Int. J. Man–Machine Studies (1984) **20**, 3–19

Strategic explanations for a diagnostic consultation system

DIANE WARNER HASLING, WILLIAM J. CLANCEY AND GLENN RENNELS
Heuristic Programming Project, Computer Science Department, Stanford University, Stanford, California 94305, U.S.A.

This article examines the problem of automatic explanation of reasoning, especially as it relates to expert systems. By *explanation* we mean the ability of a program to discuss what it is doing in some understandable way. We first present a general framework in which to view explanation and review some of the research done in this area. We then focus on the explanation system for NEOMYCIN, a medical consultation program. A consultation program interactively helps a user to solve a problem. Our goal is to have NEOMYCIN explain its problem-solving strategies. An explanation of strategy describes the plan the program is using to reach a solution. Such an explanation is usually concrete, referring to aspects of the current problem situation. Abstract explanations articulate a general principle, which can be applied in different situations; such explanations are useful in teaching and in explaining by analogy. We describe the aspects of NEOMYCIN that make abstract strategic explanations possible—the representation of strategic knowledge explicitly and separately from domain knowledge—and demonstrate how this representation can be used to generate explanations.

1. Introduction

The ability to explain reasoning is usually considered an important component of any expert system. An explanation facility is useful on several levels: it can help knowledge engineers to debug and test the system during development, assure the sophisticated user that the system's knowledge and reasoning process is appropriate, and instruct the naive user or student about the knowledge in the system (Scott, Clancey, Davis & Shortliffe, 1977; Davis, 1976; Swartout, 1981*a*).

The problems in producing explanations can be viewed in a framework of three major considerations: epistemologic issues, user modelling, and rhetoric. This section discusses what we mean by each of these and reviews work done in each area.

1.1. EPISTEMOLOGIC ISSUES

The foundation of any explanation is a model of the knowledge and reasoning process to be explained. The explanation work that we characterize as epistemological is concerned with *the knowledge that is required to solve a problem* and *the aspects of problem-solving behavior that need to be explained.* In attempting to emulate human problem-solving activities [such as electronic trouble-shooting (Brown, Burton & deKleer, 1982)], researchers found that existing models of human reasoning were too limited to support robust problem-solving and explanation. Thus one key aspect of research in this area is the study and formalization of the reasoning process in terms of the structure of knowledge and how it is manipulated. For example, in examining causal rationalizations and explanations, deKleer & Brown (1982) discovered the problems of modelling causal processes precisely so they are powerful enough to solve problems people can solve, as well as intuitive enough for people to understand. Similar

0020-7373/84/010003+17$03.00/0

studies are underway for physics problem-solving (Chi, Feltovich & Glaser, 1981) and medical diagnosis (Patil, Szolovits & Schwartz, 1981; Pople, 1982).

Another aspect of this work is the design of a representation language for formalizing a model of reasoning in a computer system. Shortliffe (1976) and Davis (1976) use a simple framework of goals and inference rules to direct a medical consultation; the translation of these rules constitutes the explanation of the inference procedure. Clancey (1981) explores the issue of representing each type of knowledge separately and explicitly in order to convey it clearly to a student. Swartout (1981*b*) uses domain principles and constraints to produce a "refinement structure" that encodes the reasoning process used in constructing the consultation program. In all cases, the task in designing these systems is to represent knowledge and reasoning in a well-structured formalism that can be used to solve problems (perhaps in compiled form as in Swartout's system) and then examined to justify the program's actions.

1.2. USER MODEL

Given an idea of the knowledge needed to solve the problem and a representational framework, a model of the user can be used as a step in determining what needs to be explained to a particular person. The basic idea is to *generate an explanation that takes into account user knowledge and preferences,* often based on previous user interactions and general *a priori* models of expertise levels. The modelling component produces this picture of the user.

For example, Genesereth (1982) takes the approach of constructing a user plan in the course of an interaction to determine a user's assumptions about a complex consultation program. In ONCOCIN, Langlotz & Shortliffe (1983) are able to highlight significant differences between the user's and system's solutions by first asking the user to solve the problem, a common approach in Intelligent Tutoring Systems. In GUIDON, Clancey (1979) uses an "overlay model", in which the student's knowledge is modelled as a subset of what the expert knows. In BUGGY, Brown & Burton (1980) compiled an exhaustive representation of errors in arithmetic to identify a student's addition and subtraction "bugs".

1.3. RHETORIC

Once the content of an explanation has been determined, there is the question of how to convey this information to the user. Rhetoric is concerned with *stating the explanation so that it will be understandable.* It is here that psychological considerations (for example, the need for occasional review to respect human limitations for assimilating new information) are also examined. In STEAMER (Williams, Hollan & Stevens, 1981), Stevens explores the medium of explanation by using a simulation of a physical device, a steam propulsion plant, to produce graphic explanations supplemented with text. Choosing the appropriate level of detail (that is, pruning the internally generated explanation) has been considered by Swartout (1981*a*) and Wallis & Shortliffe (1982).

Explanations, like all communication, have structural components. For example, BLAH (Weiner, 1980) structures explanations so that they do not appear too complex, taking such things as embedded explanations and focus of attention into account. For TEXT, McKeown (1982) examined rhetorical techniques to create schemas that encode aspects of discourse structure. The system is thus able to describe the same information in different ways for different discourse purposes. In GUIDON, Clancey (1979) developed a set of discourse procedures for case method tutorial interactions. The most trivial form of structure is syntax, a problem all natural language generators must consider. At the opposite extreme some programs can produce multiparagraph text (Mann *et al.*, 1981).

