Report No. STAN-CS-87-1178
*Also numbered KSL-86-27*

*Clancey, W. J. (1989). Viewing knowledge bases as qualitative models. IEEE/Expert, 4 (2), 9–23.*

# Viewing Knowledge Bases as Qualitative Models



by

William J. Clancey

Department of Computer Science

Stanford University
Stanford, CA 94305



# Viewing Knowledge Bases as Qualitative Models

by

William J. Clancey

Stanford Knowledge Systems Laboratory
Department of Computer Science
701 Welch Road, Building C
Palo Alto, CA 94304

The studies reported here were supported (in part) by:

The Office of Naval Research
Personnel and Training Research Programs
Psychological Sciences Division
Contract No. N00014-85K-0305

The Josiah Macy, Jr. Foundation
Grant No. B852005
New York City



# 1. Introduction

With the growing interest in expert systems in academia and industry, one question recurs that never seems to receive a satisfying answer: How are expert systems different from conventional programs? Different perspectives suggest different answers. In terms of programming language, expert systems are programs written in "logic," "rules," or "frames" (Friedland, 1985). From the perspective of new computational capabilities, expert systems are programs that can explain their reasoning (Davis, 1986). Or considering how these programs are used, expert systems are consultants, monitors, designers, etc. (Hayes-Roth, et al., 1983). But these perspectives are not completely satisfying because they fail to explain how conventional programs are inherently different: Aren't Fortran conditional statements "rules"? Why is it difficult to write a Fortran program that can explain its reasoning? Why isn't a Bayesian diagnostic program an expert system?

An alternative point of view, which this paper develops, considers the nature of the knowledge encoded in these programs. Expert systems deal with heuristics, uncertain knowledge (Feigenbaum, 1977). Their reasoning is qualitative, not in terms of precise measures. But in the AI literature, the term "qualitative reasoning," characterizing an important and growing subarea of research, has been almost exclusively applied to programs with some kind of *simulation model* of a physical system, such as an electronic circuit (Bobrow, 1984, Chandrasekaran and Milne, 1985). This common definition of qualitative reasoning excludes the majority of expert systems, and, as this paper argues, it is largely responsible for the gap in our understanding.

What is an expert system? A simple answer can be given, one that is half-known and has been half-stated by many people: Expert systems contain qualitative models of the world, in contrast with quantitative models, which involve mathematical laws, such as in physics, electronics, and economics (Padulo and Arbib, 1974). The idea is half-stated in the common view that AI is concerned with *symbolic programming* (Harmon and King, 1985), but this is not a satisfying distinction because numbers are symbols, too. An alternative description, "non-numeric programming," avoids this ambiguity and is also widely used. However, the emphasis on "programming" has distracted us; the idea that these programs contain *non-numeric models* has been generally ignored. Describing AI as "non-numeric science and engineering" is more to the point: *Artificial Intelligence is the study of computational techniques for acquiring, representing, and using qualitative models of physical, perceptual, cognitive, and social systems.*

The purpose of this paper is to give this definition force by enriching our understanding of what qualitative models are. Briefly, a qualitative model describes some system in the world in terms of causal, compositional, or subtype relations among objects and events. One of the most important ideas of this paper is that there is a well-established repertoire of methods for representing qualitative models—networks based on the idea of prototype subsumption, state-

transition, and structural and procedural composition. By studying and using these network representations, AI has established a foundation for a science and engineering of qualitative models.

From early on, we have followed the approach of decomposing knowledge from how it is used, abstracting knowledge structures and reasoning procedures, and formulating an increasingly more general understanding of what knowledge engineering and knowledge bases are all about (Swartout, 1981, Clancey, 1983a, Clancey, 1983b, McDermott, 1983, Szolovits, 1985, Clancey, 1985, Smith, 1985). It is now apparent that knowledge bases contain *models of systems* in the world. Reasoning involves *sequences of tasks*, such as "monitoring" and "diagnosis," by which an understanding or model of specific situations is related to action plans (Section 3). Programs use a simple *repertoire of qualitative modeling techniques*, commonly called "knowledge representations" (Section 4). The idea of a *situation-specific model* makes concrete what programs know and how problem solving can be described in terms of model-manipulation operators (Section 5). Finally, an historical perspective shows how AI's concern with adaptiveness and rationality of the autonomous agent emphasizes the *role* of a model as what a problem solver knows (hence, "knowledge base") (Section 6). This has been to the detriment of understanding the primary characteristic of knowledge in terms of *models* that partition the world, viewing it selectively and making it coherent for some purpose.

## 2. The need for a synthesis

Given the generality of the term "model" and the all-encompassing nature of this paper, it is worthwhile addressing a few possible objections up front.

First, we tend to adopt an idealized view of what a model is. Observing how impoverished our programs are, we tend to say that they are not models or, at least, "not real models." This critical point of view is valuable for carrying our research forward, but it creates a distinction that only confuses what we are doing and have accomplished. This paper argues that knowledge base structures are models because they describe what is happening in the world and provide a basis for action, that is they *function* as models.

Second, when a number of pieces are brought together, it may look like the picture was always obvious. For example, recalling the idea of a "frame" (Minsky, 1975)—a prototypic description providing a basis for explaining what happens in the world, making predictions, and taking action—it may seem obvious that frames are a kind of qualitative model. *But why don't we teach it this way?* Our understanding of our field is fragmented and inconsistent. Consider:

- Classification and causal network descriptions are commonly used for modeling physical processes (Lehnert, 1978, Weiss, et al., 1978, Szolovits, 1985), but this work is not integrated in the collection, "Qualitative Reasoning about Physical Systems"

(Bobrow, 1984).

- When we refer to a knowledge base as an "expert model" or "consultation model" (Weiss, 1979), we tend to forget that it contains models of systems in the world. By separating out the system descriptions in a knowledge base, we can begin to identify particular kinds of network structures as *different ways of modeling processes* (Section 4).

- In most medical expert systems, a diagnosis is taken to be the *name* of a disease rather than a coherent description of what is happening in the world—a model—that causally relates states and processes. Perhaps because the programs do not structurally simulate pathophysiological processes, researchers have not thought about inference in terms of model construction (Section 5).

- Instructional research has emphasized that people generally have *some* model of how a system works or what procedure to follow for solving a problem, sometimes buggy and incomplete (Gentner and Stevens, 1983). From this, we might expect that expert system explanations would relate the program's model to the user's, perhaps focusing on violated expectations. Instead, we describe explanation in terms of articulating what *the program* knows and what it did (Davis, 1976, Swartout, 1981, Hasling, et al., 1984). By definition, the program has something the user does not have, "expertise." Thus, we think in terms of "transferring the expertise" (Davis, 1976, Clancey, 1979), rather than *relating alternative models*.

