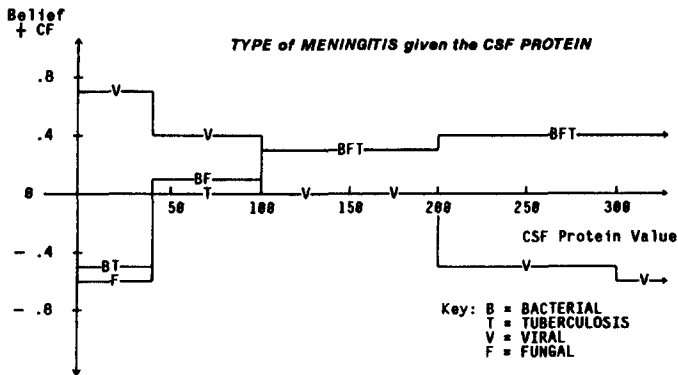
FIG. 10. Graph of the conclusions made by rule 500 (Fig. 9).

tion refer to concepts like 'compromised host' or 'organisms normally found in the body'? The reason is that in trying to understand a rule like the steroids rule, we are first trying to relate it to our understanding of what an infection is at a high, almost metaphorical level. In fact, there are lower level 'molecular' details of the mechanism that could be explained, for example, how steroids actually change the immunological system. But our first goal as understanders, our focus, is at the top level—to link the problem feature (steroids) to the global process of meningitis infection. We ask, "What makes it happen? What role do steroids play in the infectious meningitis process?"

The concept of 'compromised host' is a label for a poorly understood causal pattern that has value because we can relate it to our understanding of the infectious process. It enables us to relate the steroids or WBC evidence to the familiar metaphor in which infection is a war that is fought by the body against invading organisms. (If a patient is compromised, his defenses are down; he is vulnerable to attack.) In general, causal rules argue that some kind of process has occurred. We expect a top-level explanation of a causal rule to relate the premise of the rule to our most general idea of the process being explained. This provides a constraint for how the rule should be generalized, the subject of the next section.

4.4. Tying an explanation to a causal model

MYCIN's diagnostic rules are arguments that a process has occurred in a particular way, namely, that an infectious process has transpired in the patient along certain lines. There are many kinds of infections, which have different characteristics, but bacterial infections tend to follow the same script: entry of an organism into the body, passage of the organism to the site of infection, reproduction of the organism, and causing of observable symptoms. An explanation of a rule that concludes that an organism is causing an infection

must demonstrate that this *generic process* has occurred. In short, this is the level of abstraction that the explanation must connect to.

A program was written to demonstrate this idea. The data parameters in MYCIN's 40 diagnostic rules for bacterial meningitis are restated as one or more of the steps of the infectious process script. This restatement is then printed as the explanation of the rule. For example, the program's explanation of the alcoholic rule (alcoholism ⇒ diplococcus meningitis) is:

> The fact that the patient is an alcoholic *allows access of organisms from* the throat and mouth *to* lungs (by reaspiration of secretions).

> The fact that the patient is an alcoholic *means that the patient is a compromised host, and so susceptible to infection.*

Words in italics (of the first sentence) constitute the pattern of 'portal and passage'. We find that the premise of a rule generally supplies evidence for only a single step of the causal process: the other steps must be inferred by default. For example, the alcoholic rule argues for passage of the diplococcus to the lungs. The person reading this explanation must know that diplococcus is normally found in the mouth and throat of any person and it proceeds from the lungs to the meninges by the blood. The organism finds conditions favorable for growth because the patient is compromised, as stated in the explanation. In contrast, the leukopenia rule only argues for the patient being a compromised host, so the organisms are the default organisms, those already in the body which can proceed to the site of infection.[6]

These explanations say which steps are enabled by the data. They place the patient on the path of an infection, so to speak, and leave it to the understander to fill in the other steps with knowledge of how the body normally works. This is why the physician generally refers to the premise data as 'predisposing factors'. (MYCIN's heuristics are generally of the form "predisposing factor causes disease".) They argue that prior steps in a causal process have occurred. To be understood, a rule must be related to these prior steps, the general concepts that explain many rules.

The process of explanation is a bit more complicated in that a causal relation may exist between clauses in the rule. We have already seen that one clause may screen another on the basis of world facts, multi-component test relations, and the subtype relation. The program described here knows these relations and 'subtracts off' screening clauses from the rule. Moreover, as discussed in Section 5, some clauses describe the context in which the rule applies (the role of the first 3 clauses in the steroids rule, Fig. 4). These, too, are made explicit

---

[6] As might be expected, alcoholism also causes infection by the gram-negative rods and enterobacteriaceae. We have omitted these for simplicity. However, this example illustrates that a MYCIN rule can have multiple conclusions reached by different causal paths.

for the explanation program, and subtracted off. In the vast majority of MYCIN rules, only one premise clause remains, and this is related to the process of infection in the way described above.

When more than one clause remains after screening and contextual clauses have been removed, our study shows that a causal connection exists between the remaining clauses. We can always isolate one piece of evidence that the rule is about (for example, WBC in the leukopenia rule); we call this the *key factor* of the rule. We call the remaining clauses *restriction clauses* and observe three kinds of relations between a restriction clause and a key factor.

(1) *A confirmed diagnosis explains a symptom.* (Example: a petechial rash would normally be evidence for neisseria, but if the patient has leukemia, it may be the disease causing the rash. Therefore the rule states, "if the patient has a petechial rash (the key factor) and does not have leukemia (the restriction clause), then neisseria may be causing the meningitis".)

(2) *Two symptoms in combination suggest a different diagnosis than one taken alone.* (Example: When both purpuric and petechial rashes occur, then a virus is a more likely cause than neisseria. Therefore, the petechial rule also includes the restriction clause "the patient does not have a purpuric rash".)

(3) *Weak circumstantial evidence is made irrelevant by strong circumstantial evidence.* (Example: a head injury so strongly predisposes a patient to infection by skin organisms that the age of the patient, a weak circumstantial factor, is made irrelevant.)

Restriction clauses are easy to detect when examining the rule set because they are usually stated negatively ("some problem feature or diagnosis does not apply"). These examples are suggestive of the causal reasoning about problem features that we find in diagnosis when the causality of the system is better understood, as in electronic troubleshooting.

In summary, to explain a causal rule, a teacher must know the purposes of the clauses and connect the rule to abstractions in the relevant process script.