2. Motivation for strategic explanations in NEOMYCIN

2.1. NEOMYCIN AND STRATEGIES

The purpose of NEOMYCIN is to develop a knowledge base that facilitates recognizing and explaining diagnostic strategies (Clancey, 1981). In terms of our framework for explanation, this is an epistemological investigation. The approach has been to model human reasoning, representing control knowledge (the diagnostic procedure) explicitly. By *explicit* we mean that the control knowledge is stated abstractly in rules, rather than embedded in application specific code, and that the control rules are separate from the domain rules.† In contrast to Davis' (1980) use of metarules for refining the invocation of base-level rules, NEOMYCIN's metarules choose among lines of reasoning, as well as among individual productions. Thus the metarules constitute a strategy in NEOMYCIN's problem area of medical diagnosis.

A *strategy* is "a careful plan or method, especially for achieving an end". To *explain* is "to make clear or plain; to give the reason for or cause of".‡ Thus in a *strategic explanation* we are trying to make clear the plans and methods used in reaching a goal, in NEOMYCIN's case, the diagnosis of a medical problem. One could imagine explaining an action in at least two ways. In the first, the specifics of the situation are cited, with the strategy remaining relatively implicit. For example, "I'm asking whether the patient is receiving any medications in order to determine if she's receiving penicillin". In the second approach, the underlying strategy is made explicit; "I'm asking whether the patient is receiving any medications because I'm interested in determining whether she's receiving penicillin. *I ask a general question before a specific one when possible*". This latter example is the kind of strategic explanation we want to generate. The general approach to solving the problem is mentioned, as well as the action taken in a particular situation. Explanations of this type allow the listener to see the larger problem-solving approach and thus to examine, and perhaps learn, the strategy being employed.

Our work is based on the hypothesis that an "understander" must have an idea of the problem-solving process, as well as domain knowledge, in order to understand the solution or solve the problem himself (Brown, Collins & Harris, 1978). Specifically, research in medical education (Elstein, Shulman & Sprafka, 1978; Benbassat & Schiffman, 1976) suggests that we state heuristics for students, teaching them explicitly how to acquire data and form diagnostic hypotheses. Other AI programs have illustrated the importance of strategies in explanations. SHRDLU (Winograd, 1972) is an early program that incorporates history keeping to provide WHY/HOW explanations of procedures used by a "robot" in a simulated BLOCKSWORLD environment. The

† See Clancey (1983*a*) for discussion of how diagnostic procedures can be captured by rules and still not be explicit.

‡ *Webster's New Collegiate Dictionary.*

procedures of this robot are specific to the environment; consequently, abstract explanations such as "I moved the red block *to achieve preconditions of a higher goal*" are not possible. CENTAUR (Aikins, 1980), another medical consultation system, explains its actions in terms of domain-specific operations and diagnostic prototypes. Swartout's (1983*b*) XPLAIN program refers to domain principles—general rules and constraints about the domain—in its explanations. In each of these programs, abstract principles have been instantiated and represented in problem-specific terms.

NEOMYCIN generates strategic explanations from an *abstract* representation of strategy. In contrast with other approaches, this strategic knowledge is completely separate from the domain knowledge. This general strategy is instantiated dynamically as the consultation runs. Thus when the program discusses the problem solution, it is able to state a general approach, as well as how it applies in concrete terms.

2.2. MYCIN'S EXPLANATION CAPABILITY

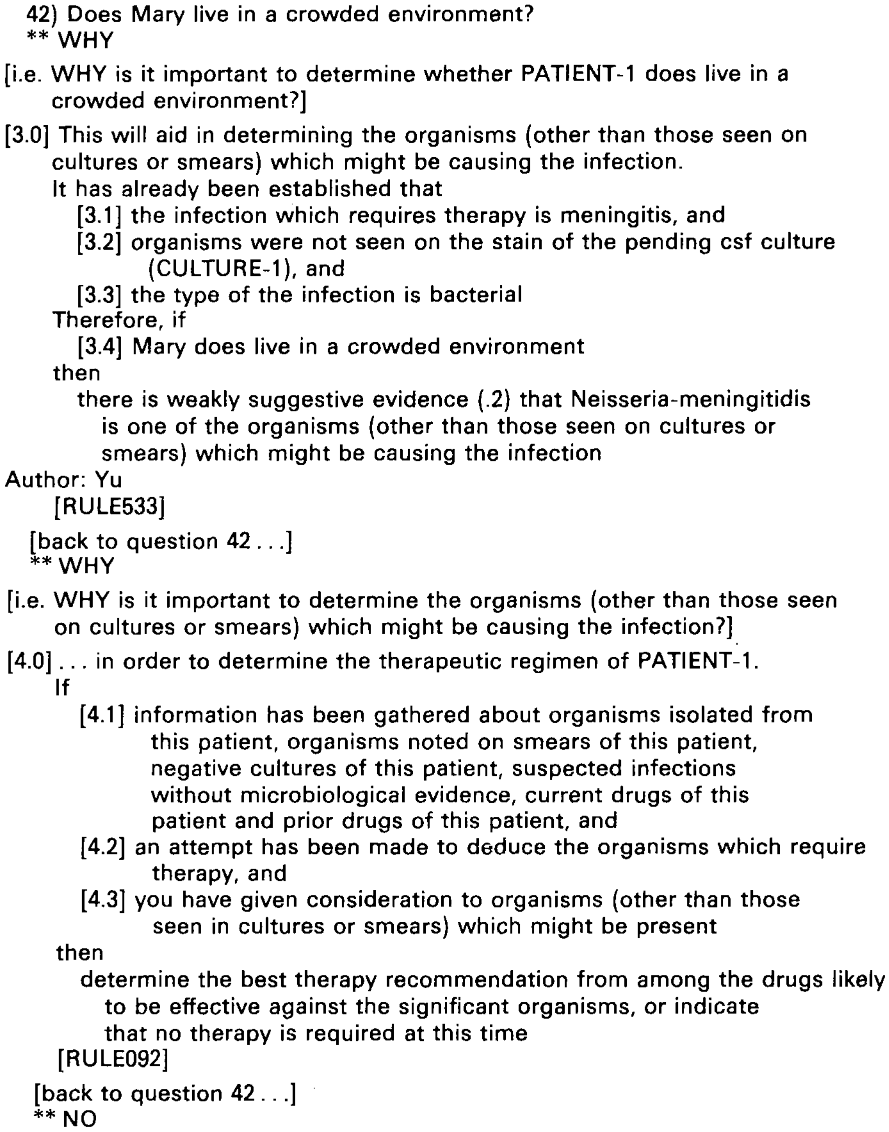
MYCIN is a rule-based medical consultation system designed, in part, so that it could explain its reasoning to a user. A consultation is run by backchaining through the applicable rules, asking questions when necessary. The reasoning employed by the system is explained in terms of this goal chain. There are two basic explanation commands: WHY is used to ascend the chain, discussing subsequently higher sub-goals, while HOW descends the chain explaining how sub-goals were or could be achieved. Figure 1 is an excerpt of a consultation in which the program is explaining its line of reasoning.

