(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

COGNITIVE FEEDBACK AS
A TOOL FOR
KNOWLEDGE ACQUISITION

by

Charles Allen Patterson

September 1990

Thesis Advisor: Kishore Sengupta

Approved for public release; distribution is unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (If applicable) Code 54	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

COGNITIVE FEEDBACK AS A TOOL FOR KNOWLEDGE ACQUISITION

12. PERSONAL AUTHOR(S) Patterson, Charles A.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Master's Thesis	FROM ______ TO ______	September 1990	143

16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Knowledge Acquisition, Cognitive Feedback, Expert Systems.

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Knowledge acquisition is often considered a "bottleneck" in the development of expert systems. This study conducted a review of 14 knowledge acquisitions methods with a survey of knowledge types, task characteristics, and representation schemes. All of the knowledge acquisition techniques are considered deficient in their ability to capture a representation of an expert's mental model and procedural knowledge.

Cognitive feedback and the lens model, drawn from Egon Brunswik's probabilistic functionalism, are proposed as an alternate knowledge acquisition methodology. Cognitive feedback's theoretical underpinnings are explained as are the various uses to which it has been put. A summary of the many research studies conducted into the effectiveness of cognitive feedback is presented. An automated knowledge acquisition tool using cognitive feedback is proposed and illustrated with state transition diagrams and sample computer screens.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL Professor Kishore Sengupta	22b. TELEPHONE (Include Area Code) (408) 646-3212	22c. OFFICE SYMBOL Code AS/Se

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

U.S. Government Printing Office: 1986—606-24

Unclassified

Cognitive Feedback as a Tool for Knowledge Acquisition

by

Charles Allen Patterson
Lieutenant, United States Navy
B.A., University of California, Los Angeles, 1982

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author: Charles Allen Patterson
Charles Allen Patterson

Approved by: Kishore Sengupta
Kishore Sengupta, Thesis Advisor

Tung Bui
Tung X. Bui, Second Reader

David R. Whipple, Chairman
Department of Administrative Sciences

ABSTRACT

Knowledge acquisition is often considered a "bottleneck" in the development of expert systems. This study conducted a review of 14 knowledge acquisition methods with a survey of knowledge types, task characteristics, and representation schemes. All of the knowledge acquisition techniques are considered deficient in their ability to capture a representation of an expert's mental model and procedural knowledge.

Cognitive feedback and the lens model, drawn from Egon Brunswik's probabilistic functionalism, are proposed as an alternative knowledge acquisition methodology. Cognitive feedback's theoretical underpinnings are explained as are the various uses to which it has been put. A summary of the many research studies conducted into the effectiveness of cognitive feedback is presented. An automated knowledge acquisition tool using cognitive feedback is proposed and illustrated with state transition diagrams and sample computer screens.

TABLE OF CONTENTS

I. INTRODUCTION

A. GENERAL

Expert Systems are rapidly gaining in popularity throughout a wide range of applications. The fascination with these systems is evidenced by the increasing number of expert system related articles appearing in academic journals and business publications (Olson and Rueter, 1987). Successful expert systems are currently being used to solve problems in such diverse fields as space shuttle crew planning, oil drilling, tactical air targeting, tax planning, Soviet radar systems identification, and wine selection (Waterman, 1986).

An expert system is a computer system that uses the experience and knowledge of one or more experts within a particular problem domain. The expert system's *knowledge base* is its store of domain specific knowledge and is symbolically represented, usually but not always, in the form of facts or rules. The knowledge base is kept separate from the reasoning mechanism or *inference engine*. These are the methods by which the symbolic knowledge is manipulated to arrive at a solution. *Knowledge engineering* is the term given to the entire process of information accumulation, representation, and manipulation. Central to the knowledge engineering process is the acquisition of knowledge from an expert. (Boose, 1986)

The acquisition of knowledge is one of the most difficult steps in the development of a knowledge base. This is due to the differences that exist between an expert's knowledge and what he or she can successfully articulate. What cannot be articulated is known as implicit knowledge and few methods are able to extract this from an expert. A gap is created in the body of knowledge acquisition techniques because most of the methods rely on the expert to consciously access information or they *assume* some underlying organization. The cause of this gap is that some knowledge may not be consciously accessible or known, *a priori*, to exist in a particular form. (Berry, 1987)

Cognitive feedback is a technique that returns some elements of output from a decision maker's cognitive processes, enabling the decision maker to solve a problem more effectively. This process captures and refines the decision makers judgement rules, permitting the application of the knowledge again at some point in the future. Research has proven the effectiveness of this technique as an aid for representing domain knowledge (Balzer, Doherty, and O'Connor, 1989). Cognitive feedback is proposed to fill the gaps that exist in the body of knowledge acquisition techniques.

1. The Importance of Knowledge Acquisition

The knowledge in an expert system may be derived from many different sources: textbooks, reports, data bases, case studies, empirical data, and personal experience. The primary source, however, is the domain expert, the individual with the expertise in the field of interest. The knowledge engineer must usually obtain this knowledge through direct interaction with the expert. (Boose, 1986)

The knowledge acquisition phase is central to the development of an expert system because the power and utility of the resulting product is dependent upon the quality of the underlying representations. The determinants of that quality seem to rest with domain knowledge rather than the complexity of the formal reasoning methods employed. This is because many difficult tasks resist the exact specifications necessary for traditional algorithmic methods. (Garg-Janardan and Salvendy, 1987)

2. The Problems of Knowledge Acquisition

Knowledge acquisition is the crucial first step in the development of expert systems. In traditional methods of standard software development most of the time is spent on coding. However, most time spent on the development of expert systems is consumed by this first step, the planning and deciding of what knowledge to include, followed by the actual extraction of the information from the expert. (Harmon and King, 1985; Waterman, 1986)

