20

Customized Explanations Using Causal Knowledge

Jerold W. Wallis and Edward H. Shortliffe

Developers of expert systems have increasingly recognized the importance of explanation capabilities to the acceptance of their programs; such capabilities are also critical in medical consultation system development (Gorry, 1973; Shortliffe, 1980). Good explanations serve four functions in a consultation system: (1) they provide a method for examining the program's reasoning if errors arise when the system is being built; (2) they assure users that the reasoning is logical, thereby increasing user acceptance of the system; (3) they may persuade users that unexpected advice is appropriate; and (4) they can educate users in areas where users' knowledge may be weak. These diverse roles impose several requirements on the system. For example, the explanations must adequately represent the reasoning processes of the program, and they should allow the user to examine the reasoning history or underlying knowledge at various levels of detail. In addition, although the program's approach to a problem need not be identical to an expert's approach, the program's overall strategy and reasoning steps must be understandable and seem logical, regardless of the user's level of expertise. This means that the system must have the capability to tailor its explanations to the varying needs and characteristics of its users.

In this chapter we describe recent experiments in the design and implementation of a prototype explanation program. Our past work in explanation for consultation systems, described in Chapter 18, dealt primarily with the ability to cite the rules involved in a particular decision. Although MYCIN's responses provide an accurate description of a portion

This chapter is based on a paper originally appearing in *Methods of Information in Medicine* 21: 127–136 (July 1982). Copyright © by Springer-Verlag,. Inc. All rights reserved. Used with permission.

```
** DO YOU TREAT PSEUDOMONAS-AERUGINOSA BACTEREMIA WITH CARBENICILLIN?
Yes, for treatment of PSEUDOMONAS-AERUGINOSA in BACTEREMIA, the drugs
of choice (ranked by relative efficacy, assuming sensitivity) are:

1st choice:
   CARBENICILLIN-AND-TOBRAMYCIN
   CARBENICILLIN-AND-GENTAMICIN
   AMIKACIN-AND-CARBENICILLIN
2nd choice:
    TOBRAMYCIN
    GENTAMICIN
    AMIKACIN
```

FIGURE 20-1 An example of an interaction with MYCIN's explanation program. Note that the rule for selecting a drug to cover *Pseudomonas aeruginosa* is adequate for allowing MYCIN to reach the correct conclusion but that the underlying reason for combining two drugs is unclear.

of its reasoning, to understand the overall reasoning scheme a user needs to request a display of *all* rules that are used. Additionally, rules such as that shown in the interaction of Figure 20-1 are designed largely for efficiency and therefore frequently omit underlying causal mechanisms that are known to experts but that a novice may need in order to understand a decision. The rule guiding the choice of carbenicillin with an aminoglycoside, for example, does not mention the synergism of the two drugs when combined in the treatment of serious *Pseudomonas aeruginosa* infections. Finally, while MYCIN does have a limited sense of discourse (viz., an ability to modify responses based on the topic under discussion), its explanations are not customized to the questioner's objectives or characteristics.

MYCIN's explanation capabilities were expanded by Clancey in his work on the GUIDON tutorial system (Chapter 26). In order to use MYCIN's knowledge base and patient cases for tutorial purposes, Clancey found it necessary to incorporate knowledge about teaching. This knowledge, expressed as *tutoring rules*, and a four-tiered measure of the baseline knowledge of the student (beginner, advanced, practitioner, or expert), enhanced the ability of a student to learn efficiently from MYCIN's knowledge base. Clancey also noted problems arising from the frequent lack of underlying "support" knowledge, which is needed to explain the relevance and utility of a domain rule (Chapter 29).

More recently, Swartout has developed a system that generates explanations from a record of the development decisions made during the writing of a consultation program to advise on digitalis dosing (Swartout, 1981). The domain expert provides information to a "writer" subprogram, which in turn constructs the advising system. The traces left by the writer, a set of domain principles, and a domain model are utilized to produce explanations. Thus both the knowledge acquisition process and automatic programming techniques are intrinsic to the explanations generated by Swartout's system. Responses to questions are customized for different kinds of users by keeping track of what class is likely to be interested in a given piece of code.

Whereas MYCIN generates explanations that are usually based on a single rule,[1] Weiner has described a system named BLAH (Weiner, 1980) that can summarize an entire reasoning chain in a single explanatory statement. The approach developed for BLAH was based on a series of psycholinguistic studies (Linde, 1978; Linde and Goguen, 1978; Weiner, 1979) that analyzed the ways in which human beings explain decisions, choices, and plans to one another. For example, BLAH structures an explanation so that the differences among alternatives are given before the similarities (a practice that was noted during the analysis of human explanations).

The tasks of interpreting questions and generating explanations are confounded by the problems inherent in natural language understanding and text generation. A consultation program must be able to distinguish general questions from case-specific ones and questions relating to specific reasoning steps from those involving the overall reasoning strategy. As previously mentioned, it is also important to tailor the explanation to the user, giving appropriate supporting causal and empirical relationships. It is to this last task that our recent research has been aimed. We have deferred confronting problems of natural language understanding for the present, concentrating instead on representation and control mechanisms that permit the generation of explanations customized to the knowledge and experience of either physician or student users.