Here the current question, "Does the patient live in a crowded environment?" by rule 533 would imply "Is Neisseria-meningitidis causing the meningitis?". This explanation is a good example of the large amount of information embedded in the rules. The clause ordering in the rule implicitly specifies a strategy of top-down refinement—the system is to determine that the patient's infection is meningitis before trying to explore the possibility of bacterial meningitis. This highlights the disease hierarchy also obviously present in the expert's model: meningitis (clause 1) is more general than bacterial meningitis (clause 3), which is the parent of Neisseria-meningitides (the conclusion). The answer to the second WHY illustrates one of MYCIN's "task rules", used to direct the consultation at the highest level. Note the implicit procedure, perhaps apparent to the experienced physician, of gathering initial information, obtaining medical history and physical examination information, and then considering laboratory data.

Although this is a very flexible mechanism which accurately portrays what the system is doing, it has several limitations. For example, in Fig. 1 MYCIN cannot explain why it establishes that the infection is meningitis [3.1] before it determines if the meningitis is bacterial [3.3]. As indicated above, a strategy of top-down refinement of diseases is being followed. Much of the information that went into writing the rules, including justification, ordering, and planning, is either lost or made implicit and thus cannot be explained. This inexplicable information is, in essence, a large part of the strategy employed to do the diagnosis.

2.3. DESIGN CRITERIA

In determining what NEOMYCIN should explain and how it should be explained, we used several design criteria.



```
    42) Does Mary live in a crowded environment?
    ** WHY
[i.e. WHY is it important to determine whether PATIENT-1 does live in a
    crowded environment?]
[3.0] This will aid in determining the organisms (other than those seen on
    cultures or smears) which might be causing the infection.
    It has already been established that
      [3.1] the infection which requires therapy is meningitis, and
      [3.2] organisms were not seen on the stain of the pending csf culture
              (CULTURE-1), and
      [3.3] the type of the infection is bacterial
    Therefore, if
      [3.4] Mary does live in a crowded environment
    then
      there is weakly suggestive evidence (.2) that Neisseria-meningitidis
        is one of the organisms (other than those seen on cultures or
        smears) which might be causing the infection
Author: Yu
    [RULE533]
  [back to question 42...]
  ** WHY
[i.e. WHY is it important to determine the organisms (other than those seen
    on cultures or smears) which might be causing the infection?]
[4.0]... in order to determine the therapeutic regimen of PATIENT-1.
    If
      [4.1] information has been gathered about organisms isolated from
              this patient, organisms noted on smears of this patient,
              negative cultures of this patient, suspected infections
              without microbiological evidence, current drugs of this
              patient and prior drugs of this patient, and
      [4.2] an attempt has been made to deduce the organisms which require
              therapy, and
      [4.3] you have given consideration to organisms (other than those
              seen in cultures or smears) which might be present
    then
      determine the best therapy recommendation from among the drugs likely
        to be effective against the significant organisms, or indicate
        that no therapy is required at this time
    [RULE092]
  [back to question 42...]
  ** NO
```

FIG. 1. Sample MYCIN explanations.

Explanations should *not presuppose any particular user population.* The long-range goal of this research is to use NEOMYCIN as the foundation of a teaching system. At that point the strategic explanations developed here will be used to teach the strategy to students to whom it might be unfamiliar. Techniques used to generate explanations should be flexible enough to accommodate a model of the user.

Explanations should be *informative*; rule numbers or task names are not sufficient.

Explanations should be *concrete or abstract,* depending upon the situation. Thus it must be possible to produce explanations in either form. This should facilitate understanding both of the strategy and how it is actually applied.

Explanations should be *useful for the designer, as well as the end user* of NEOMYCIN. The vocabularies of computer science and an application domain, such as medicine, are different in many ways. People tend to be most comfortable with the vocabulary of their field; the system should have the flexibility to accommodate a user-dependent choice of terminology.