The actual elicitation of knowledge is highly problematic because experts possess much information that is cognitively complex, pragmatic, and tacitly formulated. Getting at this information is not possible in a standard interviewing situation that deals predominantly with facts without recourse to their schematic foundations. Often the expert will not be able to adequately access the knowledge so it can be easily represented in a program. It is also difficult to determine when that information is correct, consistent, and complete. (Berry, 1987)

3. Implications for Expert System Development

The time required by the knowledge acquisition phase consumes an inordinate amount of the six to 24 months it takes to develop an expert level prototype (Boose, 1985). The traditional approach to knowledge acquisition uses a knowledge engineer who typically spends a period of "apprenticeship" within the domain. The knowledge engineer must also be well versed in several fields of computer science and computer systems. Such individuals are becoming increasingly harder to find as the pace of demand outstrips the supply of knowledge engineers. (Shaw and Gaines, 1987a)

The reliance upon knowledge engineers creates problems for knowledge acquisition in other ways, as well. The knowledge engineer-expert interaction is often mismatched because the engineer is usually a novice at the outset. They will not be seeing the same thing even when both are discussing the same phenomena. This results in an inability to take account of the greater abstraction possessed by the expert and risks development of a shallow and narrowly useful knowledge base. The knowledge engineer, in effect, acts as an imperfect filter for the knowledge that is passed from the expert to the knowledge base. (Berry, 1987) This continuing reliance upon human labor is contrary to other trends within the industry. The human labor expenses become a dominating constraint as prices for system technology drop. (Shaw and Gaines, 1987a)

B. THE PROBLEM

The background has presented two primary problems with respect to the knowledge acquisition phase of expert systems development. First, the nature of expert knowledge is such that much time is spent on extracting it. Once obtained there is no guarantee that it truly reflects the knowledge actually held by the expert.

Second, the use of knowledge engineers in traditional roles hinders translation of expertise and knowledge from the expert to the knowledge base. The shortage of knowledge engineers only exacerbates this problem by contributing to project delay and increased cost.

C. THESIS OBJECTIVES

This study has three primary objectives. The first is to determine what gaps exist in current knowledge acquisition methodologies. Of special interest are those techniques which purport to extract the implicit or tacit types of knowledge.

A second objective of this thesis is to determine whether cognitive feedback is appropriate as a tool for filling the gaps that exist in knowledge acquisition. Effort is primarily directed toward determining if cognitive feedback is capable of extracting types of knowledge that other techniques cannot.

The third objective is to determine how cognitive feedback can be put to use for extracting an individual's knowledge. To this end, the high level specifications of an automated knowledge acquisition tool will be generated.

D. RESEARCH QUESTIONS

This study will address three primary questions. The first is: Can any of the current knowledge acquisition techniques satisfactorily elicit the special knowledge that experts possess? Secondarily:

- --What is the nature of an expert's knowledge?
- --Once the knowledge is elicited, what are the ways of representing this knowledge?
- --Can the nature of a task influence the knowledge acquisition strategy?
- --Are some acquisition techniques more suited to one type of task than another?
- --What are the strengths and weaknesses of each knowledge acquisition methodology?

The second question addressed by this thesis is: Can cognitive feedback be used in the knowledge extraction process? Secondarily:

--What is cognitive feedback and what are its theoretical underpinnings?
--How is cognitive feedback operationalized?
--Is cognitive feedback better than other types of feedback?
--How is the knowledge captured by cognitive feedback represented?
--Has cognitive feedback been empirically validated?
--Are there other uses for cognitive feedback?

The last question addressed is: Can cognitive feedback can be used in an automated knowledge acquisition tool? Secondarily:

--What is a likely high level specification for such a tool?
--How would the tool work?
--Can such a tool decrease reliance on knowledge engineers?

E. SCOPE

This research focuses on determining whether cognitive feedback can make a contribution to the field of knowledge engineering. The gaps uncovered in the current knowledge acquisition techniques, with information drawn from studies on cognitive feedback, will be considered for a high level specification of an automated knowledge acquisition tool. This research does not involve empirical studies of any kind. The actual coding of the automated knowledge acquisition tool is the subject of a follow-on thesis.

F. ORGANIZATION OF THE STUDY

This thesis is divided into three main sections excluding the introduction and conclusion. The first major section, Chapter II, is a survey of current knowledge acquisition techniques. The chapter begins with background regarding the various types of knowledge and how an expert's knowledge is unique. Different knowledge representation schemes and how these are important to knowledge acquisition are presented. Task characteristics are detailed along with a theory for their classification. The body of the chapter is composed of a brief summary and a list of advantages and disadvantages for each of 14 knowledge elicitation methods.

The second major section, Chapter III, concerns the theory from which cognitive feedback has been derived, Brunswik's probabilistic functionalism. The operationalization of the theory, in the lens model, is presented and the notion of feedback in general is discussed. Cognitive feedback is defined in detail and contrasted with another form of feedback, outcome feedback. The mathematical representations of the knowledge captured by cognitive feedback are outlined. Empirical studies regarding the effectiveness of cognitive feedback are summarized.

The third major section, Chapter IV, is a proposed automated knowledge acquisition tool that uses cognitive feedback. It is set within the context of a simple personnel evaluation for promotability task.

II. KNOWLEDGE ACQUISITION TECHNIQUES AND METHODOLOGIES: A SURVEY

A. INTRODUCTION

Knowledge Acquisition has long been recognized as the "bottleneck" in the development of expert systems (Harmon and King 1985; Waterman, 1986). The source of this difficulty lies in the varied and intricate nature of expert knowledge. Any expert has knowledge that is explicit and objective, as well as knowledge that is more implicitly formulated (Hawkins, 1983). The latter is usually very difficult for experts to articulate (Broadbent, Fitzgerald, and Broadbent, 1986). Knowledge acquisition techniques that consist of standard interview methods or unstructured think-aloud protocols may bias the knowledge engineer into fixating on those aspects of the task that can be well represented within if/then, rule based systems. The resulting knowledge base may then lack vital components of the expert's knowledge (Bainbridge, 1979).