4.5. The relation of medical heuristics to principles

It might be argued that we have to go to so much trouble to explain MYCIN's rules because they are written on the wrong level. Now that we have a 'theory' for which intermediate parameters to include ('portal', 'pathway', etc.), why don't we simply rewrite the rules?

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. MYCIN's rules are empiric. Cleaning them up by representing problem feature relationships explicitly would give us the same set of rules at a higher level. But what would happen if process concepts were incorporated in completely new reasoning steps, for example, if the rule set related problem features to hypotheses about the pathway the organism took through the body? It turns out that reasoning backwards in terms of a causal model is not always

appropriate. As we discovered when explaining the rules, not all of the causal steps of the process can be directly confirmed; we can only assume that they have occurred. For example, rather than providing diagnostic clues, the concept of 'portal of entry and passage' is very often deduced from the diagnosis itself.

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 medical reasoning. Specifically, (1) if a symptom needs to be explained (is highly unusual), we ask what could cause it ("Strep-viridans? It is normally found in the mouth. How did it get to the heart? Has the patient had dental work recently?"); (2) to 'prove' that the diagnosis is correct (after it has been constructed), we use a causal argument ("He has pneumonia; the bacteria obviously got into the blood from the lungs."). Thus, causal knowledge can be used to provide feedback that everything fits.

It may be difficult or impossible to expect a set of 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 mani-festations, but he will be an inefficient problem solver if he always attempts to directly convert that model to a subgoal structure for solving ordinary prob-lems. Szolovits [29] points out that these 'first principles', used by a student, are 'compiled out' of an expert's reasoning.

In meningitis diagnosis, the problem is to manage a broad, if not incoherent, hypothesis set, rather than to pursue a single causal path. The underlying theory recedes to the background, and the expert tends to approach his problem simply in terms of weak 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, for example, the concept of 'leukopenia' described above.

5. Teaching Problem-solving Strategy

A *strategy* is an approach for solving a problem, a plan for ordering methods so that a goal is reached. It is well-accepted that strategic knowledge must be conveyed in teaching diagnostic problem solving. As Brown et al. [3] say:

> "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."

Strategic knowledge is general, much like the principles of mechanism we discussed earlier; both relate to processes that have structure. Thus, it is not

sufficient to merely show a student MYCIN's solution, the *surface structure* of the program, we must explain why the rules are invoked in a particular order.

Here it is clear how *teaching* how to do something differs from merely *explaining* how something was done: we want the student to be able to replicate what he observes, to go off and solve similar problems on his own. This is why providing mnemonics is useful when justifying a rule. Regarding strategy, we must again address human foibles and preference: we must teach a strategy that a human can follow.

The main points of this section are:
– MYCIN's strategy is different from a physician's strategy.
– MYCIN's strategic knowledge is embedded in the rules, indistinguishable from screening and problem feature clauses.
– A domain-independent representation of strategy is useful for teaching and for purposes of economy.

5.1. Surface and deep structure of MYCIN

A person trying to understand a MYCIN consultation observes that pieces of data are requested by the program (Fig. 11). Conceptually, these questions are terminals hanging below an AND (rule) node in a subgoal tree (one portion corresponding to Fig. 11 is shown in Fig. 12). Observe that this example, considering the order of questions asked by the program, is a special case of the rule network shown earlier (Fig. 3)—here we show only the hypotheses (Fig. 3, level 4) that lead directly to a question. Following the terminology of Brown et al. [3], a rule node is a *method* for achieving a goal (e.g., "organisms

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

FIG. 11. MYCIN consultation excerpt.

| GOAL | HYPOTHESIS | METHOD | QUESTION |

```
                    HYPOTHESIS      METHOD              QUESTION

                ┌── E.COLI ──────── (Rule511) ── Q32 NEUROSURGERY

                ├── NEISSERIA ── (Rule533) ── Q33 CROWD

                │                ┌── (Rule535) ── Q34 ALCOHOLIC
                ├── D.PNEUMO ──┤
                │                └── (Rule559) ── Q38 SPLENECTOMY

COVERFOR ──────┤
                │                ┌── (Rule545) ── Q35 NOSOCOMIAL
                ├── H.INFLU ───┤                        ┌── Q36 EPIGLOTTITIS
                │                └── (Rule395) ──┤
                │                                 └── Q37 OTITIS-MEDIA
                │
                └── PSEUDOMO. ── (Rule578) ── Q39 BURN
```
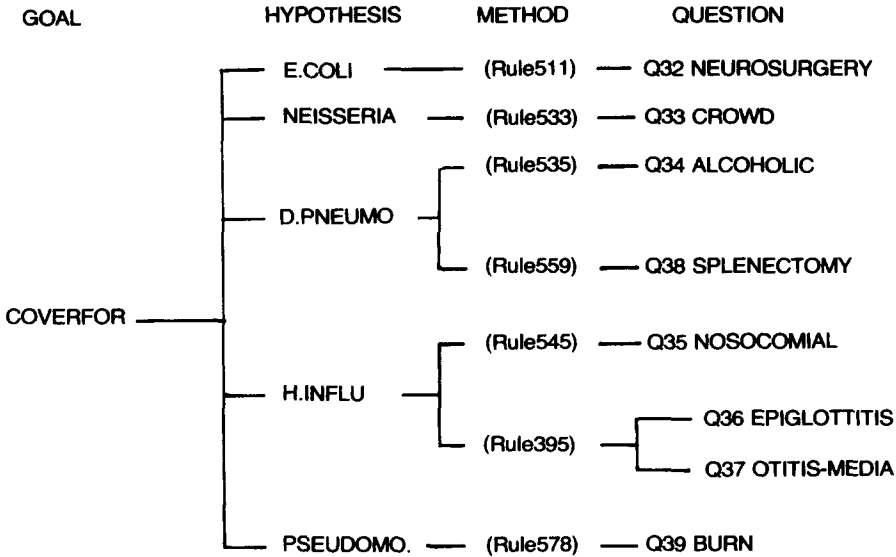
FIG. 12. Portion of the AND/OR tree corresponding to the questions shown in Fig. 11 (reorganized according to the hypothesis each rule supports).

that might be causing the infection") by asking questions or pursuing a set of subgoals to achieve (the premise of a particular rule). Therefore, the tree of rules and subgoals is part of a deep-structured trace that Brown et al. postulate is constructed when the understander makes sense of the surface problem solution.