Explanations should be *possible* at the *lowest level of interest*; the "grain level" should be fine enough to permit this. To allow for use in debugging, we chose the level of rules and tasks as our lowest level. Higher level explanations can later be generated by omitting details below the appropriate level.

The following explanation of strategy is an example of how we try to satisfy these criteria in NEOMYCIN. Note how the explanation is abstract, more similar to a MYCIN "task rule" (e.g. rule 92 in Fig. 1) than a domain rule (e.g. rule 533).

```
17) Has Mary been hospitalized recently?
** WHY
[i.e. WHY is it important to determine whether Mary
     has been hospitalized recently?]
[21.0] We are trying to round out the diagnostic
     information by looking generally into past
     medical history and by reviewing systems.
There are unasked general questions that can help us
     with the diagnosis.
```

3. How strategic explanations are possible—The NEOMYCIN system

MYCIN (Shortliffe, 1976), the precursor of NEOMYCIN, is unable to explain its strategy because much of the strategic information is implicit in the ordering of rule clauses (Clancey, 1983*a*). In NEOMYCIN, the problem-solving strategy is both explicit and general. This section provides an overview of the representation of this strategy in NEOMYCIN, since this is the basis for our strategic explanations. Other aspects of the system, such as the disease taxonomy and other structuring of the domain knowledge, are described in Clancey & Letsinger (1981).

NEOMYCIN'S strategy is structured in terms of *tasks*, which correspond to metalevel goals and subgoals, and metalevel rules (*metarules*), which are the methods for achieving these goals. The metarules invoke other tasks, ultimately invoking the base-level interpreter to pursue domain goals or apply domain rules. Figure 2 illustrates a portion of the task structure, with metarules linking the tasks. The entire structure currently includes 30 tasks and 74 metarules. This task structure represents a general diagnostic problem-solving method. Although our base-level for development has been medicine, none of the tasks or metarules mention the medical domain. As a result the strategy might be ported to other domains [see Clancey (1983*b*) for further discussion].

An ordered collection of metarules constitutes a procedure for achieving a task. Each metarule has a premise, which indicates when the metarule is applicable, and an action, indicating what should be done whenever the premise is satisfied. Figure 3 is a high-level abstraction of a task and its metarules. The premise looks in the domain knowledge base or the problem-solving history for findings and hypotheses with certain properties, for example, possible follow-up questions for a recent finding or a subtype

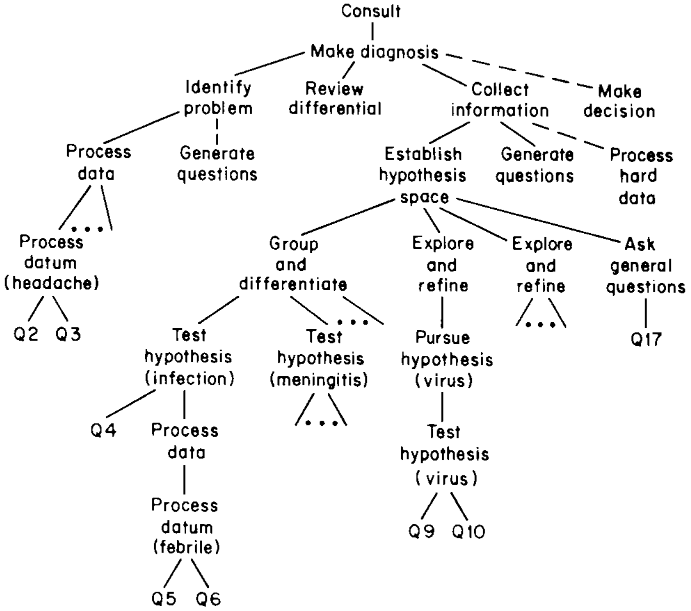


FIG. 2. Invocation of tasks in the example NEOMYCIN consultation. Question numbers correspond to questions asked in the consultation, solids lines show tasks actually done, broken lines those which might be done. Note how such tasks as TEST-HYPOTHESIS are invoked multiple times by a given task as well as by different tasks.

of an active hypothesis. Associated actions would be to ask the user a question or call a task to refine the hypothesis under consideration. The metarules associated with a task may describe the sequence of steps used to achieve the task (in which case the applicable rules are applied once in order), or may present alternate strategies for achieving the goal (in which case the preferentially ordered rules are executed until the goal of the task is achieved).

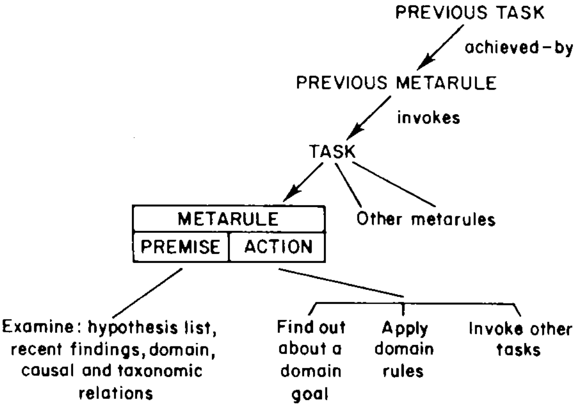


FIG. 3. Abstraction of a,task and its metarules.

Testing an hypothesis is just one reason for asking a question. Others are:

initial identification of the problem:
follow-up questions to specify recent information in greater detail;
questions triggered by recent data that suggested an hypothesis;
general questions to determine the completeness of the case history;
screening questions to determine the relevance of desired data; and
questions asked while pursuing subgoals of the domain.