It is important to recognize then, that a domain expert will possess knowledge of several different kinds (Berry, 1987). Each type of knowledge demands a technique that can most effectively capture it. The technique must transform this knowledge to a representation suitable for the inference strategy used in the problem solving process. Rather than use a single knowledge acquisition technique, several techniques should be employed, with each matched to a different kind of knowledge (Gammack and Young, 1985).

The nature of the task is an important feature that should be explored when the components of an expert's knowledge are investigated. The nature of the task is salient in that it determines the possible strategies an expert uses to complete or solve a task (Hogarth, 1974). When the particular problem solving task is isolated and identified, the type of knowledge necessary to solve that problem, independent of any particular implementation, should be analyzed and described. This enables the knowledge engineer to decompose the expert's compiled knowledge and to identify discrete tasks, types of knowledge being processed, and the relationships among the data, facts, and procedures (McGraw and Riner, 1987).

Before describing the role that cognitive feedback can play in the knowledge acquisition process it is necessary to identify where gaps in the current methodologies and techniques exist. This chapter presents a survey of the current state of the art in knowledge acquisition by first examining knowledge types, knowledge representations, and task types. This is followed by an analysis of different knowledge elicitation techniques.

B. TYPES OF KNOWLEDGE

Selection of the appropriate knowledge acquisition technique requires that the knowledge engineer recognize the type of knowledge within the domain under investigation. Major problems within knowledge engineering include recognition and analysis of domain knowledge and selection of an appropriate knowledge acquisition technique. There exists today several methodologies for the classification of knowledge, no one of which is universally accepted. McGraw and Riner employ a widely accepted scheme to classify knowledge into four basic types: procedural, declarative, semantic, and episodic. (McGraw and Riner, 1987)

1. Procedural Knowledge

This includes the skills that an individual knows. It may involve an automatic response to a stimulus, and can be reactionary in nature. Such skills are deeply ingrained and linked sequentially, one step serving as the trigger for completing the next. This knowledge is implicit and highly compiled so that the expert will have great difficulty in both identifying and verbalizing it and therefore is of primary interest to knowledge engineers (McGraw and Riner, 1987). When individuals master increasingly more knowledge to carry out a task efficiently, they also lose awareness of what they know. This has been called the "paradox of expertise" (Johnson, 1983).

Procedural knowledge is not necessarily motor in nature. Knowledge of one's native language is procedural. While most people have this knowledge, they find it difficult to describe precisely the rules of usage. This type of knowledge may also include that which is gained from implicit learning or an unconscious process such as socialization, perception, and the rules of complex games. (Gammack and Young, 1985)

2. **Declarative Knowledge**

This represents surface level information that experts can verbalize. The primary distinction between this and procedural knowledge is the ability to verbalize or express it. Declarative knowledge is what the expert is conscious of knowing. Therefore, it may not adequately reflect the cognitive foundations and concepts that will convey the expert's information in a meaningful way. This type of knowledge is relatively easy to acquire. (McGraw and Riner, 1987)

3. **Semantic Knowledge**

This represents one of the two theoretical types of long term memory. It reflects cognitive structure, organization, and representation. As a result it will be difficult for experts to express. Because this type of knowledge includes memories for vocabulary, concepts, facts, definitions, and relationships among facts, it too is of importance to knowledge engineers. It is semantic information that determines whether the expert system actually emulates the work of an expert in the given domain. It will present problems with regard to identification and retrieval. (McGraw and Riner, 1987)

4. Episodic Knowledge

This is autobiographical, experience-oriented information that the expert has grouped or chunked by episodes and is the second theoretical type of long term memory. It consists of information organized by time and place of occurrence, and often may be described in terms of perceptual characteristics. This is highly compiled information and is one of the most difficult types of knowledge to extract and dissect. Since this knowledge is chunked, the expert may or may not be aware of the separate knowledge entities and decision-making processes used to complete the task. (McGraw and Riner, 1987)

C. KNOWLEDGE REPRESENTATION SCHEMES

Selection of an appropriate knowledge representation scheme is critical in the development of an expert system. Psychological evidence suggests that there should be different representations for each type of knowledge. The representation scheme should aim at simulating the essentials of suspected basic mental models in humans. (Rouse and Morris, 1986)

An important distinction with respect to the form of mental models is whether they are spatial or verbal in nature. Since pattern recognition in human beings is highly developed, it is likely that the processing and storage of spatially oriented information is highly developed as well. Mental models may frequently be pictorial or graphic-like instead of symbolic, as in list-processing. This will present difficulties when experts attempt to verbalize their models. Additionally, the mental models may be dynamic objects with a variety of forms, even for a certain person in a precise situation. (Rouse and Morris, 1986)

The selection of a representation is therefore crucial. It must allow for both a natural mapping of the body of knowledge and an inference mechanism or algorithm that can effectively operate on that representation. A representation also should satisfy three requirements. First, it should have sufficient expressive power. Second, it should possess uniform readability. This means that an expression can be read independently of where it occurs in the program and independently even of the program itself. Third, it should ensure a preservation of structures: the many interconnections between pieces of knowledge must remain intact. These interconnections support the problem solving process. (Richter, 1986)

What follows is a brief discussion of the most common forms of knowledge representation: production rules, object-attribute-values, semantic nets, frames, decision trees, inference networks, and predicate logic.