20.1 Design Considerations: The User Model

For a system to produce customized explanations, it must be able to model the user's knowledge and motivation for using the system. At the simplest level, such a model can be represented by a single measure of what the user knows in this domain and how much he or she wants to know (i.e., to what level of detail the user wishes to have things explained). One approach is to record a single rating of a user's *expertise*, similar to the four categories mentioned above for GUIDON. The model could be extended to permit the program to distinguish subareas of a user's expertise in different portions of the knowledge base. For example, the measures could be dynamically updated as the program responds to questions and explains segments

[1] Although MYCIN's WHY command has a limited ability to integrate several rules into a single explanation (Shortliffe et al., 1975), the user wishing a high-level summary must specifically augment the WHY with a number that indicates the level of detail desired. We have found that the feature is therefore seldom used. It would, of course, be preferable if the system "knew" on its own when such a summary is appropriate.

of its knowledge. If the user demonstrates familiarity with one portion of the knowledge base, then he or she probably also knows about related portions (e.g., if physicians are familiar with the detailed biochemistry of one part of the endocrine system, they are likely to know the biochemistry of other parts of the endocrine system as well). This information can be represented in a manner similar to Goldstein's rule pointers, which link analogous rules, rule specializations, and rule refinements (Goldstein, 1978). In addition, the model should ideally incorporate a sense of dialogue to facilitate user interactions. Finally, it must be self-correcting (e.g., if the user unexpectedly requests information on a topic the program had assumed he or she knew, the program should correct its model prior to giving the explanation). In our recent experiments we have concentrated on the ability to give an explanation appropriate to the user's level of knowledge and have deemphasized dialogue and model correction.

20.2 Knowledge Representation

20.2.1 Form of the Conceptual Network

We have found it useful to describe the knowledge representation for our prototype system in terms of a semantic network (Figure 20-2).[2] It is similar to other network representations used in the development of expert systems (Duda et al., 1978b; Weiss et al., 1978) and has also been influenced by Rieger's work on the representation and use of causal relationships (Rieger, 1976). A network provides a particularly rich structure for entering detailed relationships and descriptors in the domain model. *Object nodes* are arranged hierarchically, with links to the possible attributes (*parameters*) associated with each object. The *parameter nodes*, in turn, are linked to the possible *value nodes*, and *rules* are themselves represented as nodes with links that connect them to value nodes. These relationships are summarized in Table 20-1.

The *certainty factor* (CF) associated with each value and rule node (Table 20-1) refers to the belief model developed for the MYCIN system (Chapter 11). The property *ask first/last* controls whether or not the value of a parameter is to be requested from the user before an attempt is made to compute it using inference rules from the knowledge base (see LABDATA, Chapter 5). The *text justification* of a rule is provided when the system builder has decided not to break the reasoning step into further compo-

[2] The descriptive power of a semantic network provides clarity when describing this work. However, other representation techniques used in artificial intelligence research could also have captured the attributes of our prototype system.