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 (that is, have schemas for kinds of problems that can be tackled using different *approaches* or lines of reasoning.) A plan sets up a rational sequence of applications of methods that might get you closer to the solution (though this is not guaranteed).

The hypothetico-deductive strategy used in medical problem solving constitutes a plan for focusing on hypotheses and selecting confirmatory questions [17]. However, the methods selected in Fig. 12 (rules 511 through 578) have been applied in a fixed, arbitrary order—not planned by the rule author. MYCIN has no 'deep structure' plan at this level; the program is simply applying rules (methods) exhaustively. This lack of similarity to human reasoning severely limits the usefulness of the system for teaching problem-solving.

However, MYCIN does have a problem solving strategy above the level of rule

. application, namely the control knowledge that causes it to pursue a goal at a certain point in the diagnosis. We can see this by examining how rules interact in backward chaining. Fig. 13 shows the goal rule and a rule that it indirectly invokes.

In order to evaluate the third clause of the goal rule, MYCIN tries each of the 'cover for' rules; the alcoholic rule is one of these (see also Fig. 12). We call the goal rule a *task rule* to distinguish it from inference rules. Clause order counts here; this is a procedure, not a logical conjunction. MYCIN has a few other task rules, but procedural knowledge appears in almost every rule. The first three clauses of the alcoholic rule, the *context clauses*, really control the order in which goals are pursued, just as in a task rule.

We can represent this hidden structure of goals by a tree which we call the *inference structure* of the rule base (produced by 'hanging' the rule set from the goal rule). Fig. 14 illustrates part of MYCIN's inference structure.[7]

The first three goals on the third level appear in a single task rule, while the last two goals (shown as specific hypotheses 'meningitis?' and bacterial?') correspond to the first and third clauses of the 40 'cover for' rules similar to the alcoholic rule.

The program's strategy comes to light when we list these goals in the order in which the depth-first interpreter makes a final decision about them. For example, since at least one rule that concludes 'significant' (goal 4 in Fig. 14) mentions 'contaminant' (goal 3), MYCIN applies *all* of the 'contaminant' rules

RULE092 (the goal rule)

If   (1) gather information about cultures taken from the patient and therapy he is receiving,
      (2) determine if the organisms growing on cultures require therapy,
      (3) consider circumstantial evidence for additional organisms that therapy should cover,
then determine the best therapy recommendation.

RULE535 (the alcoholic rule)

If   (1) the infection which requires therapy is meningitis,
      (2) only circumstantial evidence is available for this case,
      (3) the type of meningitis is bacterial,
      (4) the age of the patient is greater than 17 years, and
      (5) the patient is an alcoholic,
then there is evidence that the organisms which might be causing the infection are diplococcus-
     pneumoniae (.3) or e.coli (.2).

FIG. 13. The goal rule and alcoholic rule.

---

[7]Some definitions of terms used in the following discussion: TREATFOR = organisms to be treated, based on direct laboratory observation; COVERFOR = organisms to be treated, based on circumstantial evidence; SIGNIFICANT = this organism merits therapeutic attention, based on the patient's degree of sickness and validity of culture results; CONTAMINANT = the finding of this organism is spurious; it was probably introduced during sampling from the cultured site of the body (as a blood culture might include skin organisms).

before making a final decision about 'significant'. Analyzing the entire rule set in a similar way gives us the ordering (numbers correspond to the numbering in Fig. 14):

(1) Is there an infection?
(2) Is it bacteremia, cystitis, or meningitis?
(3) Any contaminated cultures?
(4) Any good cultures with significant growth?
(5) Is the organism identity known?
(6) Is there an infection? (already done in Step 1)
(7) Does the patient have meningitis? (already done in Step 2)
(8) Is it bacterial?
(9) Are there specific bacteria to cover for?

(We leave out the goals 'regimen' and 'treatfor' because they are just place holders for task rules, like subroutine names.) MYCIN's diagnostic plan is in two parts, and both proceed by top-down refinement. This demonstrates that a combination of structural knowledge (the diagnosis space taxonomy—infection, meningitis, bacterial, diplococcus . . .) and strategic knowledge (traversing the taxonomy from the top down) is procedurally embedded in the rules. In other words, we could write a program that interpreted an explicit, declarative representation of the diagnosis taxonomy and domain-independent form of the strategy to bring about the same effect.

At this level, MYCIN's diagnostic strategy is not a complete model of how physicians' think, but it could be useful to a student. As the quote from Brown et al. [3] would indicate, and has been confirmed in GUIDON research, teachers do articulate both the structure of the problem space and the nature of the
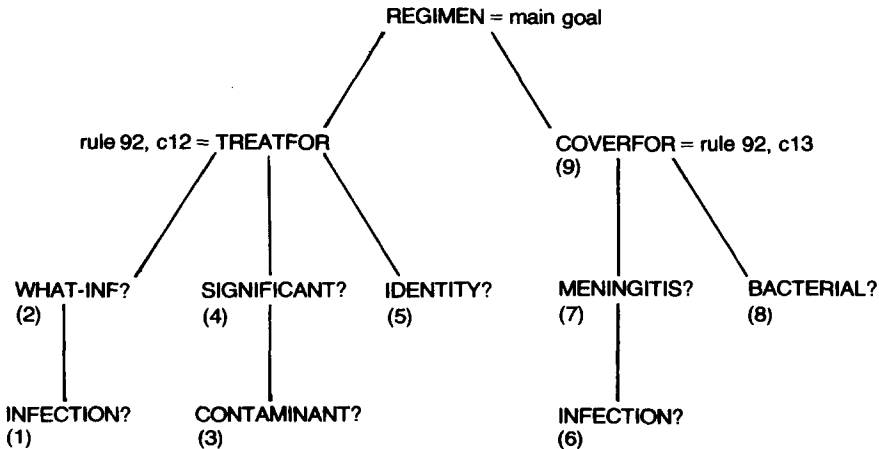

FIG. 14. Portion of MYCIN's inference structure (numbers give the order in which non-place-holder goals are achieved by the depth-first interpreter).

search strategy to students. This means that we need to explicitly represent the fact that the diagnosis space is hierarchical, and represent strategy in a domain-independent form. If the strategy is not in domain-independent form, it can be taught by examples, but not explained.