1. Production Rules

A production rule is the term used by cognitive psychologists to describe an *if-then* rule. A *production system* has a data base of production rules and a control mechanism that selects applicable production rules to reach a goal state. A major use of these systems has been to model human cognition, specifically the problem solving techniques that involve a search process. Production systems are particularly suited to the representation of procedural knowledge. (Harmon and King, 1985)

2. Object-Attribute-Value Triplets

O-A-V triplets are useful for the representation of factual knowledge. An object is an actual or conceptual item within the expert's domain. An object's properties are called attributes and they can assume many different values. (Harmon and King, 1985)

This scheme is a specialized case of the semantic network described below. Complicated links are simplified in favor of just two relationships. The *object→attribute* link is a "has-a" link and the *attribute←value* link is an "is-a" link. (Harmon and King, 1985)

3. **Semantic Nets**

This is a type of knowledge representation that portrays objects and values as nodes. The nodes are connected with arcs or links that describe relationships between the many nodes. The nodes represent objects and descriptors and the links relate objects and descriptors. Some links are definitional while others may capture heuristics. Flexibility and inheritance are two major features of these networks that attempt to comprise categorical and role-related hierarchical organization of knowledge. (Harmon and King, 1985)

4. **Frames**

This representation scheme relates an object with a assemblage of features. Each of the features are saved in a slot. A frame is that collection of slots associated with a specific object. Slots may also contain default values, pointers to other frames, rule sets, or methods (procedural attachments) by which values may be obtained. When compared to traditional computer programming, a frame is similar to a property list, schema, or record. (Harmon and King, 1985)

Frames allow for more inventive representations of knowledge but they are also more sophisticated and difficult to develop than the simpler O-A-V or rule systems. Frames can join into a single representation scheme both procedural and declarative knowledge. This is known as situation related knowledge. (Harmon and King, 1985)

5. Decision Trees

A decision tree is similar to a flow chart, but has nodes and branches. Terminal nodes are those at the bottom while those above are intermediate nodes. A path, determined by the values of attributes described in the intermediate nodes, branches down the tree until a terminal node, or decision is reached (Hart, 1986). This structure is useful for describing expert information that may be stored in a hierarchical flow (Olson and Rueter, 1987).

6. Inference Networks

An inference network is a diagram consisting of boxes that represent attributes, or states and rules, and they are generated from rule based systems. Attributes within the network are data (i.e., observations, facts) that form preconditions to some rules and targets for others. The rules form a large inference net between attributes. All possible inference chains that can be generated from the rules can also be interpreted as connections between evidence and hypotheses. Inference networks are versatile tools but are best for small rule bases as they can quickly become very complicated. (Waterman, 1986)

7. Predicate Logic

Predicate Logic lends itself to mapping propositions about arbitrary objects into a theory with well known mathematical properties. This is an extension of propositional logic. Each elementary unit is called an object, and statements about the objects are called predicates. Logic provides a way to assert facts about the world, but seeking values in this system is not as direct as seeking values in the systems described above. This is because when a fact is stated it is either true or false. Still, a good deal of theoretical refinement can be achieved when logic is used to codify a suitable knowledge domain. (Harmon and King, 1985)

D. TASK CHARACTERISTICS

Identification of the application task characteristics is important because this will influence selection of the knowledge acquisition tool and the strategies to be applied in building and refining the knowledge base. The characteristics of the task affect the manner in which an expert will store and access task-critical knowledge, and will determine the problem-solving strategy. Expert knowledge is task centered, so analyzing the processing states and considerations an expert applies when performing a task or making a decision is key to attaining an initial understanding of the domain. An expert system should not merely capture a static representation of a knowledge domain, it also should simulate a particular problem-solving task within that domain. (Riesbeck, 1984)

Artificial Intelligence currently lacks a universal theory that will map existing expert system tools or shells onto the categories of tasks that they can solve (Kidd, 1987). Indeed, there is no universal theory that will categorize all possible types of problem-solving tasks. However, Kitto and Boose (1989) summarize a widely used classification of application tasks. Generic applications can be divided into the task categories of interpretation, prediction, diagnosis, design, planning, monitoring, debugging, repair, instruction, and control. These can then be gathered into two broad groups: those associated with *analysis* (interpretive) tasks, and those concerned with *synthesis* (constructive) tasks.

Analysis tasks include diagnosis, interpretation, debugging, and identification. Synthesis tasks include design, evaluation, configuration, scheduling, and planning. Analysis-Synthesis tasks include control, instruction, monitoring, prediction, and repair. (Hayes-Roth, Klahr, and Mostow, 1986)

Once a classification scheme for application tasks is selected, the appropriate problem solving method is identified. The knowledge acquisition tool then provides the link between the application task and the problem-solving methodology. The tool must elicit the information necessary to meet the special problem-solving requirements of an application task category. (Kitto and Boose, 1989)

Two basic problem solving methods exist: *heuristic classification* and *heuristic construction*. Each of these can be applied to a wide range of application tasks. Heuristic classification is best suited to analysis tasks. This is a method in which concepts in different classification hierarchies are heuristically related using a process of data abstraction, heuristic matching and solution refinement. Heuristic construction, on the other hand, is most appropriate to synthesis tasks. With this method, the problem solver constructs solutions, either by generating complete solutions or assembling solutions from components while satisfying constraints. (Clancey, 1986)

Table 1 illustrates what problem-solving methods are supported by existing knowledge acquisition systems in a particular application task category. For example, the knowledge acquisition systems MDIS, MORE, MOLE, TEIRESIAS, ROGET, and TKAW enable diagnostic application tasks to use the heuristic classification problem-solving method. Knowledge acquisition tools do not currently exist for certain application types within each category. Most current acquisition tools support one problem-solving method, but it is possible for a complex application to require several problem-solving methods to resolve the total problem. (Kitto and Boose, 1989)

TABLE 1. KNOWLEDGE ACQUISITION TOOLS LINK APPLICATION TASKS TO PROBLEM SOLVING METHODS.