Similarly, the familiar line that AI is "constructing computer programs which exhibit behavior we call 'intelligent behavior' when we observe it in human beings" (Feigenbaum and Feldman, 1963) has provided a useful focus, but leaves out the modeling methodology upon which our work is based.

## 3. The inference structure of expert systems

Expert systems contain descriptions of system behavior and design, which are *models*, in the usual sense of selective abstractions that allow explanation and/or prediction of events in the world. The knowledge encoded in expert systems *functions* as a model, regardless of what representational form it takes. In this section we describe *what expert systems do* in terms of generic tasks for analyzing and synthesizing systems. The next section lays out a spectrum of representational methods.

All expert systems make assertions about what is true in the world or what actions might be taken: The patient has a fever, the infection might be meningitis, penicillin is a possible therapy, etc. If we lay out how assertions are related as chains of inference, we observe

regularities. A kind of secondary structure emerges. One such structure, termed "heuristic classification," involves systematic abstraction, association between class hierarchies, and refinement (Clancey, 1985). The inference process, that is, the order in which inferences are made, is of course important, but it is a different issue.

Heuristic classification is a problem-solving method by which solutions are selected from a set of pre-enumerated alternatives. The question naturally arises, for what kinds of problems is heuristic classification useful? One approach is to consider what kinds of things might be selected from a list of pre-enumerated alternatives. Examples include: user models, items in a catalog, diagnoses, skeletal plans, numeric models (Clancey, 1985).

Continuing our attempt to generalize, are there patterns in what data and solutions can be? What kinds of things are classified and heuristically related? We start by laying out common sequences:

| Inference Structure | Example Program[1] |
|---|---|
| patient -> disease -> therapy | MYCIN |
| reader -> book | GRUNDY |
| customer -> wine | WINE ADVISOR |
| structure -> numeric model -> analysis program | SACON |
| circuit behavior -> fault | SOPHIE |
| causal reasoning bug -> misconception -> instruction | WHY |
| program bug -> misconception | MENO |

Several generalizations can be made:

1. The "things" being related are *models*, that is, general, abstracted descriptions of specific things in the world. Different terminology—"disease prototype," "user stereotype," "numeric model," "structural abstraction"—has tended to obscure this basic pattern.

2. Models are related in a limited number of ways, by what are called *generic tasks*. We introduce the idea of a *system* and redescribe the things being related in terms of an operation performed to some system at each step (Clancey, 1985):

   ```
   specify -> design -> assemble
   monitor -> diagnose -> modify
   identify -> predict
   monitor -> control
   ```

   For example, "patient -> disease" is replaced by "monitor (detect abnormal body system states and environmental influences) -> diagnose (describe and determine cause of the malfunctioning subsystem's structural and state abnormalities)."

3. We can extend the argument beyond heuristic classification to include all expert

[1] References: MYCIN: (Shortliffe, 1976); GRUNDY: (Rich, 1979); WINE ADVISOR: Teknowledge, Inc.; SACON: (Bennett, 1979); SOPHIE: (Brown, et al., 1982); WHY: (Stevens, et al., 1982); MENO: (Soloway, et al., 1981)

systems. The basis for this is simple: Patterns like "monitor -> diagnose -> repair" are very familiar; we know that they go beyond how diagnoses and repairs are reasoned about. For example, we can return to the pattern of the wine advisor ("specify -> design") and include R1 because it relates a customer's requirements to a VAX system configuration.

We aim for completeness, but of course establishing that these generalizations are true for *every* expert system is a daunting task. We can only consider them to be hypotheses, albeit based on considerable experience, and see if they hold up to further analysis.

In introducing the idea of a "system" as the focus of reasoning, we view tasks not in isolation (as described in (Hayes-Roth, et al., 1983)), but as the range of things that we can do to any given system. On this basis, we can begin to argue about the completeness of the list of tasks. For example, we can reformulate Davis's idea of fault models in these terms. Fault models make explicit "what might have been done to a system" to cause faulty behavior (Davis, 1984):

- Monitor the system and determine how it is different from the intended design: Structure intended to change, such as a gate, may be fixed; components may be functioning wrong or have the wrong structure (e.g., a short).
- The system may have been assembled wrong (e.g., chip flaw).
- The design may be wrong.
- The specification may be wrong.

Davis emphasizes the heuristic value of ordering these considerations. Our system-model orientation suggests that "fault models" can be generalized to all systems and generated from the set of generic tasks. Specifically, we could add "wrong prediction" (observer's expectations were incorrect) and "wrong control" (wrong input for desired output) to Davis's list.

We are now in a position to generalize once again, condensing the four sequences of generic tasks into one sequence (Figure 3-1). In simple terms, we relate desired or observed *system behavior* to a *system description* to some *plan* for taking action in the world or *expectation* about what will happen in the world. Common names for action plans are: "assembly plan," "instructional plan," "therapy plan," and "control plan." The intermediate design or identification might describe a *subsystem*, as in the relation between a patient and his infection. Also, in moving to the third stage, we often are describing a *containing system* that will carry out the assembly, modification, or control process, as in the relation between a client and his vacation plan or a design for a molecular structure and an experiment plan.

Again, this is a description of inference structure, not the order in which inferences are made. For example, specification and design are commonly iterative. In cognitive modeling for instruction, we work backwards from an artifact (say a student's computer program), to an implicit design (the model or plan for the program), to models of the environment in which

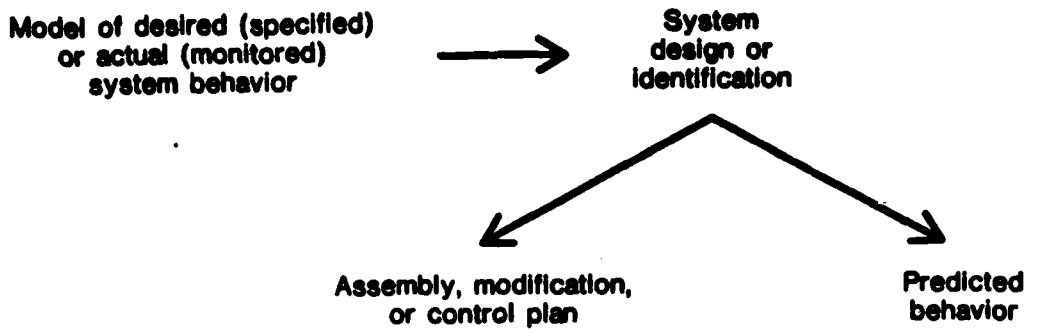


**Figure 3-1:** General inference structure for reasoning about systems

the artifact is to function (beliefs about what the program is supposed to do) and the components out of which it is made (beliefs about the programming language) (Johnson and Soloway, 1984). Note that for the special case of heuristic classification, system models and action plans are selected from pre-enumerated descriptions.