5.2. Representing strategic knowledge in meta-rules

How might we represent domain-independent strategic knowledge in a rule-based system? In the context of the MYCIN system, Davis pursued the representation of strategic knowledge by using *meta-rules* to order and prune methods. 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 in Fig. 15. Observe that this is a *strategy* for pursuing a goal. 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; in this simple way, the rule sets up a line of reasoning. This mechanism causes some goals to be pursued before others, orders the questions asked by the system, and hence changes the surface structure of the consultation.

While meta-rules like this can capture and implement strategic knowledge about a domain, they have their deficiencies. For like the MYCIN performance rules we have examined, Davis's domain-dependent examples of meta-rules leave out knowledge important for explanation. Not only do they, like the object-level rules, leave out the domain-specific support knowledge that justifies the rules, they leave out the domain-independent strategic principles that GUIDON should teach. In short, meta-rules provide the *mechanism* for controlling the use of rules, but not the domain-independent *language* for making the strategy explicit.

The implicit strategic principle that lies behind meta-rule 001 is that common (frequent) causes of a disorder should be considered first. The structural knowledge that ties this strategy to the object-level diagnostic rules is an explicit partitioning of the diagnosis space taxonomy, indicating that the group of organisms called the enterobacteriaceae are more likely than gram-positive rod organisms to cause pelvic infections. This is what we want to teach the student. One can imagine, for different infection types, different common causes, requiring a different meta-rule for each infection. But if all meta-rules

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 gram-pos-rods,
then there is suggestive evidence (.4) that the former should be done before the latter.

FIG. 15. A MYCIN meta-rule.

are as specific as meta-rule 001, principles will be compiled into many rules redundantly and the teaching points will be lost.

What does a domain-independent meta-rule look like, and how is it interfaced with the object-level rules? To explore this question, we have reconfigured the MYCIN rule base into a new system, called NEOMYCIN [9]. Briefly, meta-rules are organized hierarchically (again!) into tasks, such as "group and refine the hypothesis space". These rules manage a changing hypothesis list by applying different kinds of knowledge sources, as appropriate. Knowledge sources are essentially the object level rules, indexed in the diagnosis space taxonomy by a domain-independent structural language.

For example, one meta-rule for achieving the task of pursuing a hypothesis is "if there are unusual causes, then pursue them".[8] Suppose that the current hypothesis is "bacterial meningitis". The program will use the structural label 'unusual causes' to retrieve the nodes 'gram-negative rods', 'enterobacteriaceae', and 'listeria', add them to the hypothesis list, and pursue them in turn. (When there are no 'unusual causes' indicated the meta-rule simply does not apply.) Pursuing gram-negative rods, the program will find that leukopenia is a relevant factor, but first ask if the patient is a compromised host (Fig. 8), modelling a physician's efficient casting of wider questions.

Other terms in the structural language used by NEOMYCIN's domain independent metarules are: (disease) process features, such as *extent* and *location*; the *enabling step* of a causal process; *subtype*; *cause*; *trigger association*; *problem feature screen*; and structural properties of the taxonomy, such as *sibling*. In effect, the layer of 'structural knowledge' allows us to separate out *what* the heuristic is from *how* it will be used. How domain-specific heuristics (like MYCIN's rules) should be properly integrated with procedural, strategic knowledge is an issue at the heart of the 'declarative/procedural controversy' [30]. We conclude here that, for purposes of teaching, the hierarchies of problem features and the diagnosis space should be represented explicitly, providing a useful means for indexing the heuristics by both their premise and action (Fig. 5). A structural language of cause, class, and process connects this domain-specific knowledge to domain-independent meta-rules, the strategy for problem solving.

5.3. Self-referencing rules

Self-referencing rules provide an interesting special example of how problem solving strategies can be embedded in MYCIN's rules. A rule is self-referencing if

---

[8]This rule appears after the rule for considering common causes, and the ordering is marked as strategically significant. Domain-independent meta-rules have justifications, organization, and strategies for using them. Their justification refers to properties of the search space and the processor's capabilities.

RULE086

If   (1) the aerobicity of the organism is not known, and
        (2) the culture was obtained more than 2 days ago,
then there is evidence that the aerobicity of the organism is obligate-aerob (.5) or facultative (.5).

FIG. 16. The self-referencing aerobicity rule.

the goal concluded by the action is also mentioned in the premise. An example is the 'aerobicity rule' (Fig. 16).[9]

This rule is tried only after all of the non-self-referencing rules have been applied. The cumulative conclusion of the non-self-referencing rules is held aside, then the self-referencing rules are tried, using in each rule the reserved conclusion. Thus, the first clause of rule 86 will be true only if none of the non-self-referencing rules made a conclusion. The effect is to reconsider a tentative conclusion. When the original conclusion is changed by the self-referencing rules, this is a form of non-monotonic reasoning [32]. We can restate MYCIN's self-referencing rules in domain-independent terms.

- *If nothing has been observed, consider situations that have no visible manifestations.* (Example: The aerobicity rule—"if no organism is growing the culture, it may be an organism that takes a long time to grow (obligate-aerob and facultative organisms)".) The self-referencing mechanism makes it possible to state this rule without requiring a long premise that is logically exclusive from the remainder of the rule set.
- *If unable to make a deduction, assume the most probable situation.* (Example: "if the gram stain is unknown and the organism is a coccus, then assume that it is gram-positive.")
- *If there is evidence for two hypotheses, A and B, that tend to be confused, then rule out B.* (Example: "if there is evidence for tb and fungal, and you have hard data for fungal, rule out tb".)

Like Meta-rule 001, self-referencing rules provide a useful mechanism for controlling the use of knowledge, but they leave out both the domain-dependent justification and the general, domain-independent reasoning strategy of which they are examples. These rules illustrate that strategy involves more than a search plan; it also takes in principles for reasoning about evidence, what Collins calls 'plausible reasoning' [10]. It is not clear that a teacher needs to explicitly state these principles to a student. They tend to be either 'common sense' or almost impossible to think about independently of an example. Nevertheless, they are yet another example of strategic knowledge that is implicit in MYCIN's rules.

---

[9] Aerobicity refers to whether an organism can grow in the presence of oxygen. A facultative organism can grow with or without it; a non-aerobic organism cannot grow with oxygen present; and an obligate-aerob is aerobic only in a certain stage of growth.

6. Implications for Modifiability and Performance

MYCIN's rule authors believed that the program achieved good problem-solving performance without having to reason about the structural, strategic, and support knowledge we have been considering. However, it is possible to imagine situations in which knowledge of justification and strategy allows one to be a more flexible problem solver, to cope with novel situations. MYCIN cannot solve some kinds of difficult problems, the problems we say only an expert can solve.