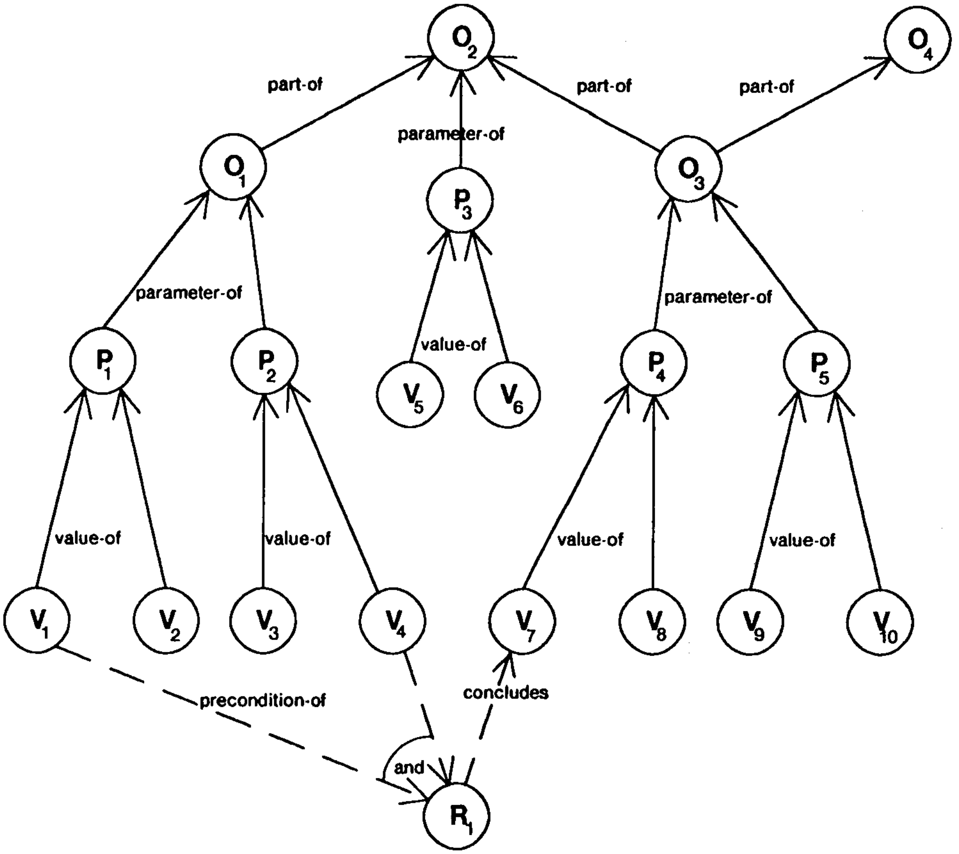

FIGURE 20-2 Sample section of network showing *object, parameter, value,* and *rule nodes.* Dashed lines indicate the following rule:

IF: PARAMETER-1 of OBJECT-1 is VALUE-1, and
PARAMETER-2 of OBJECT-1 is VALUE-4
THEN: Conclude that PARAMETER-4 of OBJECT-3 is VALUE-7

nent parts but wishes to provide a brief summary of the knowledge underlying that rule. *Complexity, importance,* and *rule type* are described in more detail below.

20.2.2 Rules and Their Use

In the network (Figure 20-2) rules connect value nodes with other value nodes. This contrasts with the MYCIN system in which rules are functionally associated with an object-parameter pair and succeed or fail only after

TABLE 20-1

Type of Node	*Static Information (associated with node)*	*Dynamic Information (consultation-specific)*
object node	part-of link (hierarchic) parameter list	
parameter node	object link value-node list default value text definition	
value node	parameter-node link precondition-rule list conclusion-rule list importance complexity ask first/last	contexts for which this value is true certainty factor explanation data ask state
rule node	precondition list (boolean) conclusion certainty factor rule type complexity text justification	explanation data

completion of an exhaustive search for *all* possible values associated with that pair. To make this clear, consider a rule of the following form:

```
IF:  DISEASE-STATE of the LIVER is ALCOHOLIC-CIRRHOSIS
THEN:  It is likely (.7) that the SIZE of ESOPHAGEAL-VEINS is INCREASED
```

When evaluating the premise of this rule to decide whether it applies in a specific case, a MYCIN-like system would attempt to determine the certainty of *all* possible values of the DISEASE-STATE of the LIVER, producing a list of values and their associated certainty factors. Our experimental system, on the other hand, would only investigate rules that could contribute information specifically about ALCOHOLIC-CIRRHOSIS. In either case, however, rules are joined by backward chaining.

Because our system reasons backwards from single values rather than from parameters, it saves time in reasoning in most cases. However, there are occasions when this approach is not sufficient. For example, if a value is concluded with absolute certainty (CF = 1) for a parameter with a mutually exclusive set of values, this necessarily forces the other values to be false (CF = − 1). Lines of reasoning that result in conclusions of absolute certainty (i.e., reasoning chains in which all rules make conclusions with

CF = 1) have been termed *unity paths* (see Chapter 3). In cases of mutually exclusive values of parameters, complete investigation of one value requires consideration of any other value that could be reached by a unity path. Thus the representation must allow quick access to such paths.

When reasoning by elimination, similar problems arise if a system focuses on a single value. One needs the ability to conclude a value by ruling out all other possible values for that parameter; this entails a slight modification of the organizational and reasoning scheme. One strategy is to use this elimination method in cases of mutually exclusive options only after the normal backward-chaining process fails (provided that the possibilities represented in the knowledge base are known to span *all* potential values).

20.2.3 Complexity and Importance

The design considerations for adequate explanations require additions to the representation scheme described above. To provide customized explanations, appropriate for different levels of expertise, we have found it useful to associate a measure of *complexity*, both with the inference rules and with the concepts about which they are concluding. Because some concepts are key ideas in a reasoning chain and should be mentioned regardless of their complexity, a measure of *importance* associated with concepts is useful as well. Currently, both measures are specified at the time knowledge is added to the system, but a dynamic modification of these initial values would improve the flexibility of the approach.

Even though complexity and importance are related, one cannot necessarily be predicted from the other. For example, biochemical details of the endocrine system are *complex* but are not *important* to an understanding of endocrine abnormalities, yet the same *complexity* of biochemical detail is *important* for understanding the glycogen storage diseases. A measure of a fact's importance was also used by Carbonell (1970a) in the form of "relevancy tags," supplemented by "distance" in a semantic network, but he did not distinguish between the two concepts discussed here.