Problem-Solving Method	Knowledge Acquisition Tool	Application Task Category
Heuristic Classification	MDIS, MOLE, MORE, ROGET, TKAW, TEIRESIAS	Analysis (Diagnosis)
	AQUINAS	Analysis (Diagnosis, Interpretation, Debugging, Identification)
	ETS, KITTEN, STUDENT	Analysis (Identification)
Heuristic Construction		
Propose-and-revise	SALT	Synthesis (Design)
Propose-and-apply	KNACK	Synthesis (Evaluation)
Skeletal plan refinement	OPAL	Synthesis (Planning)
Not	Supported	Synthesis (Configuration, Scheduling)
Not	Supported	Analysis-Synthesis (Control, Repair, Instruction, Monitoring, Prediction)

E. KNOWLEDGE ELICITATION TECHNIQUES

This section explores the actual techniques used in extracting knowledge from experts. It is exactly this phase that has become known as the "bottleneck" in the development of an expert system. This phase is also known as "knowledge extraction" and "knowledge acquisition". It refers to the transfer and transformation of problem-solving expertise from a knowledge source (i.e., human expert, documents) to a program (Hayes-Roth et al, 1986).

Several methods have been developed for knowledge acquisition, and no single technique is usually used to the exclusion of others. Sometimes a combination approach may be used while in other circumstances different techniques may be appropriate to different stages of the acquisition process. When selecting a specific technique a knowledge engineer should identify and isolate the problem-solving task to be simulated. Then the type of knowledge necessary to solve that problem should be described and analyzed, independent of any particular implementation. (Kidd, 1987)

The techniques described here are those most widely noted in the literature of knowledge acquisition and cognitive psychology, and those most widely used by knowledge engineers. Each is briefly described along with its advantages and disadvantages, type of knowledge or task it is most applicable to, and what existing automated tools are supported.

The first seven methods (Interview, Questionnaire, On-site Observation, Interruption Analysis, Protocol Analysis, Drawing Closed Curves, Inferential Flow Analysis) can be described as "direct". These techniques ask the expert to report on knowledge that he or she can directly articulate. They are free form, so the possibility exists for the knowledge engineer to find any type of information. The most likely type of knowledge that will be uncovered is declarative or surface knowledge. These methods will extract only what the expert is able to verbalize and overreliance on these techniques will exclude important information. (Olson and Rueter, 1987)

The next seven techniques (Multidimensional Scaling, Hierarchical Clustering, General Weighted Networks, Ordered Trees from Recall, Repertory Grid Analysis, Decision Analysis, Machine Induction) can be described as "indirect". These do not rely on the expert's ability to articulate the information used. They collect other behaviors from which the knowledge engineer makes inferences about what the expert must have known to perform as he or she did. These may uncover a deeper (procedural or semantic) knowledge, but will involve assumptions about the underlying form of the representation employed by the expert. Therefore, these techniques could be misused to the extent that their basic assumptions are not supported by the data. (Olson and Rueter, 1987)

1. Interviews

The interview is the most common technique for the elicitation of domain knowledge from an expert (Gammack and Young, 1985). Interviews quickly allow the knowledge engineer to grasp important domain concepts and vocabulary. The expert may reveal the objects he or she thinks about, how they are related, the judgmental processes used in solving a problem, and some inference rules (Olson and Rueter, 1987). This can be the most free form of the direct methods and is the most likely to uncover unexpected information. Most interviews are conducted in an unstructured form and will seldom provide complete or well organized descriptions of cognitive processes (Olson and Rueter, 1987).

The breadth and accuracy of what can be extracted in this free form style soon reaches a limit (Olson and Rueter, 1987). At this point the knowledge engineer should switch to focused or structured interviews that involve careful preplanning of the questions and their order. This represents a more goal oriented approach that may uncover additional data on factual knowledge, types of problems, functions of expertise, and explanations. (Kidd, 1987)

All interviews are most appropriate to uncovering only declarative or surface forms of knowledge. Since this technique ultimately relies on the expert's ability to articulate what he or she knows, much information will not be uncovered (i.e., procedural and semantic knowledge). This technique will not reveal how an expert's thinking or deeply compiled mental processing is conducted. Interviews therefore, are good only for the initial knowledge acquisition sessions (Olson and Rueter, 1987). There are several automated knowledge acquisition tools based on some form of the interview (see Table 2).

TABLE 2. KNOWLEDGE ACQUISITION TOOLS BASED ON INTERVIEW TECHNIQUES.

TOOL	SOURCE
MORE	Kahn, Nowlan, and McDermott (1985)
MOLE	Eshelman (1988)
SALT	Marcus (1987)
KNACK	Klinker, Boyd, Genetet, and McDermott, (1987)
KADS	Breuker and Weilinga (1987)
TEIRESIAS	Davis (1979)

2. Questionnaires

Questionnaires can be a much more efficient way of gathering information than interviews, which are very time consuming. These questions are not similar to statistical surveys, they are open-ended and very much like those presented in an interview. The expert usually feels more in control and can fill out the questionnaires at his or her convenience, without the pressure of a one-on-one session with a knowledge engineer. This technique is particularly useful in uncovering the objects of the domain with its relationships and uncertainties (if instructed to attach these to conclusions). (Olson and Rueter, 1987)

Most people are very poor at estimating probabilities, they will overestimate low ones and underestimate high ones (von Winterfeldt, 1988). So, verbal responses to questions requiring probabilities will be unreliable. Questionnaires can overcome this problem by eliciting probability estimates with pre-formatted response scales. Two examples of this are the bar on which the expert marks a point to indicate degree of uncertainty, or a five point verbal scale on which the expert checks the description most closely associated with their impression of certainty. (Olson and Rueter, 1987)

The questionnaire is useful for illuminating the objects, relations, and inference rules of a domain. This method is similar to interviews in that it relies on the expert's introspection and articulation. Deep causal knowledge of the procedural or semantic kind will not usually be extracted through this method. Therefore, this technique is best for uncovering declarative knowledge early in the acquisition process. (Olson and Rueter, 1987)