## 4. Representation of qualitative models

By relating qualitative methods for modeling systems, we can see our enterprise as a whole. One possible taxonomy distinguishes between the explanatory power of classification and simulation models (Figure 4-1).

Classification models can be used to *recognize* processes, to "account for behavior" of a system, by naming or categorizing the observed pattern (Minsky, 1963). As a phenomenological description, this corresponds, in general terms, to the idea of *frames* (Minsky, 1975), with the temporal nature of processes made explicit in the idea of *scripts* (Schank, 1975, Stevens, et al., 1982) and described more carefully as an "encapsulated history" (Forbus, 1984). A process description identifies a sequence of events that happen over time, spread over several locations (Ackoff, 1974). For example, an infectious process involves intrusion of an organism into the body, movement to a favorable growth site, response by the body, and inflammation. Typically such descriptions are aggregated to reflect general properties and omit sequential details, thus treating the system as an "object" (e.g., descriptions of people as stereotypes (Rich, 1979)). Such descriptions are *models* because they provide an explanatory accounting of what happens in the world and a basis for action (Achinstein, 1983).

Simulation models, broadly construed, provide some description of how the system produces the observed behavior. The model is "runnable," allowing predictions to be made of how a system will change given a set of initial conditions. A *behavioral simulation* describes how a system *appears*, in terms of "hidden states," "observed manifestations," and causal relations among them. A *functional simulation* model places behavior in a larger context, indicating the *role* it plays in achieving the properties of a larger system. For example, a functional model of a radio would refer to amplification and locking on to a broadcast station, while a behavioral model would only describe current flow and changing voltages. Functional models

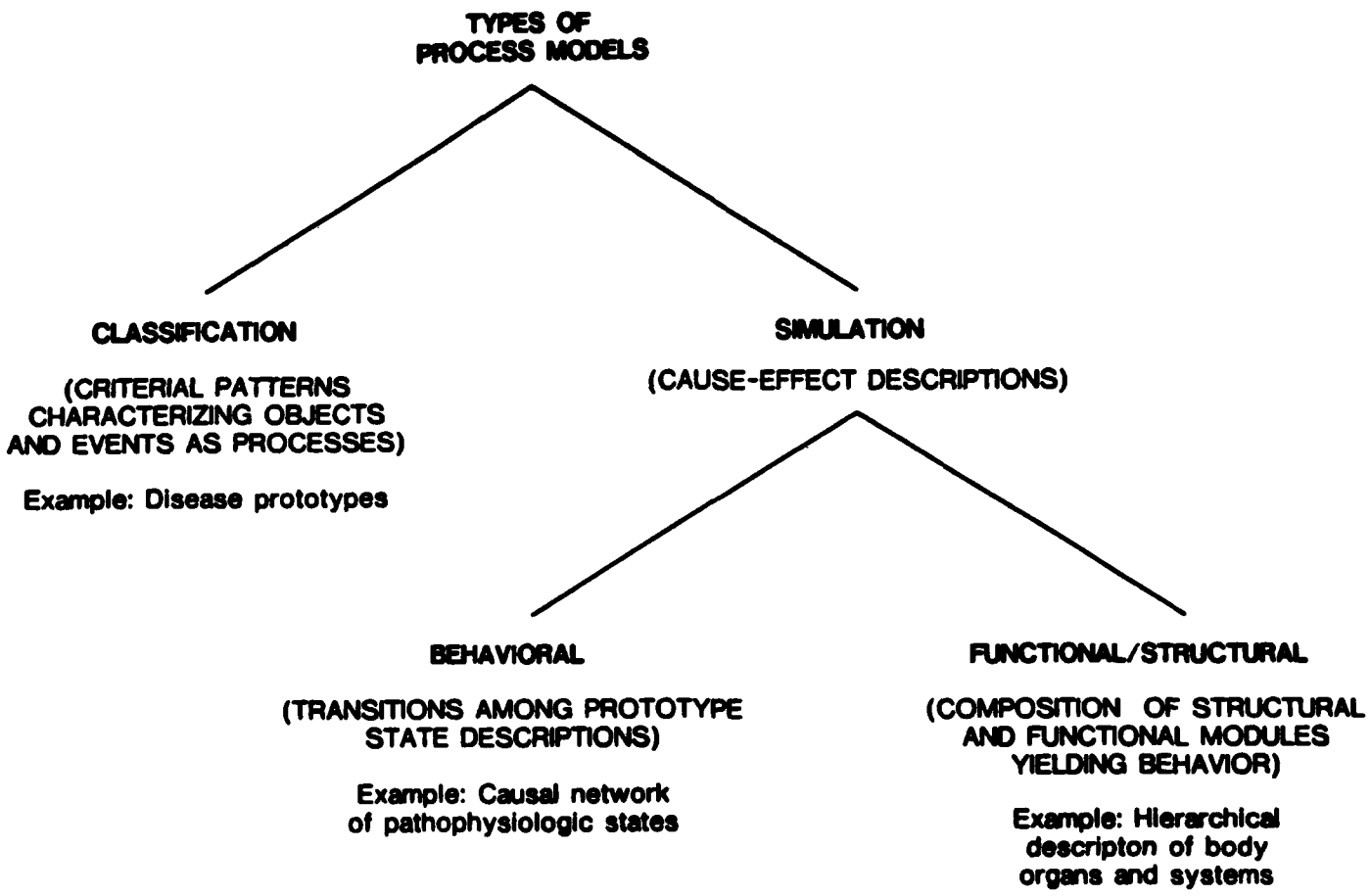


**Figure 4-1:** Types of qualitative models of processes

*account for patterns in behavior* and indicate what larger goals these patterns satisfy (Kosslyn, 1980). As an abstraction, a function is something the system can accomplish in multiple ways, depending on its state and the demands of its environment (Ackoff, 1974).

Regarding completeness, classification and behavioral models do not necessarily characterize the full state of the system being reasoned about on any level of analysis, and cannot necessarily predict what state will follow from arbitrary initial conditions (for example, allowing that parts of the system are functioning normally). "Hidden" internal states may be described, but the purpose of transitions and how transitions follow from the structure of the system (the physical components) are not completely described (Weiss, et al., 1978, Szolovits, 1985).

In contrast, a functional model makes a claim about *completeness* of the explanation or system description. The purpose of the system is captured procedurally on multiple levels of abstraction, so states can be related to functional goals. For example, a functional model of diagnosis describes reasoning in terms of general goals for supporting and refining alternative explanations (Patil, 1981, Clancey, 1984), rather than behaviorally, in terms of domain-specific inferences.

Finally, a structure/function model, to which the term "qualitative model" is usually applied in AI research, gives a full accounting for each component in the system in terms of its role in fulfilling the function of the system. Thus, a functional/structural simulation constitutes a strong theory of the design of a system and the mechanism that lies behind observed behavior (De Kleer and Brown, 1984). A model may be functional, without a structural component, as in models of human problem solving, which typically mention brain components in only very general terms (Kosslyn, 1980).