20.3 Explanation Capabilities

20.3.1 Tailored Explanations

The measurements of complexity and importance described above facilitate the generation of tailored explanations. Consider a linear causal chain representing a simplified causal mechanism for the existence of kidney stones (Figure 20-3). A sample interaction based on this reasoning chain might be as follows:[3]

Sample Interaction 1:

```
**EXPERTISE 3 [the user indicates his or her level of expertise on a scale of 1 to
            10 where 10 = expert]

**HOW DOES HYPERPARATHYROIDISM CAUSE RENAL STONES?
      Renal stones are caused by hypercalcemia
          Hypercalcemia is caused by hyperparathyroidism

**MORE DETAIL
       Renal stones are caused by increased urinary calcium
           Increased urinary calcium is caused by hypercalcemia
             Hypercalcemia is caused by increased bone breakdown
                 Increased bone breakdown is caused by hyperparathyroidism

**EXPERTISE 7 [the user indicates that he or she is more expert here than above]

**HOW DOES HYPERPARATHYROIDISM LEAD TO INCREASED BONE BREAKDOWN?
      Bone breakdown is caused by increased osteoclast activity
          Increased osteoclast activity is caused by increased cyclic-AMP
             Increased cyclic-AMP is caused by hyperparathyroidism
```

This sample dialogue demonstrates: (1) the user's ability to specify his or her level of expertise, (2) the program's ability to employ the user's expertise to adjust the amount of detail it offers, and (3) the user's option to request more detailed information about the topic under discussion.

Two user-specific variables are used to guide the generation of explanations:[4]

EXPERTISE: A number representing the user's current level of knowledge. As is discussed below, reasoning chains that involve simpler concepts as intermediates are collapsed to avoid the display of information that might be obvious to the user.

DETAIL: A number representing the level of detail desired by the user when receiving explanations (by default a fixed increment added to the EXPERTISE value). A series of steps that is excessively detailed can be collapsed into a single step to avoid flooding the user with information. However, if the user wants more detailed information, he or she can request it.

As shown in Figure 20-3, a measure of complexity is associated with each value node. Whenever an explanation is produced, the concepts in

[3] Our program functions as shown except that the user input requires a constrained format rather than free text. We have simplified that interaction here for illustrative purposes. The program actually has no English interface.

[4] Another variable we have discussed but not implemented is a focusing parameter that would put a ceiling on the number of steps in the chain to trace when formulating an explanation. A highly focused explanation would result in a discussion of only a small part of the reasoning tree. In such cases, it would be appropriate to increase the detail level as well.

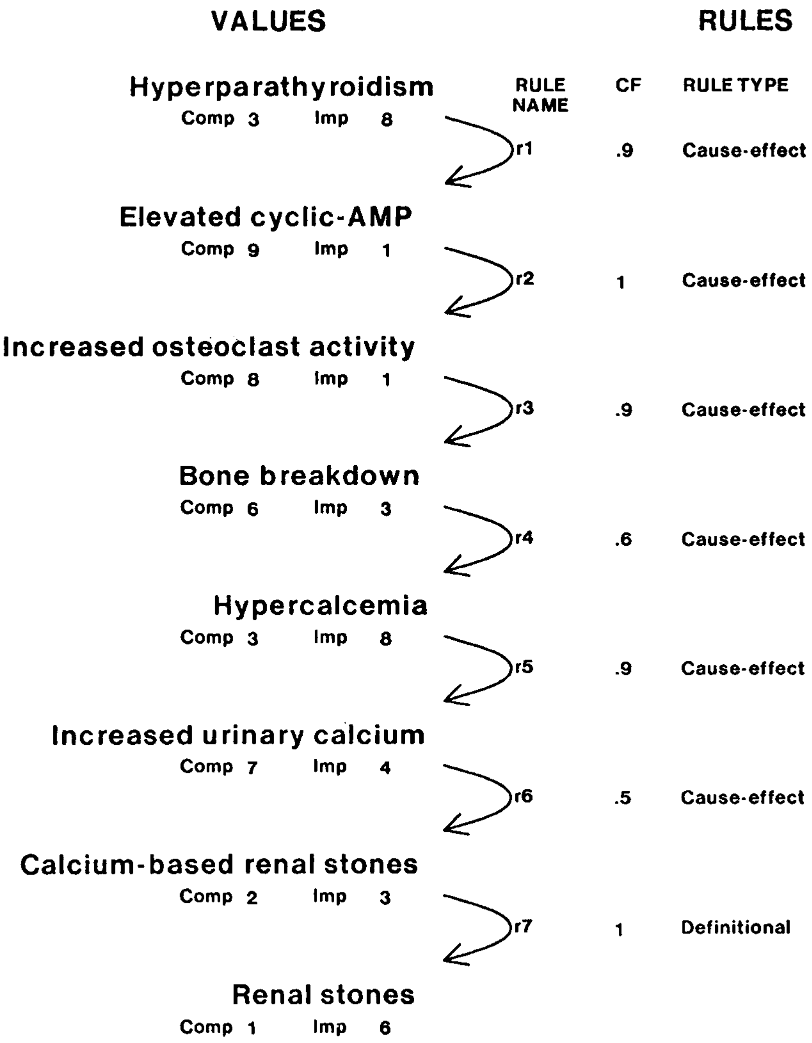

FIGURE 20-3 An example of a small section of a causal knowledge base, with measures of the complexity (Comp) and importance (Imp) given for the value nodes (concepts). This highly simplified causal chain is provided for illustrative purposes only. For example, the effect of parathormone on the kidney (promoting retention of calcium) is not mentioned, but it would have an opposite causal impact on urinary calcium. This reasoning chain is linear (each value has only one cause) and contains only cause-effect and definitional rules. Sample Interactions 1 and 2 (see text) are based on this reasoning chain.