Knowing the basis of a rule allows you to know when not to apply it, or how to modify it for special circumstances. For example, knowing that tetracycline won't kill the patient, but the infection might, you may have to dismiss social ramifications and prescribe the drug. You can deliberately break the rule because you understand the assumptions underlying it.

There will also be problems which cannot be diagnosed using MYCIN's rules. For example, several years ago coccidioides meningitis strangely appeared in the San Francisco Bay Area. We would say that such a case 'violates all the rules'. To explain what was happening, one has to reason about the underlying mechanisms. The organisms were travelling from the San Joaquin Valley to the Bay Area by "freak Southeastern winds", as the newspapers reported. The basic mechanism of disease was not violated, but this time the patients didn't have to travel to the Valley to come in contact with the disease. A human expert can understand this because he can fit the new situation to his model. Examples like these make us realize that AI systems like MYCIN can only perform some of the functions of an expert.

Regarding modifiability, the process of reconfiguring MYCIN's rules in NEOMYCIN's terms required many hours of consultation with the original rule authors in order to unravel the rules. As shown in this paper, the lack of recorded principles for using the representation makes it difficult to interpret the purposes of clauses and rules. The strategy and overall design of the program have to be deduced by drawing diagrams like Fig. 14. Imagine the difficulty any physician new to MYCIN would have modifying the CSF protein table (Fig. 9); clearly he would first need the program to explain why it is correct.

We also need a principled representation to avoid a problem we call *concept broadening*. When intermediate problem abstractions are omitted, use of goals becomes generalized and weakened. This happened in MYCIN as the meaning of 'significance' grew to include both 'evidence of infection' and 'non-contaminated cultures'. (So rules were written in the form "if X then significant disease", rather than "if X then evidence of infection" and "if evidence of infection, then significant disease".) As long as the rule author makes an association between the data and some parameter he wants to influence, it doesn't matter for correct performance that the rule is vague. But vague rules are difficult to understand and modify.

A rule base is built upon and extended like any other program. Extensive documentation and a well-structured design are essential, as in any engineering endeavor. The framework of knowledge types and purposes that we have described would constitute a 'typed' rule language that could make it easier for an expert to organize his thoughts. On the other hand, we must realize that this meta-level analysis may impose an extra burden by turning the expert into a taxonomist of his own knowledge—a task that may require considerable assistance, patience, and tools.

7. Application of the Framework to Other Systems

To illustrate further the idea of the strategy, structure, support framework and to demonstrate its usefulness for explaining how a program reasons, several knowledge-based programs are described in Table 1. For generality, we will call inference associations like MYCIN's rules 'knowledge sources' (KS). This analysis is not concerned with the representational notation used in a program, whether it be frames, production rules, units, and so on. Instead we are trying to establish an understanding of the knowledge contained in the system; what kinds of inferences are made at the KS level, how these KS's are structured explicitly in the system, and how this structure is used by strategies for invoking KS's.

7.1. The character of structural knowledge

One product of this study is a characterization of different ways of structuring KS's for different strategical purposes. In all cases, the effect of the structural knowledge is to provide a handle for separating out what the KS is from when it is to be applied.[10]

The different ways of structuring KS's are summarized here according to the processing rationale:

(a) *Organize KS's hierarchically by hypothesis for consistency in data-directed interpretation.* In DENDRAL, if a functional group is ruled out, more specific members of the family are not considered during forward-directed, preliminary interpretation of spectral peaks. Without this organization of KS's earlier versions of DENDRAL could generate a subgroup as a plausible interpretation, while ruling out a more general form of the subgroup, as if to say, "this is an ethyl ketone but not a ketone". [5]

(b) *Organize KS's hierarchically by hypothesis to eliminate redundant effort in hypothesis-directed refinement.* In DENDRAL, the family trees prevent the exhaustive structure generator from generating subgroups whose more general

---

[10] In this section, the term 'hypothesis' generally refers to a diagnostic or explanatory interpretation made by a KS (in terms of some model), though it can also be a hypothesis that a particular problem feature is present, as in CRYSALIS.

TABLE 1. Relation of strategical, structural, and support meta-knowledge to knowledge sources of expert programs

System	Meta-levels of problem-solving knowledge			
	Strategy	Structure	KS examples	Support
DENDRAL	Domain = chemistry, mass spectrometry analysis			
Buchanan et al. [4]	Aggregation heuristics = build superatms and generate all plausible interstitial structures	Family trees of functional groups (ketones, ethers, etc.)	Identification rules relating functional groups to spectral peaks	Molecular chemistry
HEARSAY II	Domain = speech understanding			
Lesser et al. [21]	Policy KS's control hypoths to generate and thresholds (data-directed)	Hierarchy of interpretation levels with links to KS's	Example: KS's hypothesizing words use syllable level for data	Grammar and identification properties of phonemes, syllables and words
AM	Domain = concept formation, mathematical discovery			
Lenat [20]	Activity heuristics propose tasks (priority agenda and focus heuristic)	Hierarchy of heuristics associated with most general concept/context to which they apply	Rules to create concepts and fill in facets, e.g., heuristics to fill in the 'examples' facet for ANY-RELATION	Theory of interestingness— chiefly based on generalizing and specializing
MOLGEN	Domain = molecular genetics, experiment planning			
Stefik [26]	Determine differences, sketch plan, refine steps (message passing)	Hierarchy of laboratory operation types (used by refinement design operator)	Specific lab techniques: input objects → molecular changes and byproducts	Molecular biology processes
CENTAUR	Domain = medical diagnosis, pulmonary function			
Aikins [1]	Hypothesis-directed, top-down refinement (agenda)	Hierarchy of disease prototypes	Disease component → evidence for prototype	Disease patterns; biological processes
NEOMYCIN	Domain = medical diagnosis, diseases causing neurological symptoms			
Clancey [9]	Grouping and refining hypothesis list (meta-rules focus and pursue)	Multiple hierarchies of etiological processes	Data → evidence for disease process or causal state/category	Same as above

forms have been ruled out. The same principle is basic to most medical diagnosis systems that organize diagnoses in a taxonomy and use a top-down refinement strategy, such as CENTAUR and NEOMYCIN.