3. On-site Observation

On-site observation involves the knowledge engineer observing the expert solving real problems on the job instead of invented but reasonable problems in a laboratory setting. The knowledge engineer does not interfere but acts as a silent observer. This approach lends insight into the complexity of the problem. This method also gives the knowledge engineer some idea of the interface required for the finished system to operate in the field. This technique is not appropriate to those domains where privacy or time is a limitation. It is appropriate however for discovering an expert's judgment, diagnosis, or design decision, if the task is normally conducted in a relaxed atmosphere. This technique also may uncover the objects, relations, and inference rules of a domain. (Olson and Rueter, 1987)

This method has several disadvantages, which include the observer bias of the knowledge engineer and time pressure. The knowledge engineer may not understand the significance of a particular action or the underlying decisions that led to it. Discussions with the expert afterward will rely upon the expert's introspection and articulation of the process. The expert may not be able to verbalize everything. Details regarding all of the knowledge or mental processes will not be available. (Olson and Rueter, 1987)

4. Interruption Analysis

On-site observation becomes interruption analysis at the point where the knowledge engineer can no longer understand the expert's thought processes and interrupts. This method will capture the same types of knowledge and information as will on-site observation. It has the added advantage of instantly capturing the core of the expert's concentration and the types of decisions made for the accompanying procedures. (Olson and Rueter, 1987)

Interruption Analysis has the same disadvantages as on-site observation. While this technique can be very illuminating about the procedure observed, there is little chance that this process can be resumed in a way that will leave it unaffected by the interruption. This technique may be most valuable when the expert system is coded and in the prototype stage; then the expert's performance is compared to that of the system's in an effort to find discrepancies. (Olson and Rueter, 1987)

5. **Protocol Analysis**

Protocol analysis has been widely used for many years as a technique for knowledge acquisition. This method is still widely regarded as a solution to the problem of experts providing unreliable answers when recalling information and tasks about the domain in question. The expert is observed actually solving problems and must concurrently verbalize the decisions made during the task. The dialogue is recorded and later analyzed for the expert's problem solving strategies. This may be useful for eliciting some procedures that experts use, but may be unable to articulate. The intent (mistakenly) has been to use this technique as a tool to extract implicit or procedural knowledge. (Gammack and Young, 1985)

This method has several drawbacks. The knowledge engineer must be sufficiently familiar with the domain to understand the expert's task and commentary. Running commentaries may be difficult for the expert and will affect the task being performed. It is useful only for those tasks in which articulation is a natural part of thinking. It may prove counterproductive in those instances where unique language is used or where the aloud explaining may be distorted or even wrong. The explanations may arrogate attention and energy from perceptual-motor tasks and prove distracting to the expert. (Berry and Broadbent, 1984)

Protocols are often incomplete and cannot establish the boundaries of an expert's knowledge (Berry, 1987). This is because protocol analysis relies on the expert's ability to introspect and articulate. The expert simply does not have access to all their knowledge and mental processes. The deficiency of this technique is noted by Burton and Shadbolt (1988) in their empirical study of knowledge elicitation techniques. Protocol analysis yielded significantly worse results when compared to other methods. It retrieved less information and took a longer amount of time. Laboratory studies show that concurrent verbalization can affect the way in which an expert will perform a task (Berry and Broadbent, 1984). It may force an expert to choose a different line of reasoning than would otherwise be the case.

In light of the latest studies involving protocol analysis, this technique is unlikely to uncover knowledge that resides deeper than surface or declarative knowledge. As a tool for eliciting deep knowledge of the procedural or semantic type, it suffers from substantial weaknesses (Rouse and Morris, 1986).

6. Drawing Closed Curves

This is a direct method that unlike the previous techniques, does not attempt to reveal cognitive processes during the solution of a problem. The previous methods highlight vocabulary used to express objects along with their links to one another and the types of inferences drawn. They do not draw out the form of the relationships, be it networks, tables, lists, or physical space. (Olson and Rueter, 1987)

The drawing of closed curves is a specialized method designed for extracting those relationships that are assumed to be coded in a physical space. This technique requires the expert to show which of a collection of physical objects belong together. A line, in the form of a closed curve, is drawn around those objects that are in some way associated. (Reitman, 1976)

The advantages of this method are that it is graphical and no verbalization is required. It can be applied to any spatial representation, such as a X-ray or CAT scan, a position on a game board, or a typeset formula. Knowledge obtained in this manner is most often represented as networks or as a physical space of some sort. (Olson and Rueter, 1987)

The disadvantages of this technique are much like those for the previous techniques. Though no verbalization is required, this method still relies on the expert's introspection and articulation. Access to deeper knowledge and mental processes may be limited. (Olson and Rueter, 1987)

7. **Inferential Flow Analysis**

This is an adaptation of the interview. With this technique, the expert answers particular questions regarding causal relations. A causal network is then built incorporating all the objects within the domain of interest. The expert initially provides a list of some key objects within the domain. The knowledge engineer questions the expert about relations between two of the objects. The answers will reveal linkages between objects and the directions of those linkages. (Olson and Rueter, 1987)

The expert's responses over a set of questions should reveal consistencies in associations between intervening objects. Each time an object is mentioned by the expert in an answer, it is linked with previous objects and the association is labelled either positive or negative. The linked objects are joined in a network with weights assigned to each link. The weighting on a link is raised in strength with each subsequent mention of that link. (Olson and Rueter, 1987)

The resulting network, though appearing somewhat contrived, is balanced and agreeable with other sets of behaviors. Knowledge extracted by this method is most often represented as flows or networks. This technique is simple to employ and most useful as a tool for displaying to the expert facets of the knowledge they have so far revealed. The resulting display can be used constructively as a springboard for further interviews. (Olson and Rueter, 1987)

The disadvantage of this technique is that it is good only for that which can be expressed as relationships between objects. This method too, relies on the introspection and articulation of the expert who may not have access to all details of knowledge or mental processes. (Olson and Rueter, 1987)