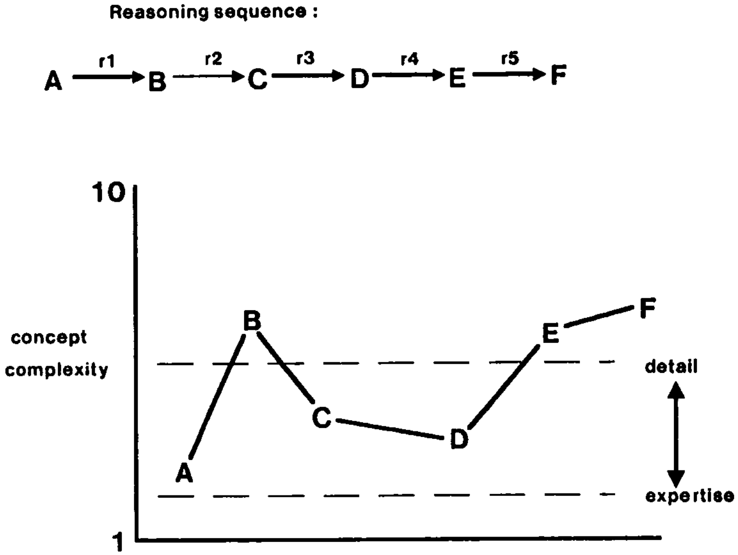

FIGURE 20-4 Diagram showing the determination of which concepts (parameter values) to explain to a user with a given expertise and detail setting. The letters A through F represent the concepts (values of parameters) that are linked by the inference rules rl through r5. Only those concepts whose complexity falls in the range between the dashed lines (including the lines themselves) will be mentioned in an explanation dialogue. Explanatory rules to bridge the intermediate concepts lying outside this range are generated by the system.

the reasoning chain are selected for exposition on the basis of their complexity; those concepts with complexity lying between the user's expertise level and the calculated detail level are used.[5] Consider, for example, the five-rule reasoning chain linking six concepts shown in Figure 20-4. When intermediate concepts lie outside the desired range (concepts B and E in this case), broader inference statements are generated to bridge the nodes that are appropriate for the discussion (e.g., the statement that A leads to C would be generated in Figure 20-4). Terminal concepts in a chain are always mentioned, even if their complexity lies outside the desired range (as is true for concept F in the example). This approach preserves the

[5] The default value for DETAIL in our system is the EXPERTISE value incremented by 2. When the user requests more detail, the detail measure is incremented by 2 once again. Thus, for the three interchanges in Sample Interaction 1, the expertise-detail ranges are 3–5, 3–7, and 7–9 respectively. Sample Interaction 2 demonstrates how this scheme is modified by the importance measure for a concept.

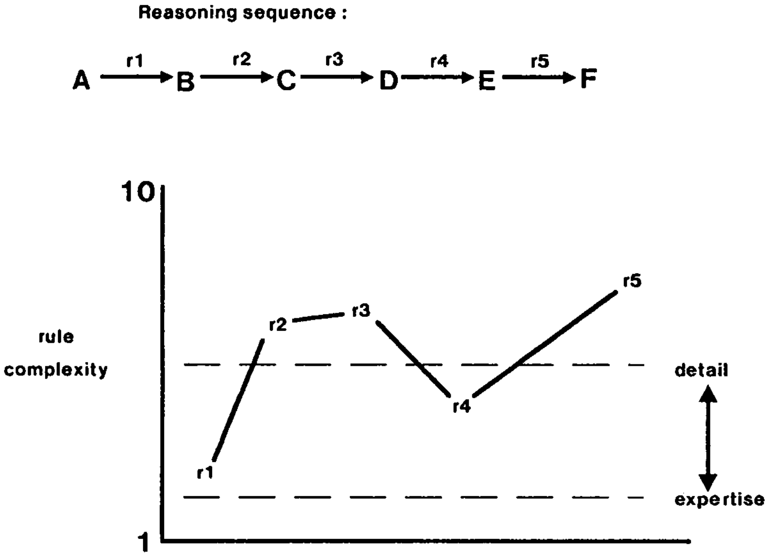

FIGURE 20-5 Diagram showing the determination of which rules to explain further for a user with a given expertise and detail setting. When a rule is mentioned because of the associated concepts, but the rule itself is too complex, further text associated with the rule is displayed.

logical flow of the explanation without introducing concepts of inappropriate complexity.