(c) *Organize KS's by multiple hypothesis hierarchies for efficient grouping (hypothesis-space splitting).* Besides using the hierarchy of generic disease processes (infectious, cancerous, toxic, traumatic, psychosomatic, etc.), NEOMYCIN groups the same diseases by multiple hierarchies according to disease process features (organ system involved, spread in the system, progression over time, etc.). When hypotheses are under consideration that do not fall into one confirmed subtree of the primary etiological hierarchy, the 'group and differentiate' strategy is invoked to find a process feature dimension along which two or more current hypotheses differ. A question will then be asked, or a hypothesis pursued, to differentiate among the hypotheses on this dimension.

(d) *Organize KS's for each hypothesis on the basis of how KS data relates to the hypothesis, for focusing on problem features.* In NEOMYCIN, additional relations make explicit special kinds on connections between data and hypotheses, such as "this problem feature is the enabling causal step for this diagnostic process", and meta-rules order the selection of questions (invocation of KS's) by indexing them indirectly through these relations ("if an enabling causal step is known for the hypothesis to be confirmed, try to confirm that problem feature"). The meta-rules that reference these different relations ('enabling step', 'trigger', 'most likely manifestation') are ordered arbitrarily. Meta-meta-rules don't order the meta-rules because we currently have no theoretical basis for relating the first order relations to one another.

(e) *Organize KS's into data/hypothesis levels for opportunistic triggering at multiple levels of interpretation.* HEARSAY's blackboard levels (sentence, word sequence, word, etc.) organize KS's by the level of analysis they use for data, each level supplying data for the hypothesis level above it. When new results are posted on a given level, KS's that care about that level of analysis are polled to see if they should be given processing time. *Policy KS's* give coherence to this opportunistic invocation by affecting which levels will be given preference. CRYSALIS [18] (a program that constructs a 3-dimensional crystal structure interpretation of nuclear magnetic resonance data) takes the idea a step further by having multiple planes of blackboards; one abstracts problem features (the 'density plane') and the other abstracts interpretations (the 'model plane').

(f) *Organize KS's into a task hierarchy for planning.* In MOLGEN, laboratory operators are referenced indirectly through tasks that are steps in an abstract plan. For example, the planning level *design decision* to refine the abstract plan step, MERGE, is accomplished by indexing laboratory operators by the MERGE task (e.g., MERGE could be refined to using a ligase to connect DNA structures, mixing solutions, or causing a vector to be absorbed by an organism). Thus, tasks in planning are analogous to hypotheses in interpretation problems.

(g) *Organize KS's into a context specialization hierarchy for determining task relevance.* In AM, relevant heuristics for a task (typically filling in a mathematical concept slot) are inherited from all contexts (concepts) that appear above it in the specialization hierarchy. Thus, AM goes a step beyond most other systems by showing that policy KS's must be selected on the basis of the kind of problem being solved. Lenat's work suggests that this might be simply a hierarchical relationship among kinds of problems.

The above characterizations of different organizations for knowledge are a first step towards a vocabulary or language for talking about indirect reference of KS's. It is clear that strategy and structure are intimately related; to make this clearer, we return to our original interest in explanation.

Teaching a strategy might boil down to saying, "think in terms of such-and-such a structural vocabulary in order to get this strategical task done"—where the vocabulary is the indexing scheme for calling KS's to mind. So we might say, "think in terms of families of functional subgroups in order to rule out interpretations of the spectral peaks". Or, "consider process features when diseases of different etiologies are possible". That is, teaching a strategy involves in part the *teaching of a perspective for relating KS's hierarchically* (e.g., 'families of functional subgroups' or 'disease process features'), and then *showing how these relations provide leverage for managing a large amount of data or number of hypotheses.* The explanation of the sought-after leverage must be in terms of some task for carrying the problem forward, thus tying the structuring scheme to the overall process of what the problem solver is trying to do (thus we say, "to rule out interpretations" or "to narrow down the problem to one etiological process' or (recalling Fig. 5) "to broaden the spectrum of possibilities"). In this way, we give the student a meta-rule that specifies what kind of vocabulary or 'cut on the problem' to consider for a given strategical task.

Davis's study of meta-rules [14] suggested a need for a vocabulary of meta-rule knowledge. His examples suggested just a few conceptual primitives for describing refinement (ordering and utility of KS's) and a few primitives for describing object-level knowledge (KS input and output). All of the strategies in our examples deal with ordering and utility criteria for KS's; so we have nothing to add there. All of the examples given here reference KS's by the data they act upon, the hypotheses they support or the tasks they accomplish, except for AM, which references KS's by their scope or domain of applicability. What is novel about the analysis here is the focus on *relations* among hypotheses and among data.

From our domain-independent perspective, strategical knowledge selects KS's on the basis of the causal, subtype, process, or scoping relation they bear to hypotheses or data currently thought to be relevant to the problem at hand. Thus, our meta-rules make statements like: "Consider KS's that would demonstrate a prior cause for the best hypotheses." "Don't consider KS's that

are subtypes of ruled-out hypotheses." "Consider KS's that abstract known data." "Consider KS's that distinguish between two competing kinds of processes." "Consider KS's relevant to the current problem domain."

To summarize, the structural knowledge we have been studying consists of relations that hierarchically abstract data and hypotheses. These relations constitute the vocabulary by which domain-independent meta-rules invoke KS's. The key to our analysis is our insistence on domain-independent statement of meta-rules—a motivation deriving from our interest in explanation and teaching.

7.2. Explicitness of strategical knowledge

Another consideration for explanation is whether or not the strategy for invoking KS's is explicit or encoded indirectly. To an extent system designers are not generally interested in representing high-level strategies that are always in effect and never reasoned about by the program. Instead system designers are satisfied if their system can be programmed in the primitives of their representation language to bring about the high-level effect they are seeking. For example, top-down refinement is 'compiled into' CENTAUR's hierarchy itself by the control steps that specify on each level what to do next (e.g., "after confirming obstructive airways disease, determine the subtype of obstructive airways disease"). By separating control steps from disease inferences, Aikins's improved the explanation facility, one of the goals of CENTAUR. However, the rationale for these control steps is not represented—it is just as implicit as it was in PUFF's contextual clauses. In contrast, NEOMYCIN's 'explore and refine' task clearly implements top-down refinement through domain-independent meta-rules. However, these meta-rules are ordered to give preference to siblings before descendents—an example of an implicit strategy.