8. **Multidimensional Scaling**

Multidimensional scaling refers to a group of techniques for deriving structure from a matrix of data. These data are usually measures of relatedness among a set of objects whose underlying dimensions of classification are not well known, but are assumed to vary along a translatable number of dimensions. Therefore, this procedure should only be used on data that are assumed to have come from stored models of multi-dimensional space. The knowledge obtained from this method, which is primarily declarative, can be represented in lists, tables, or physical space. (Null, 1980)

This technique involves having the expert compare each object as it is paired successively with the others. Then he or she gives an estimate of the objects similarity by answering the set of questions: "How similar are A and B?". The objects should be typical of the larger domain from which they are drawn. So all objects are judged on a similar basis they should comprise a relatively uniform set without including plainly unusual items. Examples are differentiating flavors of cola drinks or comparing farm animals. (Gammack, 1987)

The similarity estimates provided by the expert are assumed to be balanced and comparable, and able assume a continuous value. From the expert's solution set an explanation can be offered about the nature of the dimensions that distinguish the objects. This is done by arranging the similarity judgments in a half-matrix. This is then input to an analysis program that scans for the best position of these objects in a space of user specified dimension. A "stress", measuring deviation from a perfect fit, is assigned to each dimensional solution. (Olson and Rueter, 1987)

The knowledge engineer selects the lowest "stress" measurements with the fewest dimensions and plots them. The plot is examined for the best placement of the axes. A suitable labelling scheme must be chosen for the axes. For example, size versus ferocity when comparing jungle animals.

This procedure produces significant clusters of objects, relations, and outliers on the plot. (Olson and Rueter, 1987)

A variation on multidimensional scaling is the card sort. This provides a qualitative multidimensional mapping of the elements in the domain. This is done through repeated sorts of a deck of cards, each of which is marked with a domain element. With each sort, the expert labels the overall scale and the individual piles. Rules are then extracted through classification matches. (Burton, Shadbolt, Hedgecock, and Rugg, 1988)

Advantages of this technique include its straightforward manner and the production of quantitative information not present with many other methods. This method may provide a complementary view to a hierarchy by using cross-sectional data on similarities to uncover important global features. It is best when used on sets of low dimensionality and may cause a knowledge intensive activity to deliver large amounts of information. (Gammack, 1987)

Disadvantages of this technique include the tedium of collecting the data. Pairing comparisons of even a small set of objects can very quickly run into the hundreds or thousands and interpretation of the results is not very straightforward. It is difficult for the knowledge engineer to find the dimension with the best "stress" value and then determine placement and names for the axes (Olson and Rueter, 1987).

This method is inappropriate when used on objects that vary in too many dimensions or when too few dimensions are common to all (Gammack, 1987). Another problem with inferring the underlying structure in this manner is that assumptions, which may not be correct, must be made about the underlying representation. (Olson and Rueter, 1987).

9. Hierarchical Clustering

Hierarchical clustering is similar to multidimensional scaling in that it begins with a half matrix of similarity judgments. The assumptions that underlie this technique are the opposite of those for multidimensional scaling. Multidimensional scaling presupposes symmetric distances and ranked properties, whereas this technique just assumes that an object belongs to a cluster or not. Objects cannot simultaneously satisfy assumptions for both multidimensional scaling and hierarchical clustering. This technique is too often misused by some when used to show "clusters" of points on a multidimensional scaling system. (Olson and Rueter, 1987)

This technique uses an uncomplicated algorithm that begins with the half matrix and ends with a hierarchical organization of the objects. Pairs of objects that are neighbors in the matrix are joined to a single cluster, and a new matrix drawn with that cluster representing a new object. This process is iterative, and with each new matrix the distances between unclustered objects are copied from the original matrix. Then, a joining algorithm is selected. The distances between objects and clusters are figured as either the maximum, minimum, or average distance of all cluster objects (all objects within that new cluster) to the object. This results in a hierarchical tree diagram with all the objects listed at the bottom as terminal nodes. The degree of similarity is shown by how far up the tree one must go until two objects become members of the same overarching category. (Olson and Rueter, 1987)

The advantage of this procedure is that it can be accomplished with just a pencil and paper. Additionally, this technique may help the expert identify a structure that they recognize as a natural and effective way of describing some underlying patterns. Knowledge obtained through this technique is primarily represented as relations in hierarchies, a form most people can easily relate to. (Gammack, 1987)

There are several disadvantages of this method. The half matrix which this technique begins with is tedious to develop. If there is no firm theoretical justification for selecting a certain joining algorithm, whether maximum, minimum, or average, one must make an arbitrary choice. The different algorithms will produce substantially different hierarchies, making this technique a somewhat subjective analysis. As with multidimensional scaling, this is a technique based on underlying assumptions and can be misused to the extent that the assumptions are not supported. (Olson and Rueter, 1987)

10. General Weighted Networks

General weighted networks are similar to the previous two techniques in that an expert gives balanced pair-wise distance judgments for all objects. It is assumed that the network derives from the expert negotiating a mental network of relations. The network has a primary path between each pair of objects and possibly a secondarily coded path as well. (Olson and Rueter, 1987)

The first step is to create a *minimal connected network* from a distance matrix. It is formed by connecting the most closely linked items. Then additional links are added and the resulting structure is called a *minimal elaborated network.* This second step adds a link only if it is shorter than the links currently in the network between two nodes. The network is then examined for *dominating concepts* and *members of cycles.* Dominating concepts are those that have many connections to several other nodes. Members of cycles are those that linked into circles. (Olson and Rueter, 1987)

A general weighted network created by an expert is very different from that created by a novice. An expert's network is simpler, connecting larger integrated conceptual structures. They can more easily identify link relations with phrases such as "is-a", "affects", "desirable", etc. This technique can reveal the notable features of expertise. Knowledge obtained from this method is primarily represented as a network but it also may provide lists, tables, hierarchies, or physical models. (Olson and Rueter, 1987)

The primary disadvantage of this technique is that it assumes an underlying form. Like the previous techniques, it relies loosely on the ability of the expert to make a single-valued similarity judgment from whatever form it is stored in. (Olson and Rueter, 1987)