The number of reasons for asking questions testifies to the complexity of NEOMYCIN's diagnostic strategy and illustrates why strategic explanations are both useful and interesting.

4. NEOMYCIN's strategic explanations

4.1. WHY AND HOW EXPLANATIONS

Excerpts from a NEOMYCIN consultation illustrate the strategic explanation capability most clearly. User input follows the double asterisks (**) and is shown in capital letters. The consultation begins:

```
 Please enter information about the patient.
       Name                      Age      Sex       Race
1) ** MARY                       42       F         LATINO
 Please describe the chief complaints:
 ** HEADACHE
 ** STIFF-NECK
 ** NAUSEA
 **

2) How many days has Mary's headache lasted?
 ** 10

3) How severe is Mary's headache (on a scale of 0 to 4
   with 0 for very mild and 4 for very severe)?
 ** WHY
 [i.e. WHY is it important to determine the severity of
     Mary's headache?]
 [1.0] We are trying to characterize the data just
     obtained and develop the differential.
 Headache severity is a charactizing feature of headache.
 [back to question 3...]
```

To explain WHY something is being done by NEOMYCIN, we first restate the question as "WHY are we trying to do this task?". Because each task is invoked by a metarule in another task (see Fig. 3), we answer the query by stating the task and metarule that invoked the task in question. At the lowest level, there are implicit tasks of requesting data and applying rules. The template for an initial WHY explanation is:

```
⟨request for data⟩
* WHY
[i.e. WHY is it important to determine ⟨data⟩?]
```

```
[1.0] ⟨immediately preceding task⟩
⟨what is true about the domain knowledge base or the
problem-solving history that enables the metarule that
accomplishes this task to succeed⟩
```

In this case the current metarule is:

```
METARULE073
IF there is a datum that can be requested that is a
     characterizing feature of the recent finding
     that is currently being considered,
THEN find out about the datum.
```

We mentioned earlier the idea of providing concrete or abstract explanations as appropriate. In the context of a diagnostic consultation, it seems reasonable to provide concrete explanations when possible. Abstract explanations can always be produced, however. The abstract explanation in this situation is:

```
** ABSTRACT
[i.e. WHY is it important to determine the severity of
     Mary's headache?]
[1.0] We are trying to characterize the data just obtained
     and develop the differential.
A datum that can be requested is a characterizing feature
     of the recent finding that is currently being
     considered.
```

When subsequent WHYs are given, we ascend the strategy tree, as illustrated below.

```
** WHY
[i.e. WHY are we trying to characterize the data just
     obtained and develop the differential?]
[2.0] We are trying to give thorough consideration
     to any recent findings.
The fact that Mary has suffered from headaches recently
     is a new finding, which first must be characterized
     and then used to assist the diagnosis.
[back to question 3...]
** WHY
[i.e. WHY are we trying to give thorough consideration to
     any recent findings?]
[3.0] We are trying to gather identifying information--
     the chief complaint and major problems.
The user must state the problem.
[back to question 3...]
** 3
DIFFERENTIAL:
(VIRUS 564) (CHRONIC-MENINGITIS 422)
```

Questions 4 and 14 below are both asked as a result of the TEST-HYPOTHESIS task. Notice how the explanations of this single task reflect the different situations of the two WHY questions. Different metarules apply—question 4 is based on a strong "triggering" relation, while question 14 is based on only moderate evidence.

```
4) Does Mary have a fever?
** WHY
[i.e. WHY is it important to determine whether Mary has
     a fever?]
[4.0] We are trying to decide whether Mary has an
     infection.
Fever is strongly associated with infection.
[back to question 4...]
** YES
     .
     .
     .
14) Does Mary have a history of granulomas on biopsy of
     the liver, lymph node, or other organ?
** WHY
[i.e. WHY is it important to determine whether Mary has a
     history of granulomas on biopsy of the liver, lymph
     node, or other organ?]
[19.0] We are trying to decide whether Mary has tuberculosis.
A history of granulomas on biopsy of the liver, lymph node,
     or other organ makes tuberculosis more likely.
[back to question 14...]
```

HOW is asked in terms of a previously mentioned task. The explanation presents any metarules associated with the relevant instance of the task that have been completed, as well as the one currently being executed. As a result of the structure of tasks and metarules, the WHY/HOW explanations have an interesting property. A WHY explanation essentially states the premise of the metarule; this is exactly the reason the metarule succeeded. A HOW explanation is a statement of the action of a metarule; this is exactly what was done.

In the next example, we are asking HOW about a low-level task. Note that the answer is at the lowest level of applying rules.

```
** HOW 19.0
[i.e. HOW are we trying to decide whether Mary has
     tuberculosis?]

We did the following:

[20.1] Applied a rule mentioning a causal precursor to
     tuberculosis (tb risk factor; rule525), and
[20.2] Gave thorough consideration to any recent findings.
[20.3] Applied a rule mentioning general evidence for
     tuberculosis (focal neurological signs; rule366), and
[20.4] Gave thorough consideration to any recent findings.
[20.5] Applied a rule mentioning general evidence for
     tuberculosis (compromised host; rule309), and
[20.6] Gave thorough consideration to any recent findings.
```

```
We are now:
[20.7] Applying a rule mentioning general evidence for
     tuberculosis (a history of granulomas on biopsy of
     the liver, lymph node, or other organ; rule002).
We will next:
[20.8] Give thorough consideration to any recent findings.
[back to question 14...]
** NO
```