Figure 4-2 summarizes the different perspectives by which qualitative models describe systems: as *processes* (a prototype hierarchy of I/O or cause/effect patterns); *states* (a graph of states linked by cause and subtype); *functions* (procedural modules hierarchically composed by I/O relations); and *structures* (physical components composing functional modules). The next step in the analysis, beyond the scope of this paper, is to relate these alternative models for recognizing a system and understanding its behavior to the demands of generic tasks (Section 3). Notice in particular that the qualitative modeling repertoire is not specific to physical systems; for example, similar techniques are used for modeling the cognitive processes of discourse, diagnosis, and planning.

| CONCEPT | RELATION | COMMON NAME | PROGRAM EXAMPLES |
|---|---|---|---|
| I. PROCESS | SUBSUMPTION | PROTOTYPE CLASSIFICATION | (Pauker, 1977) (Lehnert, 1978) |
| II. STATE | CONDITIONAL TRANSITION (CAUSE) | CAUSAL NETWORK | (Brown, et al.,1973) (Rieger & Grinberg, 1977) (Weiss, et al., 1978) |
| III. FUNCTION | COMPOSITION | PROCEDURAL NETWORK | (Sacerdoti, 1974) (Brown, et al., 1982) (Clancey, 1984) |
| IV. STRUCTURE | COMPOSITION | STRUCTURE-FUNCTION MODEL | (de Kleer, 1979) (Genesereth, 1982) |

**Figure 4-2:** Conceptual structure of network representations for systems

This framework can be used as a starting point for understanding the value of multiple representations. For example, it is common for a causal-associational state network to be mapped to a disease process hierarchy (Weiss and Kulikowski, 1984, Szolovits, 1985, Clancey and Letsinger, 1984), thus relating a descriptive view of current system behavior to a developmental accounting. Process descriptions can be combined with a structural simulation for coping with complex or unusual situations (Bylander and Chandrasekaran, 1985, Koton, 1985, Fink, 1985). Qualitative models can be useful for controlling and interpreting quantitative simulations (Brown, 1975, Apte and Weiss, 1985). Finally, note that both

simulation and classification models can be quantitative (Nilsson, 1965), so the classification/simulation and qualitative/quantitative distinctions are orthogonal.

## 5. Example: Applying the model perspective to diagnosis

A diagnosis explains why observed system behavior is different from expected, generally in terms of a structural variation from the design. Corresponding to the types of qualitative models (Figure 4-1), the diagnostic process takes different forms:

- recognizing a disorder as a set of features;
- constructing an "historical" accounting for behavior (usually partial because only abnormal manifestations are explained, not normal functioning of the system); and
- constructing a "complete" description of the system being diagnosed (accounting for both abnormal and normal system behavior in terms of structural components).

As a model, a medical diagnosis is not just the name of a disease, but a *causal story* that relates the manifestations that need to be explained (because they are abnormal) to the processes that brought them about. Such an argument, shown as an inference network in Figure 5-1, is called a *patient-specific model* (Patil, 1981). In simple terms, it copies over from models of system processes, states, function, and structure the concepts and relations that are believed to hold in this particular problem.

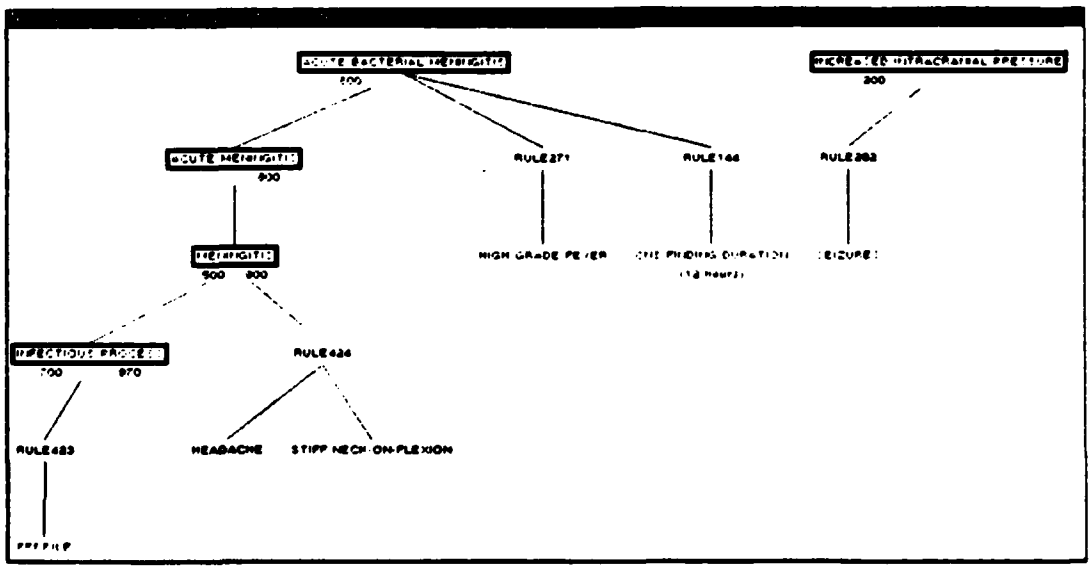


**Figure 5-1:** Partial diagnostic model in NEOMYCIN

The process of diagnosis can be viewed as the construction of a "proof tree." Proceeding upwards, the explanation becomes more specific in terms of the cause or subtype of the process that is occurring. At some intermediate stage when solving the problem, the network is disconnected and partial (Figure 5-1). The state-transition model indicates that seizures might be caused by increased intracranial pressure; this link is placed in the patient-specific model.

There is evidence for acute meningitis, but the alternative hypotheses have not been related. Is there some underlying cause or process that could account for all of the manifestations? Could meningitis cause increased intracranial pressure? Thus, the problem solver returns to the general model to search for connections that will allow a coherent explanation to be constructed.

While this description of diagnosis has intuitive appeal, most expert systems and their developers never check to see whether all of the findings are covered by the final "diagnosis." Cursory examination of the inference network (as shown for NEOMYCIN in Figure 5-1) reveals surprising gaps in the program's model: Abnormal findings needing to be explained are found to be unrelated to the most likely diagnosis. How can we account for the fact that these gaps go unnoticed?

In part, our language is too loose: The program prints out the *name* of a disorder, and we say, "The program has *made* a diagnosis." We don't think of a diagnosis as a description of a system and how it evolved. Instead, emphasis is on "inferring the right answer." We view inference in terms of adding up belief or finding a linear path of assertions that ends in the right diagnosis.