We have also found it useful to associate a complexity measure with each inference rule to handle circumstances in which simple concepts (low complexity) are linked by a complicated rule (high complexity).[6] This situation typically occurs when a detailed mechanism, one that explains the association between the premise and conclusion of a rule, consists of several intermediate concepts that the system builder has chosen not to encode explicitly.[7] When building a knowledge base, it is always necessary to limit the detail at which mechanisms are outlined, either because the precise mechanisms are unknown or because minute details of mechanisms are not particularly useful for problem solving or explanation. Thus it is useful to add to the knowledge base a brief text justification (Table 20-1) of the mechanism underlying each rule.

Consider, for example, the case in Figure 20-5, which corresponds to

[6] The opposite situation does not occur; rules of low complexity do not link concepts of higher complexity.

[7] Patil has dealt with this problem by explicitly representing causal relationships concerning acid-base disorders at a variety of different levels of detail (Patil et al., 1981).

the reasoning chain represented in Figure 20-4. Although rule r3 links two concepts (C and D) that are within the *complexity-detail range* for the user, the relationship mentioned in rule r3 is itself considered to be outside this range. When generating the explanation for this reasoning chain, the program mentions concepts C and D, and therefore mentions rule r3 despite its complexity measure. Since the rule is considered too complex for the user, however, the additional explanatory text associated with the rule is needed in this case. If the rule had fallen within the complexity-detail range of the user, on the other hand, the text justification for the rule would not have been required.[8]

Further modulation of rule and concept selection is accomplished using the importance measure associated with parameters. A high importance forces the inclusion of a reasoning step in an explanation, thereby overriding the complexity considerations that were shown in Figures 20-4 and 20-5. When the importance level of a concept is two or more points above the expertise of the user, the item is included in the explanation. Consider, for example, the following dialogue, which demonstrates the way in which the importance measure is used:

Sample Interaction 2:

```
**EXPERTISE 6

**HOW DOES HYPERPARATHYROIDISM CAUSE RENAL STONES?
        Renal stones are caused by increased urinary calcium
            Increased urinary calcium is caused by hypercalcemia
                  Hypercalcemia is caused by increased bone breakdown
                      Bone breakdown is caused by increased osteoclast activity
                          Increased osteoclast activity is caused by hyperparathyroidism
```

Note that this example shows a response to the same question asked in Sample Interaction 1. This time, however, the expertise level is 6 rather than 3. Hypercalcemia is therefore mentioned only because its importance level of 8 (see Figure 20-3) is 2 points higher than the expertise of the user; the complexity level of hypercalcemia does not fall within the expertise-detail range of the user and thus would not have been included in the explanation if it were not for its high importance. The other items mentioned are either terminal concepts in the chain (renal stones and hyperparathyroidism) or have a complexity measure lying within the user's expertise-detail range of 6–8.

Many reasoning chains are not as simple as those shown in Figures 20-3, 20-4, and 20-5. When explaining a branched reasoning chain, for example, the explanation system can set aside the branches of the chain and mention them only when it is appropriate to the level of detail required by the user. This feature provides users with an overview of the reasoning process to help them decide whether it is necessary to examine the more detailed steps. This capability is illustrated in the following dialogue, which

[8] An example of this approach is included in Sample Interaction 4 in Section 20.3.2.

involves a patient with hypercalcemia and a possible malignancy who has undergone prolonged bed rest:

Sample Interaction 3:

```
**WHY DOES THE PATIENT HAVE INCREASED SERUM CALCIUM?
      Increased serum calcium is suggested by immobilization and malignancy
**MORE DETAIL
          Increased serum calcium is implied by increased bone breakdown
             Increased bone breakdown is suggested by 2 paths of reasoning:
                Increased bone breakdown is implied by increased osteoclast activity
                   Increased osteoclast activity is implied by prolonged immobilization
                Increased bone breakdown is also implied by malignant bone invasion
```

20.3.2 Types of Rules

Our refinement of the rule types presented by Clancey (Chapter 29) yields five types of rules[9] that are relevant to explanation strategies:

definitional: the conclusion is a restatement of the precondition in different terms

cause-effect: the conclusion follows from the precondition by some mechanism, the details of which may not be known

associational: the conclusion and the precondition are related, but the causal direction (if any) is not known

effect-cause: the presence of certain effects are used to conclude about a cause with some degree of certainty

self-referencing: the current state of knowledge about a value is used to update that value further[10]

The importance of distinguishing between cause-effect and effect-cause rules is shown in Figure 20-6, which considers a simplified network concerning possible fetal Rh incompatibility in a pregnant patient. Reasoning backwards from the goal question "Is there a fetal-problem?" one traverses three steps that lead to the question of whether the parents are Rh incompatible; these three steps use cause-effect and definitional links only. However, in order to use the laboratory data concerning the amniotic fluid to form a conclusion about the presence of fetal hemolysis, effect-cause links must be used.

The sample interactions in Section 20.3.1 employed only cause-effect

[9] Rules considered here deal with domain knowledge, to be distinguished from strategic or meta-level rules (Davis and Buchanan, 1977).

[10] In many cases self-referencing rules can be replaced by strategy rules (e.g., "If you have tried to conclude a value for this parameter and have failed to do so, then use the default value for the parameter").