The HOW question below refers to a high level task (ESTABLISH-HYPOTHESIS-SPACE) mentioned in the omitted sequence of WHYs. Notice that the explanation is at a high level and is entirely abstract.

```
17) Has Mary been hospitalized recently?
** WHY
    .
    .
    .
** HOW 25.0
[i.e. HOW are we trying to develop the differential using
     basic history and pertinent physical exam?]
We did the following:
[26.1] Got a general idea of the problem: categorized it
     into one of several pathogenic classes or disease
     loci, or both.
[26.2] Confirmed and refined the differential diagnosis
     through specific questions.
We are now:
[26.3] Rounding out the diagnostic information by looking
     generally into past medical history and by reviewing
     systems.
[back to question 17...]
** NO
```

Besides these strategic WHYs and HOWs, the user can ask about the current hypothesis, the set of hypotheses currently being considered, and evidence for hypotheses at the domain level.

4.2. COMPARISON TO MYCIN

NEOMYCIN uses an explanation approach similar to MYCIN's, that of explaining its actions in terms of goals and rules, so a brief comparison of the two systems is useful (Fig. 4).

The structure of explanations is parallel, except that in MYCIN rules invoke subgoals through their premises, while NEOMYCIN metarules invoke subtasks through their actions. In fact, NEOMYCIN's rules, which are in the format:

```
If ⟨premise⟩
Then invoke subtasks
```

MYCIN	NEOMYCIN
Basic reasoning: goal → rule → subgoal	Basic reasoning: task → metarule → subtask
A goal is pursued to satisfy the premise of a domain rule (backward chaining)	A task is pursued when executing the action of a metarule (forward reasoning with rule sets)
To explain *why* a goal is pursued, cite the domain rule that uses it as a subgoal (premise)	To explain *why* a task is done, cite the metarule that invokes it (action)
To explain *how* a goal is determined, cite the rules that conclude it	To explain *how* a task is accomplished, cite the metarules that achieve it

FIG. 4. Comparison of MYCIN and NEOMYCIN explanations.

could be rewritten in the MYCIN style of:

If ⟨premise⟩
and subtasks done
Then higher task achieved.

However, we have no specific conclusion to make about the higher task, so the actions of all metarules for a given task would be identical. Moreover, the subtasks are clearly different from the database look-up operations of the premise. It is therefore natural to view the subtasks as actions. What makes NEOMYCIN's explanations qualitatively different from MYCIN's is that they are generated at the level of general strategies, instantiated with domain knowledge, when possible, to make them concrete.

4.3. INTEGRATING METALEVEL AND BASE-LEVEL GOALS

Our attempts to provide strategic explanations have clarified for us some of the basic differences between metarules and domain rules. Originally, we thought that tasks were logically equivalent to domain goals, as metarules were the analog of domain rules. Specifically, when Neomycin asked a question, we thought that the stack of operations would show a sequence like this:

task 1
metarule 1
task 2
metarule 2
.
.
.
task n
metarule n
domain goal 1
domain rule 1
goal 2
backward chained rules {
.
.
.
}
goal m = question asked of the user

Under this goal-rule-goal scheme, WHY questions could proceed smoothly from the domain level to the metalevel. But in fact, metarules sometimes invoke a specific domain rule directly, so the following sequence occurs:

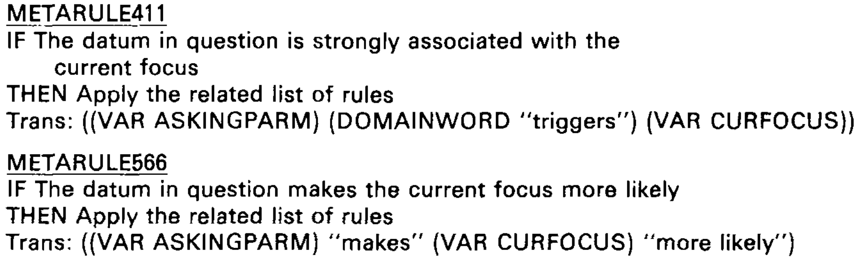
task n
metarule n
domain rule 1
goal 1
.
.
.
goal m = question asked of the user

In this case, there is an implicit task of "apply a domain rule" (invoked by metarule n). Identifying and explaining implicit tasks like this is what we mean by the problem of integrating metalevel and base-level goals. In MYCIN, when Davis (1976) cites the domain rule being applied, he is skipping the immediate intervening metalevel rationale: "We're asking a question to achieve the goal because we were unable to figure out the answer from rules", or "For this goal, we always ask the user before trying rules". In a more recent version of NEOMYCIN, we do make this rational explicit; however, this is uninformative for most users, and the explanation should properly proceed to higher tasks.

4.4. IMPLEMENTATION ISSUES

We mentioned earlier that NEOMYCIN was designed with the intent of guiding a consultation with a general diagnostic strategy. A given task and associated metarules may be applied several times in different contexts in the course of the consultation, for example, testing several hypotheses. To produce concrete explanations, we keep records whenever a task is called or a metarule succeeds; this is sometimes called an *audit trail.* Data such as the focus of the task (e.g. the hypothesis being tested) and the metarule that called it are saved for tasks. Metarules that succeed are linked with any additional variables they manipulate, as well as any information that was obtained as an immediate result of their execution, such as questions that were asked and their answers. When an explanation of any of these is requested, the general translations are instantiated with this historical information.