Accumulating evidence by some scoring function (Pople, 1982) or certainty factor scheme—the predominant approach in expert system diagnosis—disguises the *structural* aspects of diagnosis, that is, how the hypotheses explain the findings. It would be difficult to find a better example of the proverbial groping around the elephant, with each researcher proclaiming a different aspect of the nature of diagnosis:

- In INTERNIST, a scoring function reduces a hypothesis score by the "importance" of the observed findings it fails to explain (Pople, 1982).
- Viewing diagnosis in terms of traversing a network of processes and states, CASNET extracts the longest path from findings to a disease process (Weiss, et al., 1978).
- Reggia defines the best diagnosis as one that satisfies the principle of parsimony, covering the largest set of findings (Reggia, et al., 1984).
- In CADUCEUS, orthogonal relations and the possibility of multiple disorders make linear network traversal combinatorially intractable. Operators piece together state and process descriptions according to subtype and causal relations (Pople, 1982).
- In NEOMYCIN, diagnostic operators focus on, group, refine, and support the *differential*, the most-specific state and process descriptions that explain the findings (Clancey, 1984).
- Finally, in ABEL, these ideas are brought together, so that a diagnosis is a constructed graph that explains the findings on multiple levels of detail (Patil, 1981).

Thus, research has progressed from viewing diagnosis in terms of inferring the name of disease by chains of reasoning (MYCIN), or finding the best match (INTERNIST), to *reasoning about* a constructed inference network, the causal story. Inference is described not in terms of particular evidence rules and backchaining, but in terms of operators for manipulating the

patient-specific model.

An interesting result is that the uncertainty of the diagnosis no longer resides in a numeric score, but in the structure of the model: Alternative hypotheses are ranked by how well they cover the findings (Patil, 1981). Furthermore, by taking into account the *purpose* of the model, the issue becomes not to derive a precise measure of belief, but to determine if the uncertainty needs to be resolved. Medical diagnoses, like all engineering models, are partial. They need only be good enough to discriminate among choices for action, perhaps allowing for monitoring successful completion of a plan and improving an inadequate choice (Petroski, 1985). However, today's programs make a diagnosis independently of how it will be used, or therapeutic distinctions are implicit in the disorder classification. For example, MYCIN does not distinguish among types of viral meningitis because they are treated equivalently. Further development of this point would contrast the engineering demands of diagnosis with *scientific* modeling of systems (as in Dendral (Buchanan, et al., 1969)), which seeks mechanistic detail, and naive device models (Kieras, 1984, Suchman, 1985).

Even if a knowledge base lacks the detailed causal relations that enable reasoning about interactions as in ABEL, the patient-specific model can be useful. First, it can be used as a global perspective for controlling inference. For example, NEOMYCIN's patient-specific model indicates that unnecessary assertions are being made. Several new metarules prevent this, for example (stated negatively for clarity), "If a new disease hypothesis does not explain all abnormal findings, but some existing hypothesis does, then do not add the new hypothesis to the patient-specific model."

Second, as a form of output, the model (Figure 5-1) is a powerful knowledge acquisition tool. It reveals:

- **Gaps in the knowledge base**, evident by a missing link between an abnormal finding an hypothesis (a feature we are exploiting in our teaching program, GUIDON-DEBUG (Clancey, et al., 1986))
- **Implicit inferences**, evident by the program's inability to link terms such as "headache chronicity" and "cns finding duration" (something we must correct for the program to explain its reasoning).
- **Missing qualitative abstractions**, evident by a direct link between a numeric finding (e.g., "csf protein is 100") and a disease. We must make explicit what ranges are high or low and what is abnormal in order for the program to know what findings need to be explained.
- **Undirected causal links**, where no distinction is made between causes or correlated predispositions and effects (e.g., the patient's age is never an effect of a disease). Again, the program cannot know if its explanation is complete if it doesn't know what facts need to be explained.
- **Side-effects**, evident by links composing a causal relation with a model-manipulation strategy (e.g., if the csf is bloody, rule out bacterial-meningitis). The model perspective disciplines the rule writer to record the correct causal association (bloody csf is caused by a subarachnoid hemorrhage) and to write control rules that reason about alternative explanations (such as the metarule shown above).

These observations do not mean that the classification/behavioral approach should be abandoned in favor of structure-function models. First, diagnosis based on structure-function has generally been restricted to component failures, rather than interactions the system has with its environment. Given the open nature of the world, an "experiential" classification model appears to be practical, and perhaps necessary. Second, regardless of diagnostic value, classification and behavioral descriptions represent what people know in domains such as medicine and everyday reasoning (Roech, 1978, Feltovich, et al., 1984, Kolodner, 1984). Third, belief maintenance descriptions of inference (e.g., (De Kleer and Brown, 1984)) emphasize model coherence, but do not capture the structural aspects of situation-specific model manipulation, evident in the description of diagnostic tasks as graph-construction operators (Patil, 1981, Pople, 1982, Clancey, 1984). In applying the model perspective to classification and behavioral descriptions, we are not dismissing them as inferior, but rather recognizing their status as models and gaining a measure of quality for diagnostic inference.

Finally, the model-based perspective of Figure 5-1 is compatible with the blackboard paradigm (Hayes-Roth, 1984), viewing reasoning in terms of: A shared database; flexible, opportunistic operators for posting and modifying solution elements; and a separate control structure for scheduling inference. However, the blackboard is not *just* a database. It is a situation-specific model of a system, made apparent by the use of different blackboards or "panels" to separate different systems and action plans. The study of alternative kinds of models (Section 4) and the view of inference as model-manipulation operators is a step towards formalizing principles for structuring a blackboard and describing what "knowledge sources" do. This analysis also suggests that heuristic classification is not adequate for diagnosis and other systems problems when pre-enumerated processes or action plans can occur together and interact.

## 6. Historical perspective

We can resolve some of the uncertainty about the nature of expert systems and AI in general by relating this work to other science and engineering perspectives, placing it in historical context, and defining it comparatively. This is the chapter that AI textbooks consistently omit. We need to tell students not just what our goals and methodology are, but how we got here and how what we're doing is different from traditional science and engineering.

Today we take for granted that the words "machine" and "intelligence" go together because the idea of a machine today allows for adaptivity (McCorduck, 1979). But this was not always so. For scientists and engineers of the past, a "mechanism" was the antithesis of change and flexibility. It was something fixed, composed of known parts, and fully predictable. The "scientific perspective" in part had its origin in this mechanistic or deterministic conception of nature. Thus, the mysterious concepts of "a superhuman final cause" and "purpose" were rejected, and all observed behavior reduced to separate, discrete parts studied in isolation

(Frank, et al., 1948, Bertalanffy, 1968).