One common way of selecting KS's is on the basis of numerical measures of priority, utility, interestingness, etc. For example CENTAUR, like many medical programs, will first request the data that gives the most weight for the disease under consideration. Thus, the weight given to a KS is another form of indexing by which a strategy can be applied. If we wish to explain these weights, we should ideally replace them by descriptors that 'generate' them, and then have the strategy give preference to KS's having certain descriptors. NEOMYCIN's meta-rules for requesting data (described above) are a step in this direction.

MOLGEN's 'least-commitment' meta-strategy is a good example of implicit encoding by priority assignment. The ordering of tasks specified by least commitment is: "look first for differences, then use them to sketch out an abstract plan, and finally refine that plan . . .". This ordering of tasks is implicit in the numerical priorities that Stefik has assigned to design operators (e.g., propose-goal, refine-object, find-features). Therefore, an explanation system

for MOLGEN could not explain the least-commitment strategy, but only say that the program performed one task before another because its priority was higher.

7.3. Absence of support knowledge

We have little to say about support knowledge in these systems because none of them represent it. That is, the causal or mathematical models, statistical studies, or world knowledge that justifies the KS's is not used during reasoning. As discussed in Section 6, this limitation calls into question the problem-solving flexibility or 'creativeness' of these programs. In any case, the knowledge is not available for explanation.

7.4. Summary

The strategy/structure/support framework can be applied to any knowledge-based system by asking the questions: What are the KS's in the system (what kinds of recognition or construction operations are performed)? How are the KS's labeled or organized (by data/constraint or hypothesis/operation)? Is this indexing used by the interpreter or by explicit strategical KS's, or is it just an aid for the knowledge engineer? What theoretical considerations justify the KS's? Is this knowledge represented? With this kind of analysis, it should be clear how the knowledge represented needs to be augmented or decomposed, if an explanation facility is to be built for the system. Quite possibly, as in MYCIN, the representational notation will need to be modified as well.

8. Conclusions

The production rule formalism is often chosen by expert system designers because it is thought to provide a perspicuous, modular representation. But we have discovered that there are points of flexibility in the representation that can be easily exploited to embed structural and strategic knowledge (in task rules, context clauses and screening clauses). Arguing from a teacher's perspective, we showed that hierarchies of problem features and diagnoses (allowing rules to be generalized), in addition to domain-independent statement of strategy, are useful to justify a rule and teach an approach for using it. Also, when the rule is causal, satisfying explanations generalize the rule in terms of an underlying process model. This same knowledge should be made explicit for purposes of explanation after a consultation, ease of modification, and potential improvement of problem solving ability.

Characterizing knowledge in three categories, we concluded that MYCIN's rules were used like a programming language to embed strategic and structural principles. However, while context and screening clauses are devices that don't precisely capture the paths of expert reasoning, the basic connection between data and hypothesis is a psychologically valid association. As such, the 'core

rules' represent the experts' knowledge of causal processes in proceduralized form (again, not necessarily compiled into this form, but compiled with respect to causal models which may be incomplete or never even learned). For this reason, support knowledge needs to be represented in a form that is somewhat redundant to the diagnostic associations, while structure and strategy can be directly factored out and represented declaratively.

The lessons of this study apply to other knowledge-based programs, including those which don't use the production rule representation. The first moral is that one cannot simply slap an interactive front end onto a good AI program and expect to have an adequate teaching system. Similarly, an explanation system may have to do more than just read back reasoning steps and recognize questions: it may be useful to abstract the reasoning steps, relating them to domain models and problem-solving strategies.[11]

Other knowledge bases could be studied as artifacts to evaluate the expressiveness of their representation. Is the design of the inference structure explicit? Can it be reasoned about and used for explanation? You must ask: where are the choice points in the representation and what principles for their use have not been represented explicitly? For production systems one should ask: What is the purpose of each clause in the rule and why are clauses ordered this way? Why is this link between premise and conclusion justified? Under what circumstances does this association come to mind (structure and strategy)?

Finally, future 'knowledge engineering' efforts in which human experts are interviewed and their knowledge codified could benefit from first constructing an epistemology along the lines of the strategy-structure-KS-support distinction, and then, relative to that framework, representing knowledge using their chosen notation (rules, units, etc.). Then, when the system fails to behave properly (whether the purpose is teaching or problem solving), changes to *either* the epistemology or the rules should be entertained. In fact, this is a cyclic process where changes are made to the rules that subtly tear at the framework, and after incorporating a series of changes, a new, better epistemology and revised notation can be arrived at. (So a single MYCIN rule might seem awkward, but a pattern such as 40 rules with the same first 3 clauses suggests the underlying nature of the knowledge.) Thus, a methodology for converging on an adequate epistemology comes in part from constant cycling and re-examining of the entire *system* of rules.

The epistemology that evolved from attempts to reconfigure MYCIN's rules is NEOMYCIN's etiological taxonomy, multiple disease process hierarchies, data that trigger hypotheses, etc., plus the domain-independent task hierarchy of meta-rules. In our use of terms like "problem feature," we have moved very far from the too abstract 'clinical parameter' which did not distinguish between data and

---

[11] Swartout [28] reached the same conclusion when developing an explanation system for the Digitalis Advisor, a non-production rule system. He used an automatic programming approach to build structural and strategic knowledge into the program.

hypotheses. Our epistemology provides an improved basis for interpreting expert reasoning, a valuable foundation for knowledge engineering, as echoed by Swanson [27]:

> "Three aspects of the expert's adaptation are especially important to the design of decision support systems: the generative role of basic principles of pathophysiology, the hierarchical structure of disease knowledge, and the heuristics used in coping with information processing demands."

These categories of knowledge provide a framework for understanding an expert. We ask, "What *kind* of knowledge is he describing?" This framework enables us to focus our questions so that we can separate out detailed descriptions of the expert's causal model from his associations that link symptom to disorder, and his strategies for using this knowledge.

9. Postscript: How the Rule Formalism Helped

Despite the now apparent shortcomings of MYCIN's rule formalism, we must remember that the program was influential because it worked well. The uniformity of representation, so much the cause of the 'missing knowledge' and 'disguised reasoning' described here, was perhaps an important asset. With knowledge so easy to encode, it was perhaps the simple parameterization of the problem that made MYCIN such a success. The program could be built and tested quickly at a time when little was known about the structure and methods of human reasoning. Again demonstrating the importance of the original research, we can treat the knowledge base as a reservoir of expertise, something never before captured in quite this way, and use it to suggest better representations.