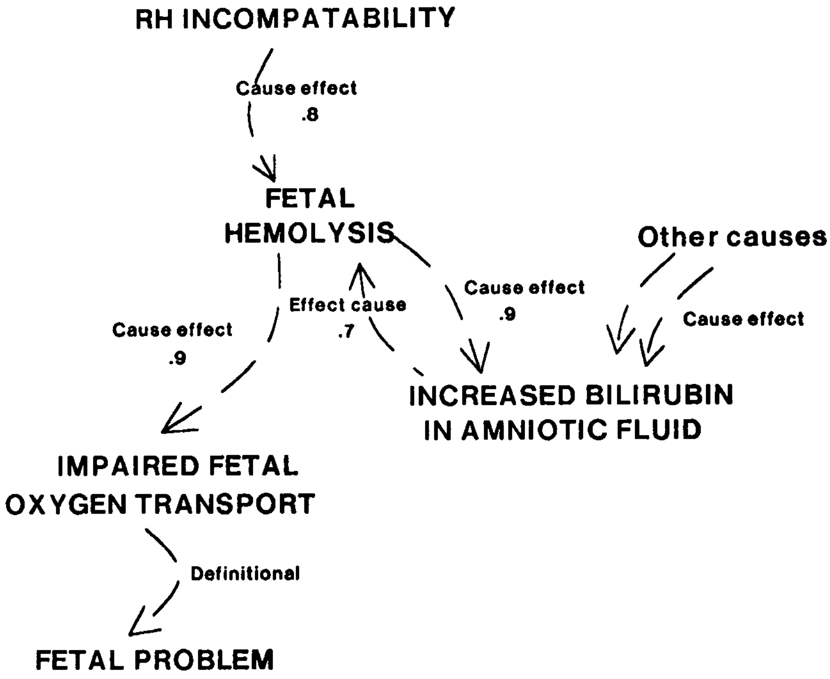

FIGURE 20-6 A simple causal network showing the difference in reasoning between effect-cause and cause-effect rules in the medical setting. The number beside a link indicates the *certainty factor* (CF) associated with the rule. Note that an actual rule network for this domain would be more complex, with representation of intermediate steps, associated medical concepts, default values, and definitions.

and definitional rules. An explanation for an effect-cause rule, on the other hand, requires a discussion of the inverse cause-effect rule (or chain of rules) and a brief mention of other possibilities to explain the certainty measure associated with the rule. As discussed above, the expertise of a user may also require that the program display a text justification for the causal relationships cited in a cause-effect rule. Consider, for example, an interaction in which an explanation of the effect-cause rule in Figure 20-6 is produced:

Sample Interaction 4:

```
**WHY DO INCREASED BILIRUBIN COMPOUNDS IN THE AMNIOTIC FLUID IMPLY FETAL
  HEMOLYSIS?
     Fetal hemolysis leads to bilirubin compounds in the fetal circulation;
```

equilibration then takes place between the fetal plasma and the amniotic
fluid, leading to increased bilirubin compounds in the amniotic fluid
While the relationship in this direction is nearly certain, the inverse
relationship is less certain because of the following other possible
causes of increased bilirubin compounds in the amniotic fluid:
Maternal blood in the amniotic fluid from trauma
Maternal blood in the amniotic fluid from prior amniocentesis

The response regarding the equilibration of fetal plasma and amniotic fluid is the stored text justification of the cause-effect rule that leads from "fetal hemolysis" to "increased bilirubin in amniotic fluid." The individual steps could themselves have been represented in causal rules if the system builder had preferred to enter rule-based knowledge about the nature of hemolysis and bilirubin release into the circulation. The second component of the response, on the other hand, is generated from the other cause-effect rules that can lead to "increased bilirubin in amniotic fluid."

The other types of rules require minor modifications of the explanation strategy. Definitional rules are usually omitted for the expert user on the basis of their low complexity and importance values. An explanation of an associational rule indicates the lack of known causal information and describes the degree of association. Self-referencing rules frequently have underlying reasons that are not adequately represented by a causal network; separate support knowledge associated with the rule (Chapter 29), similar to the text justification shown in Sample Interaction 4, may need to be displayed for the user when explaining them.

20.4 Causal Links and Statistical Reasoning

We have focused this discussion on the utility of representing causal knowledge in an expert system. In addition to facilitating the generation of tailored explanations, the use of causal relationships strengthens the reasoning power of a consultation program and can facilitate the acquisition of new knowledge from experts. However, an attempt to reason from causal information faces many of the same problems that have been encountered by those who have used statistical approaches for modeling diagnostic reasoning. It is possible to generate an effect-cause rule, and to suggest its corresponding probability or certainty, only if the information given in the corresponding cause-effect rule is accompanied by additional statistical information. For example, Bayes' Theorem may be used to determine the probability of the *i*th of *k* possible "causes" (e.g., diseases), given a specific observation ("effect"):

$$P(\text{cause}_i|\text{effect}) = \frac{P(\text{effect}|\text{cause}_i)\ P(\text{cause}_i)}{\sum_{j=1}^{k} P(\text{cause}_j)\ P(\text{effect}|\text{cause}_j)}$$