However, in this century, this view was found to be inadequate for understanding complex systems. The concept of goal was re-introduced in cybernetics by models of self-regulation, based on the concept of feedback (Wiener, 1961). Several sciences, including particularly biology and economics, turned to the study of systems not as "constitutive characteristics," of elements in isolation, but as relations among elements (Bertalanffy, 1968). Thus, study turned from invariant properties (such as molecular weight) to properties of a *complex*, dependent on relations among interacting elements (such as *topological structure* of a molecule and the *function* of molecular units).

In modeling the *open* and *non-linear* character of systems, analytic techniques broadened to include concepts such as: "compartmentalization" (near decomposability (Simon, 1969)), topology, adaptiveness, information flow, state-transition, and rationality (Bertalanffy, 1968, Newell and Simon, 1972). In particular, these perspectives address the need to represent discontinuities (non-linearity) in system behavior: "Representation by differential equations is too restricted for a theory to include biological systems and calculating machines where discontinuities are ubiquitous" (Bertalanffy, 1968).

From this perspective, Artificial Intelligence merges the *teleological concepts* of choice, regulation, adaptiveness and rationality with the *modeling concepts* of set, graph, net, and automata theory and formal logic (Newell and Simon, 1972). Furthermore, emphasis has shifted from modeling systems in themselves, to modeling "self-directed personalities" (Frank, et al., 1948) that incorporate a model of the world, goals, and a reasoning process for solving particular problems. The idea of the *autonomous agent* crucially frames our research. To determine what could be the basis for rationality and adaptation, we have focused on the *individual* interacting with the world to solve some problem. In order to *generate* intelligent behavior in an autonomous agent, a program, we put inside a *representation* of the agent's knowledge of the world and his goals. Rather than proving theorems about structures in themselves (as in graph theory), we use them to *represent* concepts and relations. Thus have evolved the well-known representational structures of our field: the augmented-transition network, the procedural network, the causal-association network, the semantic network, and so on.

In this historical context, we see that AI combines a framework for modeling (the idea of the autonomous agent) with qualitative modeling techniques. In describing AI, we have tended to emphasize the idea of generating intelligent behavior, and not made clear how our modeling techniques are different. Rather than numeric *measures*, we have devised methods for *describing* physical and cognitive systems. Our programs describe a system's physical appearance (structure), behavior, goals, and role in a larger context (function). These relations are distinguished in our advanced representation languages (Bobrow, 1984, Brachman and

Levesque, 1985).

In the buzzword "knowledge," we emphasize that what the problem solver knows or believes is central, providing a crucial focus for our research. But the term "knowledge" emphasizes the *role* of a representation, obscuring its primary status as a model of a system in the world. The discussion in Sections 3 and 5 gives examples of how our understanding of inference, uncertainty, and knowledge base design can be improved by applying the systems model perspective.

In general, the idea that knowledge bases, like all models, are selective, based on assumptions, and prone to failure has been given almost no consideration in knowledge engineering research. Perhaps the most basic engineering problem in constructing a model is to determine the range of its applicability. How can we determine what cases—situations in the world—will not be successfully modeled? Just as the structural engineer must ensure that his model of a bridge or building will be adequate for every real world event that occurs (given some assumptions) (Petroski, 1985, Weinberg, 1982), the knowledge engineer must certify the validity of his model. Yet, outside of established domains with a strong theory, such as electronics, no strong statements can be made about the accuracy of expert systems, except perhaps that they will work on the cases they have been tested on. While expert systems may only be designed to work for "typical cases," we must find some way to describe the extremes and articulate procedures for detecting when they occur. Until we shift from continuously blessing our programs with the name of "expert," and realize that they are only models, it is unlikely that the required engineering methodology will develop.

## 7. Conclusions

Focusing on the function of knowledge bases as models of systems, I have described expert-system problem solving in terms of generic tasks, provided a unifying description of qualitative representations, and related AI to the concerns of traditional science and engineering. Using medical diagnosis as an example, I showed the benefits of this perspective for detecting gaps in a knowledge base and understanding inference in terms of a model-manipulation strategy.

All knowledge bases contain qualitative models. Describing expert systems has been difficult because a "knowledge base" could be almost anything. Now, confronted with a knowledge base, you might ask: What system is it a model of? Whose model is it? Why is it believed? Is it a classification, state-transition, functional, or structural model? Are these composed? What is the inference procedure? Are situation-specific models selected or constructed? What is the explanatory/predictive or system synthesis capability? For what world? For what task? Under what assumptions? How do you know when it isn't good enough?

Along these lines, problem solving, learning, and explanation can be reconceptualized in terms of model acquisition, representation, and use. Many researchers have already adopted

this perspective (for example, see (Weiss, et al., 1978, Gaschnig, 1980, Swartout, 1981, Kahn, et al., 1985, Soo, et al., 1985)). But our work has barely begun in developing this new science and engineering of qualitative models:

We have explored a range of systems of different types: physical, perceptual, cognitive, and social. But the modeling methods are not practically accessible to professionals in other fields.

We now can see that the *locus* of a model can be strikingly varied: mathematical equations, a physical replica, a computer program, and a mental model. But we are struggling to understand what a representation is and how it might be confused with "reality" (Smith, 1982, Smith, 1985, Sloman, 1985).

We have developed qualitative models in expert systems for scientific and engineering applications (Sriram and Rychener, 1986). But we have not developed a methodology for testing models that distinguishes between these goals.

We have qualitative models of systems in the world, second order models (e.g., student models), and discourse and instructional models for communication and modification of models. But we are uncertain about how to represent these explicitly and to what extent they can be separated.

We continue to study the evolution of models, their origin and development in the individual, with recent successes in failure-driven and explanation-based learning (e.g., (Kolodner, 1984, Mitchell, et al., 1985)). But this has underscored how models from different domains and different perspectives interact in learning, in sharp contrast with the sparseness of today's knowledge bases.

We have developed a repertoire of computational methods for representing qualitative models and inference methods. But the relation of these methods to numeric techniques and their psychological aspects remain to be understood (Gentner and Stevens, 1983, Kuipers, 1985, Patel and Groen, 1986, Rouse and Morris, 1985).

Knowledge engineering is not just a new kind of programming. It is a new methodology for constructing models of systems in the world so they can be automatically assembled, modified, and controlled. We are not so much programmers as engineers, scientists, and perhaps philosophers. If we view our work in this way, consistently attempting to unify it from a modeling perspective, our claims and successes will be better understood.

# References


Achinstein, P. *The Nature of Explanation.* New York: Oxford University Press 1983.

Ackoff, R.L. Towards a system of systems concepts. In J.D. Couger and R.W. Knapp (editors), *System Analysis Techniques*, pages 27-38. John Wiley & Sons, New York, 1974.

Apte, C.V. andd Weiss, S.M. An approach to expert control of interactive software systems. *Pattern Analysis and Machine Intelligence*, September 1985, *7(5)*, 586-591.