Figure 5 presents several metarules for the TEST-HYPOTHESIS task translated abstractly. A sample of the audit trail created in the course of a consultation is shown in Fig. 6; this is a snapshot of the TEST-HYPOTHESIS task after question 14 in the



```
METARULE411
IF The datum in question is strongly associated with the
     current focus
THEN Apply the related list of rules
Trans: ((VAR ASKINGPARM) (DOMAINWORD "triggers") (VAR CURFOCUS))

METARULE566
IF The datum in question makes the current focus more likely
THEN Apply the related list of rules
Trans: ((VAR ASKINGPARM) "makes" (VAR CURFOCUS) "more likely")
```

FIG. 5. Sample NEOMYCIN metarules for the TEST-HYPOTHESIS task.

```
TEST-HYPOTHESIS
STATIC PROPERTIES
TRANS: ((VERB decide) whether * has (VAR CURFOCUS))
TASK-TYPE: ITERATIVE
TASKGOAL: EXPLORED
FOCUS: CURFOCUS
LOCALVARS: (RULELST)
CALLED-BY: (METARULE393 METARULE400 METARULE171)
TASK-PARENTS: (GROUP-AND-DIFFERENTIATE PURSUE-HYPOTHESIS)
TASK-CHILDREN: (PROCESS-DATA)
ACHIEVED-BY: (METARULE411 METARULE566 METARULE603)
DO-AFTER: (METARULE332)
AUDIT TRAIL
FOCUS-PARM: (INFECTIOUS-PROCESS MENINGITIS VIRUS
                   CHRONIC-MENINGITIS MYCOBACTERIUM-TB)
CALLER: (METARULE393 METARULE400 METARULE171 METARULE171
                   METARULE171)
HISTORY: [(METARULE411 ((RULELST RULE423)
                         (QUES 4 FEBRILE PATIENT-1 RULE423)))
          (METARULE411 ((RULELST RULE060)
                         (QUES 7 CONVULSIONS PATIENT-1
                           RULE060)))
          .
          .
          (METARULE566 ((RULELST RULE525)
                         (QUES 11 TBRISK PATIENT-1 RULE525))
                   METARULE603
                   ((RULELST RULE366)
                     (QUES 12 FOCALSIGNS PATIENT-1 RULE366))
                   METARULE603
                   ((RULELST RULE309)
                     (QUES 13 COMPROMISED PATIENT-1 RULE309))
                   METARULE603
                     ((RULELST RULE002)
                     (QUES 14 GRANULOMA-HX PATIENT-1 RULE002]
```

FIG. 6. Sample task properties.

consultation excerpt. An example of how the general translations thus relate to the context of the consultation can be seen in the differing explanations for questions 4 and 14, both asked because an hypothesis was being tested.

In order to generate explanations using an appropriate vocabulary for the user, we've identified general words and phrases used in the translations that have parallels in the vocabulary of the domain. At the start of a consultation, the user identifies himself as either a "domain" or "system" expert. Whenever a marked phrase is encountered while explaining the strategy, the corresponding domain phrase is substituted for the medical expert. For example, "triggers" is replaced by "is strongly associated with" for the domain expert.

5. Lessons and future work

The implementation of NEOMYCIN's explanation system has shown us several things. We've found that for a program to articulate general principles, strategies should be represented explicitly and abstractly. They are made explicit by means of a representation in which the control knowledge is explicit, that is, not embedded or implicit in the domain knowledge, such as in rule clause ordering. In NEOMYCIN this is done by using metarules, an approach first suggested by Davis (1976). The strategies are made abstract by making metarules and tasks domain-independent. We've seen that it is possible to direct a consultation using this general problem-solving approach and that resulting explanations are, in fact, able to convey this strategy. As far as the utility of explanations of strategy, trials show that, as one might expect, an understanding of domain level concepts is an important prerequisite to appreciating strategic explanations.

In regard to representation issues, we've found that if control is to be assumed by the tasks and metarules, *all* control must be encoded in this way. Implicit actions in functions or hidden chaining in domain level rules lead to situations which do not fit into the overall task structure and cannot be adequately explained. This discovery recently encuraged us to implement two low-level functions as tasks and metarules, namely MYCIN's functions for acquiring new data and for applying rules. Not only do the resulting explanations reflect more accurately the actual activities of the system, they're also able to convey the purpose behind these actions more clearly.

There is still much that can be done with NEOMYCIN's strategic explanations. We mentioned that our current level of detail includes every task and metarule. We'd like to develop discourse rules for determining a reasonable level of detail for a given user. We also plan to experiment with summarization, identifying the key aspects of a segment of a consultation or the entire session. We might also explain why a metarule failed, why metarules are ordered in a particular way, and the justifications for the metarules. An advantage of our abstract representation of the problem-solving structure is that when the same procedure is applied in different situations, the system is able to recognize this fact. This gives us the capability to produce explanations by analogy, another area for future research.

The design and implementation of the NEOMYCIN explanation system is primarily the work of Diane Warner Hasling, in partial fulfillment of the Master's degree in Artificial Intelligence at Stanford University. We gratefully acknowledge the assistance of Bruce Buchanan, Ted Shortliffe, and Derek Sleeman. Bill Swartout provided us with abstracts of research on explanation presented at the Idylwild Conference in June 1982. This research has been supported in part by ONR and ARI contract N00014-79C-0302 and NR contract 049-479. Computational resources have been provided by the SUMEX-AIM facility (NIH grant RR00785).