REFERENCES

1. Aikins, J.S., Prototypes and production rules: a knowledge representation for computer consultations, Ph.D. thesis, Computer Science Department, Stanford University, HPP-80-17, STAN-CS-80-814, 1980.
2. Brachman, R.J., What's in a concept: structural foundations for semantic networks, BBN Report No. 3433, 1976.
3. Brown, J.S., Collins, A. and Harris, G., Artificial intelligence and learning strategies, in: O'Neill, Ed., *Learning Strategies* (Academic Press, New York, 1977).

4. Buchanan, B.G., Sutherland, G. and Feigenbaum, E.A., HEURISTIC DENDRAL: a program for generating explanatory hypotheses in organic chemistry, in: Meltzer, B and Michie, D., Eds., *Machine Intelligence* 4 (Edinburgh University Press, Edinburgh, 1969) 209–254.

5. Buchanan, B.G., Sutherland, G. and Feigenbaum, E.A., Rediscovering some problems of artificial intelligence in the context of organic chemistry, in: B. Meltzer and D. Michie (Eds.), *Machine Intelligence* 5 (Edinburgh University Press, Edinburgh, 1970) 253–280.

6. Clancey, W.J., Tutoring rules for guiding a case method dialogue, *Internat. J. Man–Machine Stud.* 11 (1979) 25–49.

7. Clancey, W.J., Transfer of rule-based expertise through a tutorial dialogue, Computer Science Doctoral Dissertation, Stanford University, STAN-CS-769, 1979.

8. Clancey, W.J., Shortliffe, E.H. and Buchanan, B.G., Intelligent computer-aided instruction for medical diagnosis, Proceedings of the Third Annual Symposium on Computer Applications in Medical Care, Silver Spring, MD (1979) 175–183.

9. Clancey, W.J. and Letsinger, R., NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching, Seventh International Joint Conference on Artificial Intelligence (1981) 829–836.

10. Collins, A., Fragments of a theory of human plausible reasoning, TINLAP-2 (1978) 194–201.

11. 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, Stanford, CA, 1976.

12. Davis, R., Generalized procedure calling and content-directed invocation, Proceedings of the Symposium on Artificial Intelligence and Programming Languages, *SIGPLAN* 12(8) (1977) 45–54.

13. Davis, R., Interactive transfer of expertise: acquisition of new inference rules, *Artificial Intelligence* 12 (1979) 121–157.

14. Davis, R., Meta-rules: reasoning about control, *Artificial Intelligence* 15 (1980) 179–222.

15. Davis, R., Buchanan, B. and Shortliffe, E.H., Production rules as a representation for a knowledge-base consultation program, *Artificial Intelligence* 8 (1977) 15–45.

16. Davis, R. and King, J.J., An overview of production systems, in: Elcock, E.W. and Michie, D., Eds., *Machine Intelligence* 8 (Wiley, New York, 1977) 300–332.

17. Elstein, A.S., Shulman, L.S. and Sprafka, S.A., *Medical Problem-solving: An Analysis of Clinical Reasoning* (Harvard University Press, Cambridge, 1978).

18. Engelmore, R. and Terry, A., Structure and function of the CRYSALIS system, Sixth International Joint Conference on Artificial Intelligence (1979) 250–256.

19. Fagan, L.M., Kunz, J.C., Feigenbaum, E.A. and Osborn, J.J., Representation of dynamic clinical knowledge: measurement interpretation in the intensive care unit, Sixth International Joint Conference on Artificial Intelligence (1979) 260–262.

20. Lenat, D.B., AM: an artificial intelligence approach to discovery in mathematics as heuristic search, Ph.D. thesis, Computer Science Department, Stanford University, CS-STAN-76-570, AIM-286, 1976.

21. Lesser, V.R., Fennell, R.D., Erman, L.D. and Reddy, D.R., Organization of the HEARSAY II speech understanding system, *IEEE Trans. Acoust. Speech Signal Process.* 23 (1975) 11–24.

22. McDermott, J., R1: a rule-based configurer of computer systems, Department of Computer Science, Carnegie-Mellon University, CMU-CS-80-119, 1980.

23. Mizoguchi, F., Maruyama, K., Yamada, T., Kitazawa, K. Saito, M. and Kulikowski, C.A., A case study of EXPERT formalism—an approach to a design of medical consultation system through EXPERT formalism, Sixth International Joint Conference on Artificial Intelligence (1979) 583–585.

24. 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 (Elsevier, New York, 1976).

25. Shortliffe, E.H., Buchanan, B.G. and Feigenbaum, E.A., Knowledge engineering for medical

decision making: a review of computer-based clinical decision aids, *Proc. IEEE* **67** (1979) 1207–1224.

26. Stefik, M.J., Planning with constraints, Ph.D. thesis, Computer Science Department, Stanford University, HPP-80-2, STAN-CS-80-784, 1980.
27. Swanson, D.B., Feltovich, P.J. and Johnson, P.E., Psychological analysis of physical expertise: implications for design of decision supprt systems, MEDINF077 (1977) 161–164.
28. Swartout, W.R., Explaining and justifying expert consulting programs, Seventh International Joint Conference on Artificial Intelligence (1981) 815–823.
29. Szolovits, P. and Pauker, S.G., Categorical and probabilistic reasoning in medical diagnosis, *Artificial Intelligence* **11**(1) (1978) 115–144.
30. Winograd, T., Frame Representations and the declarative/procedural controversy, in: D.G. Bobrow and A. Collins, Eds., *Representation and Understanding* (Academic Press, New York, 1975) 185–210.
31. Winograd, T., Extended inference modes in reasoning by computer systems, *Artificial Intelligence* **13** (1980) 5–26.
32. Woods, W.A., What's in a link: foundations for semantic networks, in: D.G. Bobrow and A. Collins, Eds., *Representation and Understanding* (Academic Press, New York, 1975) 35–82.
33. Yu, V.L., Buchanan, B.G., Shortliffe, E.H., Wraith, S.M., Davis, R., Scott, A.C. and Cohen, S.N., Evaluating the performance of a computer-based consultant, *Computer Programs in Biomedicine* **9** (1979) 95–102.
34. Yu, V.L., Fagan, L.M., Wraith, S.M., Clancey, W.J., Scott, A.C., Hannigan, J.F. Blum, R.L., Buchanan, B.G. and Cohen, S.N., Antimicrobial selection by a computer—a blinded evaluation by infectious disease experts, *J. Amer. Medical Assoc.* **242** (1979) 1279–1282.