Bennett, J. S. and Engelmore, R. S. *SACON: A Knowledge-based Consultant for Structural Analysis*, in *Proc. IJCAI-79*, pages 47-49, IJCAI, Tokyo, Japan, August, 1979. Originally published as a Techreport in 1978 and as an Article in 1979.

von Bertalanffy, L. *General System Theory: Foundations, Development, Applications.* New York: George Braziller, Inc. 1968.

Bobrow, D. G. Special issue on qualitative reasoning about physical systems. *Artificial Intelligence*, 1984, *24(1-3)*, 1-5.

Brachman, R.J. and Levesque, H.J. (eds.). *Readings In Knowledge Representation.* Los Altos: Morgan Kaufmann Publishers Inc. 1985.

Brown, J. S. and Burton, R. R. Multiple Representations of Knowledge for Tutorial Reasoning. In D. G. Bobrow and A. Collins (editor), *Representation and Understanding: Studies in Cognitive Science*, pages 311-350. Academic Press, New York, 1975.

Brown, J. S., Burton, R. R., and Zydbel, F. A model-driven question-answering system for mixed-initiative computer-assisted instruction. *IEEE Transactions on Systems, Man, and Cybernetics*, 1973, *SMC-3(3)*, 248-257.

Brown, J. S., Burton, R. R., and de Kleer, J. Pedagogical, natural language, and knowledge engineering techniques in SOPHIE I, II, and III. In D. Sleeman and J. S. Brown (editors), *Intelligent Tutoring Systems*, pages 227-282. Academic Press, London, 1982.

Buchanan, B. G., Sutherland, G., and Feigenbaum, E. A. Heuristic Dendral: A program for generating explanatory hypotheses in organic chemistry. In B. Meltzer and D. Michie (editors), *Machine Intelligence*, pages 209-254. Edinburgh University Press, Edinburgh, 1969.

Bylander, T. and Chandrasekaranm, B. *Understanding behavior using consolidation*, in *Proceedings of the Ninth Joint Conference on Artificial Intelligence*, pages 450-454, Los Angeles, August, 1985.

Chandrasekaran, B. and Milne, R. (eds.). Special section on reasoning about structure, behavior and function. *SIGART Newsletter*, July 1985, *93*, 4-7.

Clancey, W. J. *Transfer of rule-based expertise through a tutorial dialogue.* PhD thesis, SU, August, 1979.

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

Clancey, W. J. *The advantages of abstract control knowledge in expert system design*, in *Proceedings of the National Conference on AI*, pages 74-78, Washington, D.C., August, 1983.

Clancey, W. J. *Acquiring, representing, and evaluating a competence model of diagnosis.* HPP Memo 84-2, Stanford University, February 1984. (To appear in M. Chi, R. Glaser, and M. Farr (Eds.), *Contributions to the Nature of Expertise*, in preparation.).

Clancey, W. J. Heuristic Classification. *Artificial Intelligence*, December 1985, *27*, 289-350.

Clancey, W. J. and Letsinger, R. NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching. In Clancey, W. J. and Shortliffe, E. H. (editors), *Readings in Medical Artificial Intelligence: The First Decade*, pages 361-381. Addison-Wesley, Reading, 1984.

Clancey, W. J., Richer, M., Wilkins, D., Barnhouse, S., Kapsner, C., Leserman, D., Macias, J., Merchant, A., Rodolitz, N. *Guidon-Debug: The student as knowledge engineer.* KSL Report 86-34, Stanford University, 1986. (Submitted to special issue of *Journal of Artificial Intelligence* on instructional programs.).

Davis R. *Applications of meta-level knowledge to the construction, maintenance, and use of large knowledge bases.* HPP Memo 76-7 and AI Memo 283, Stanford University, July 1976.

Davis, R. Reasoning from first principles in electronic troubleshooting. In M.J. Coombs (editor), *Developments in Expert Systems*, pages 1-21. Academic Press, New York, 1984.

Davis, R. Knowledge-based systems. *Science*, February 1986, *231*, 957-963.

de Kleer, J. *Casual and teleological reasoning in circuit recognition.* PhD thesis, Massachusetts Institute of Technology, January, 1979. Also Report No. AI-TR-529.

De Kleer, J. and Brown, J.S. A qualitative physics based on confluences. *Artificial Intelligence*, December 1984, *24(1-3)*, 7-83.

Feigenbaum, E. A. *The art of artificial intelligence: I. Themes and case studies of knowledge engineering*, in *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pages 1014-1029, August, 1977.

Feigenbaum, E.A. and Feldman, J. *Computers and Thought.* Malabar, Florida: Robert E. Krieger Publishing Company, Inc. 1963.

Feltovich, P. J., Johnson, P. E., Moller, J. H., and Swanson, D. B. The role and development of medical knowledge in diagnostic expertise. In W. J. Clancey and E. H. Shortliffe (editors), *Readings in Medical Artificial Intelligence: The First Decade*, pages 275-319. Addison-Wesley Publishing Company, Reading, 1984.

Fink, P.K. *Control and integration of diverse knowledge in a diagnostic expert system*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 427-431, 1985.

Forbus, K.D. Qualitative Process Theory. *Artificial Intelligence*, December 1984, *24(1-3)*, 85-168.

Frank, L.K., Hutchinson, G.E., Livingstone, McCulloch, W.S, Wiener, N. Teleological Mechanisms. *Ann. N.Y. Acad. Sci.*, 1948, *50*, .

P. Friedland (ed.). Special section on architectures for knowledge-based systems. *Communications of the ACM*, September 1985, *28(9)*, 902-941.

Gaschnig, J. *Development of uranium exploration models for the prospector consultant system.* Technical Report Final Report, SRI International, March 1980.

Genesereth, M. R. *Diagnosis using hierarchical design models*, in *Proceedings of the National Conference on AI*, pages 278-283, Pittsburgh, PA, August, 1982.

Gentner, D. and Stevens, A. (editors). *Mental models.* Hillsdale, NJ: Erlbaum 1983.

Harmon, P. and King D. *Expert Systems: Artificial Intelligence in Business.* New York: John Wiley & Sons 1985.

Hasling, D. W., Clancey, W. J., Rennels, G. R. Strategic explanations in consultation. *The International Journal of Man-Machine Studies*, 1984, *20(1)*, 3-19. Also in Development in Expert Systems, ed. M.J. Coombs, Academic Press, London.

Hayes-Roth, B. *BB1: An architecture for blackboard systems that control, explain, and learn about their own behavior.* HPP Memo 84-16, Stanford University, December 1984.

Hayes-Roth, F., Waterman, D., and Lenat, D. (eds.). *Building Expert Systems.* New York: Addison-Wesley 1983.