This computation of the probability that the *i*th possible cause is present given that the specific effect is observed, $P(\text{cause}_i|\text{effect})$, requires knowledge of the *a priori* frequencies $P(\text{cause}_i)$ for each of the possible causes (cause_1, cause_2 . . . cause_k) of the effect. These data are not usually available for medical problems and are dependent on locale and prescreening of the patient population (Shortliffe et al., 1979; Szolovits and Pauker, 1978). The formula also requires the value of $P(\text{effect}|\text{cause}_j)$ for all cause-effect rules leading to the effect, not just the one for the rule leading from cause_i to the effect. In Figure 20-6, for example, the effect-cause rule leading from "increased bilirubin in amniotic fluid" to "fetal hemolysis" could be derived from the cause-effect rule leading in the opposite direction only if all additional cause-effect rules leading to "increased bilirubin in amniotic fluid" were known (the "other causes" indicated in the figure) and if the relative frequencies of the various possible causes of "increased bilirubin in amniotic fluid" were also available. A more realistic approach is to obtain the inference weighting for the effect-cause rule directly from the expert who is building the knowledge base. Although such subjective estimates are fraught with danger in a purely Bayesian model (Leaper et al., 1972), they appear to be adequate (see Chapter 31) when the numerical weights are supported by a rich semantic structure (Shortliffe et al., 1979).

Similarly, problems are encountered in attempting to produce the inverse of rules that have Boolean preconditions. For example, consider the following rule:

```
IF:  (A and (B or C))
THEN:  Conclude D
```

Here D is known to imply A (with a certainty dependent on the other possible causes of D and their relative frequencies) only if B or C is present. While the inverse rule could be generated using Bayes' Theorem given the *a priori* probabilities, one would not know the certainty to ascribe to cases where *both* B *and* C are present. This problem of conditional independence tends to force assumptions or simplifications when applying Bayes' Theorem. Dependency information can be obtained from data banks or from an expert, but cannot be derived directly from the causal network.

It is instructive to note how the Present Illness Program (PIP) and CADUCEUS, two recent medical reasoning programs, deal with the task of representing both cause-effect and effect-cause information. CADUCEUS (Pople, 1982) has two numbers for each manifestation of disease, an "evoking strength" (the likelihood that an observed manifestation is caused by the disease) and a "frequency" (the likelihood that a patient with a disease will display a given manifestation). These are analogous to the inference weightings on effect-cause rules and cause-effect rules, respectively. However, the first version of the CADUCEUS program (INTERNIST-1) did not allow for combinations of manifestations that give higher (or lower) weighting than the sum of the separate manifestations,[11] nor did it provide a way to explain the inference paths involved (Miller et al., 1982).

PIP (Pauker et al., 1976; Szolovits and Pauker, 1978) handles the implication of diseases by manifestations by using "triggers" for particular disease frames. No weighting is assigned at the time of frame invocation; instead PIP uses a scoring criterion that does not distinguish between cause-effect and effect-cause relationships in assigning a numerical value for a disease frame. While the information needed to explain the program's reasoning is present, the underlying causal information is not.[12]

In our experimental system, the inclusion of both cause-effect rules and effect-cause rules with explicit certainties, along with the ability to group manifestations into rules, allows flexibility in constructing the network. Although causal information taken alone is insufficient for the construction of a comprehensive knowledge base, the causal knowledge can be used to propose effect-cause relationships for modification by the system-builder. It can similarly be used to help generate explanations for such relationships when effect-cause rules are entered.

20.5 Conclusion

We have argued that a need exists for better explanations in medical consultation systems and that this need can be partially met by incorporating a user model and an augmented causal representation of the domain knowledge. The causal network can function as an integral part of the reasoning system and may be used to guide the generation of tailored explanations and the acquisition of new domain knowledge. Causal information is useful but not sufficient for problem solving in most medical domains. However, when it is linked with information regarding the complexity and importance of the concepts and causal links, a powerful tool for explanation emerges.

Our prototype system has been a useful vehicle for studying the techniques we have discussed. Topics for future research include: (1) the development of methods for dynamically determining complexity and importance (based on the semantics of the network rather than on numbers provided by the system builder); (2) the discovery of improved techniques for using the context of a dialogue to guide the formation of an expla-

[11] This problem is one of the reasons for the move from INTERNIST-1 to the new approaches used in CADUCEUS (Pople, 1982).

[12] Recently the ABEL program, a descendent of PIP, has focused on detailed modeling of causal relationships (Patil et al., 1981).

nation; (3) the use of linguistic or psychological methods for determining the reason a user has asked a question so that a customized response can be generated; and (4) the development of techniques for managing the various levels of complexity and detail inherent in the mechanistic relationships underlying physiological processes. The recent work of Patil, Szolovits, and Schwartz (1981), who have separated such relationships into multiple levels of detail, has provided a promising approach to the solution of the last of these problems.