References

AIKINS, J. S. (1980). Prototypes and production rules: a knowledge representation for computer consultations. *Ph.D. thesis*, Stanford University (STAN-CS-80-814).

BENBASSAT, J. & SCHIFFMANN, A. (1976). An approach to teaching the introduction to clinical medicine. *Annals of Internal Medicine*, **84**(4), 477–481.

BROWN, J. S. & BURTON, R. R. (1980). Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science*, **2**, 155–192.

BROWN, J. S., COLLINS, A. & HARRIS, G. (1978). Artificial Intelligence and learning strategies. In O'NEIL, H., Ed., *Learning Strategies*. New York: Academic Press.

BROWN, J. S., BURTON, R. R. & DEKLEER, J. (1982). Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II, and III. In SLEEMAN, D. & BROWN, J. S., Eds, *Intelligent Tutoring Systems*, pp. 227–282. London: Academic Press.

CHI, M. T. H., FELTOVICH, P. J. & GLASER, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, **5**, 121–152.

CLANCEY, W. J. (1979). Transfer of rule-based expertise through a tutorial dialogue. *Ph.D. thesis*, Stanford University (August) (STAN-CS-769).

CLANCEY, W. J. (1981). Methodology for building an intelligent tutoring system. *Technical Report*, Stanford University, 1981 (STAN-CS-81-894, HPP-81-18). (Also to appear in KINTSCH, W., POLSON, P. & MILLER, J., Eds, *Methods and Tactics in Cognitive Science*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.)

CLANCEY, W. J. (1983*a*). The epistemology of a rule-based expert system: a framework for explanation. *Artificial Intelligence*, **20**(3), 215–251.

CLANCEY, W. J. (1983*b*). The advantages of abstract control knowledge in expert system design. In *Proceedings of AAAI-83*, pp. 74–78.

CLANCEY, W. J. & LETSINGER, R. (1981). NEOMYCIN: reconfiguring a rule-based expert system for application to teaching. In *Proceedings of the Seventh IJCAI*, pp. 829–836.

DAVIS, R. (1976). Applications of meta-level knowledge to the construction, maintenance and use of large knowledge bases. *Ph.D. thesis*, Stanford University (July) (STAN-CS-76-552, HPP-76-7).

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

DEKLEER, J. & BROWN, J. S. (1982). Assumptions and ambiguities in mechanistic mental models. In GENTNER, D. & STEVENS, A. S., Eds, *Mental Models*. Lawrence Erlbaum Associates, Inc.

ELSTEIN, A. S., SHULMAN, L. S. & SPRAFKA, S. A. (1978). *Medical Problem Solving: An Analysis of Clinical Reasoning*. Cambridge, Massachusetts: Harvard University Press.

GENESERETH, M. R. (1982). The role of plans in intelligent teaching systems. In SLEEMAN, D. & BROWN, J., Eds, *Intelligent Tutoring Systems*, pp. 136–156. London: Academic Press.

LANGLOTZ, C. & SHORTLIFFE, E. H. (1983). Adapting a consultation system to critique user plans. *Technical Report*, Stanford University (April) (HPP-83-2).

MANN, W. C., BATES, M., GROSZ, B., MCDONALD, D., MCKEOWN, K. & SWARTOUT, W. (1981). Text generation: the state of the art and the literature. *Technical Report RR-81-101*, ISI (December).

MCKEOWN, K. R. (1982). Generating natural language text in response to questions about database structure. *Ph.D. thesis*, University of Pennsylvania. Published by University of Pennsylvania as *Technical Report MS-CIS-82-5*.

PATIL, R. S., SZOLOVITS, P. & SCHWARTZ, W. B. (1981). Causal understanding of patient illness in medical diagnosis. In *Proceedings of the Seventh IJCAI*, pp. 893–899 (August). (Also to appear in CLANCEY, W. J. & SHORTLIFFE, E. H., Eds, *Readings in Medical Artificial Intelligence: The First Decade*. New York: Addison–Wesley.)

POPLE, H. (1982). Heuristic methods for imposing structure on ill-structured problems: the structuring of medical diagnosis. In SZOLOVITS, P., Ed., *Artificial Intelligence in Medicine*. Boulder, Colorado: Westview Press.

SCOTT, A. C., CLANCEY, W., DAVIS, R. & SHORTLIFFE, E. H. (1977). Explanation capabilities of knowledge-based production systems. *American Journal of Computational Linguistics*, microfiche 62.

SHORTLIFFE, E. H. (1976). *Computer-based Medical Consultations: MYCIN*. New York: Elsevier.

SWARTOUT, W. R. (1981*a*). Producing explanations and justifications of expert consulting programs. *Ph.D. thesis*, Massachusetts Institute of Technology (January) (MIT/LCS/TR-251).

SWARTOUT, W. R. (1981*b*). Explaining and justifying expert consulting programs. In *Proceedings of the Seventh IJCAI*, pp. 815–822 (August).

WALLIS, J. W. & SHORTLIFFE, E. H. (1982). Explanatory power for medical expert systems: studies in the representation of causal relationships for clinical consultations. *Methods of Information in Medicine*, **21**, 127–136.

WEINER, J. (1980). BLAH, a system which explains its reasoning. *Artificial Intelligence*, **15**, 19–48.

WILLIAMS, M., HOLLAN, J. & STEVENS, A. (1981). An overview of STEAMER: an advanced computer-assisted instruction system for propulsion engineering. *Behavior Research Methods & Instrumentation*, **13**, 85–90.

WINOGRAD, T. (1972). *Understanding Natural Language.* New York: Academic Press.