Johnson, W. Lewis, and Elliot Soloway. *Intention-Based Diagnosis of Programming Errors*, in *Proceedings of the National Conference on AI*, pages 162-168, Austin, TX, August, 1984.

Kahn, G., Nowlan, S., and McDermott, J. *MORE: An intelligent knowledge acquisition tool*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 581-584, Los Angeles, August, 1985.

Kieras, D. E. *A simulation model for procedure inference from a mental model for a simple device.* Technical Report UARZ/DP/TR-84/ONR-15, University of Arizona, May 1984.

Kolodner, J.L. Towards an understanding of the role of experience in the evolution from novice to expert. In M.J. Coombs (editor), *Developments in Expert Systems*, pages 95-116. Academic Presss, New York, 1984.

Kosslyn, S. M. *Image and Mind.* Cambridge: Harvard University Press 1980.

Koton, P.A. *Empirical and model-based reasoning in expert systems*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 297-299, 1985.

Kuipers, B. *The limits of qualitative simulation*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 128-136, 1985.

Lehnert, W.G. *Representing physical objects in memory.* Techreport 131, Computer Science Department, Yale University, 1978.

McCorduck, P. *Machines Who think: A Personal Inquiry Into The History And Prospects Of Artificial Intelligence.* San Francisco: W.H. Freeman and Company 1979.

McDermott, J. *Extracting knowledge from expert systems*, in *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 100-107, Karlsruhe, West Germany, August, 1983.

Minsky, M. Steps toward Artificial Intelligence. In E.A. Feigenbaum and J. Feldman (editors), *Computers and Thought*, pages 406-450. Robert E. Krieger Publishing Company, Inc., Malabar, Florida, 1963.

Minsky, M. A framework for representing knowledge. In P. H. Winston (editor), *The Psychology of Computer Vision*, . McGraw-Hill, New York, 1975.

Mitchell, T.M., Mahadevan, S., Steinberg, L.I. *LEAP: A learning apprentice for VLSI design*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 573-580, Los Angeles, August, 1985.

Newell, A. and Simon, H. A. *Human Problem Solving*. Englewood Cliffs: Prentice-Hall 1972.

Nilsson, N.J. *Learning Machines: Foundation of Trainable Pattern-classifying Systems*. New York: McGraw Hill Book Company 1965.

Padulo, L. and Arbib, M.A. *System Theory: A unified state-space approach to continuous and discrete systems*. Washington: Hemisphere Publishing Corporation 1974.

Patel, V.L. and Groen, G.J. Knowledge based solution strategies in medical reasoning. *Cognitive Science*, January-March 1986, *10(1)*, 91-116.

Patil, R. S., Szolovits, P., and Schwartz, W. B. *Causal understanding of patient illness in medical diagnosis*, in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 893-899, August, 1981.

Pauker, S. G., Gorry, G. A., Kassirer, J. P., and Schwartz, W. B. Toward the simulation of clinical cognition: taking a present illness by computer. *AJM*, 1976, *60*, 981-995.

Petroski, H. *To Engineer is Human: The role of failure in successful design*. New York: St. Martin's Press 1985.

Pople, H. Heuristic methods for imposing structure on ill-structured problems: the structuring of medical diagnostics. In P. Szolovits (editor), *Artificial Intelligence in Medicine*, pages 119-190. Westview Press, 1982.

Reggia, J.A., Nau, D.S. and Wang, P.Y. Diagnostic expert systems based on a set covering model. In *Developments in Expert Systems*, pages 35-58. Academic Press, New York, 1984.

Rich, E. User modeling via stereotypes. *Cognitive Science*, 1979, *3*, 355-366.

Rieger, C. and M. Grinberg. *The Declarative Representation and Procedural Simulation of Causality in Physical Mechanisms*, in *Proc. IJCAI-77*, pages 250-256, IJCAI, MIT, Cambridge, MA, August, 1977.

Rosch, E. Principles of categorization. In E. Rosch and B. B. Lloyd (editors), *Cognition and Categorization*, pages 27-48. Lawrence Erlbaum Associates, Hillsdale, NJ, 1978.

Rouse, W.B. and Morris, N.M. *On looking into the black box: prospects and limits in the search for mental models*. Technical report 85-2, School of Industrial and Systems Engineering, Georgia Institute of Technology, May 1985.

Sacerdoti, E. D. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 1974, *5(2)*, 115-135.

Schank, R. C., and Abelson, R. P. *Scripts, Plans, Goals, and Understanding*. Hillsdale, NJ: Lawrence Erlbaum Associates 1975.

Shortliffe, E. H. *Computer-based medical consultations: MYCIN*. New York: Elsevier 1976.

Simon, H. A. *The Sciences of the Artificial*. Cambridge: The MIT Press 1969.

Sloman, A. *What enables a machine to understand*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 995-1001, 1985.

Smith, B.C. The computational methaphor. .

Smith, B. *Models in expert systems*, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 1308-1309, 1985.

Soloway, E.M, Woolf, B., Rubin, E., Barth, P. *Meno-II: An intelligent tutoring system for novice programmers*, in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 975-977, Vancouver, August, 1981.

Soo, V., Kulikowski, C.A., and Garfinkel, D. *Qualitative modeling and clinical justification for experimental design*, in *Proceedings of the International Conference on Artificial Intelligence in Medicine*, pages 21-36, Amsterdam, 1985. Participants edition.

Sriram, D. and Rychener, M.D. (eds.). Expert systems for engineering applications. *IEEE Software*, March 1986, *3(2)*, 4-5.

Stevens, A., Collins, A. and Goldin, S.E. Misconceptions in students' understanding. In D. Sleeman and J.S. Brown (editors), *Intelligent Tutoring Systems*, pages 13-24. Academic Press, New York, 1982.

Suchman, L.A. *Plans and situated actions: the problem of human-machine communication.* Technical report ISL-6, Xerox Parc, February 1985.

Swartout W. R. *Explaining and justifying in expert consulting programs*, in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 815-823, August, 1981.

Szolovitz, P. *Types of knowledge as bases for reasoning in Medical AI Programs*, in *Proceedings of the International Conference on Artificial Medicine in Medicine*, pages 31-48, Pavia, September, 1985.

Weinberg, G.M. *Rethinking Systems Analysis and Design*. Boston: Little, Brown and Company 1982.

Weiss, S. M. and Kulikowski, C. A. *EXPERT: A system for developing consultation models*, in *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, pages 942-947, August, 1979.

Weiss, S. M. and Kulikowski, C. A. *A Practical Guide to Designing Expert Systems*. Totowa, NJ: Rowman and Allanheld 1984.

Weiss, S. M., Kulikowski, C. A., Amarel, S., and Safir, A. A model-based method for computer-aided medical decision making. *Artificial Intelligence*, 1978, *11*, 145-172.

Wiener, N. *Cybernetics*. Cambridge, MA: MIT Press 1961.