11. Ordered Trees From Recall

Ordered trees from recall derive from research by Reitman and Rueter into how memory structure differs between experts and novices (Reitman and Rueter, 1980). This technique does not begin with a distance matrix but with recall trials. It starts by assuming whether objects belong to a cluster or not, then builds upon a model of memory organization that states an expert will remember all data from a particular cluster before recalling data from another. The basic assumption is that people recall from learned organization. Knowledge obtained via this technique is represented as lists or hierarchies. (Olson and Rueter, 1987)

Patterns found over the set of recalled data are assumed to reflect memory organization. The experts recall object names ten to twenty times and are sometimes told to begin the recall with different names. Pauses during the recall process suggest a transition from one chunk to another. All objects recalled together are identified as chunks and are scrutinized for regularities. The chunks are written into a lattice, then redrawn into an ordered tree structure. The objects are listed at the bottom of the tree as terminal nodes. Horizontal arrows (to represent unidirectional or bidirectional relationships) are drawn over chunk components that were recalled consistently in a certain order. Computer analysis can then be done on the tree structure, scanning for indices of organization or outliers. When these are removed the tree will exhibit a good deal more structure. (Olson and Rueter, 1987)

This technique can be used to prove that experts will show much more organization than novices in a particular domain. Experts within the same domain will exhibit a high degree of similarity with this method. Close analysis of the ordered tree can uncover features of what the expert perceives within their domain of expertise. (McKeithen, Reitman, Rueter, and Hirtle, 1981)

A disadvantage of this technique may be the explicit assumption upon which the entire methodology is based.

Ordered trees assume that domain objects are stored in nested clusters and that *all* objects of this cluster are recalled before shifting to another cluster. The method relies heavily upon the expert's ability at introspection and recall. Additionally, the algorithm employed by the computer analysis may be restricted in what it sees as outliers or indices of organization. The analysis can be done by hand but it is tedious and open to perceptual error by the knowledge engineer. (Olson and Rueter, 1987)

12. Repertory Grid Analysis

The repertory grid technique has its origins in Personal Construct Theory developed by George Kelly in 1955. This theory states that each person functions as a "scientist" who classifies and organizes their own world. Based on these classifications, the individual can construct personal theories of how the particular domain functions. They can then predict and act in that domain based on these theories. When the expert's classifications have been identified and their constructs analyzed, a repertory grid can be developed to represent the expert's understanding of a specific object. (Hart, 1986)

To develop a grid, the knowledge engineer first elicits from the expert a set of constructs that are bipolar in nature. Next, the expert provides a set of examples called elements. The knowledge engineer then requests that the expert rate each element along a linear scale developed to represent each construct. An example construct is "opaque-transparent". The resultant linear scale has "opaque" at one end and "transparent" at the opposite. The element (a type of plastic) is rated on this scale by placing an "X" at the appropriate point along the line. This represents the degree of opaqueness or transparency it possesses. (Hart, 1986)

In the more traditional method of the grid elicitation technique, and the one used by George Kelly, three elements are presented to the expert who picks the odd one. The expert names a dimension such that the odd one is at one pole and the other two form the opposite pole. The remaining objects are rated along this dimension and this is repeated until all objects are distinct in multidimensional space. (Boose, 1986)

When each element is rated according to each construct, the results may be analyzed with a variety of techniques including factor analysis and cluster analysis (of the objects or of the dimensions). The purpose of the analysis is to measure similarities and distances among objects and to represent these graphically as a grid. The elements of the domain are used to define the scope of the problem domain for the grid. The constructs are used to help the expert make useful distinctions among the elements. (Hart, 1986)

The finished grid becomes a cross-referencing system between vital constructs and domain elements. This can be used to find patterns or relations during initial knowledge acquisition efforts (Hart, 1986). It also can be used for extracting the objects, and inference rules of the expert's domain (Olson and Rueter, 1987). This technique is better for analysis problems (debugging, diagnosis, interpretation, and classification) than for synthesis problems (Shaw and Gaines, 1987). It is particularly applicable to classification problems, where features of a new object are observed and the object sorted into one of the known categories (Olson and Rueter, 1987).

This technique is one of the most complete and widely used. Advantages include its free form recall and rating sessions, and the production of a similarity matrix much less tedious to produce than the direct similarity rating of pairs. It also can be used to combine the expertise of two experts within the same domain. (Olson and Rueter, 1987)

This technique has several disadvantages. It is difficult to apply to deep procedural or semantic knowledge (Shaw and Gaines, 1987). Repertory grid analysis elicits traits and builds relationships, but does not find out much about how or when this data is used in the problem-solving process. The constructs used are strictly bipolar; it may be more appropriate sometimes to describe a single trait that could take on any number of discrete values. This technique assumes that the set of elements provided by the expert sufficiently represents the domain. It is more difficult to verify that a representative set of constructs has been extracted. Additionally, interrelationships between constructs or between elements cannot be easily depicted within the grid. (Boose, 1986)

Many knowledge acquisition tools have been based on Personal Construct Theory (see Table 3). The tools based on repertory grids can help a knowledge engineer determine the expert's conceptualization of the domain. This is an important precursor to follow-on efforts in organizing and developing a knowledge base. Most of these tools interact directly with the expert to stimulate them to refine, expand, analyze, and test problem-solving knowledge. See Table 3 for a list of some of these tools. (Boose, 1986)

13. Decision Analysis

Decision Analysis can be useful for capturing an expert's inferences or decision rules within their domain. Knowledge obtained via this method can be represented as an inference network or decision tree, and a "knowledge dictionary" of key concepts. It is a technique that has been widely used in many areas of management and business. This technique is fairly simple and straightforward. The knowledge engineer asks the expert to list all possible decisions when confronted with a particular problem. For each of those decisions all possible consequences are listed. The expert must assess the worth of each consequence and its probability of occurrence. The expected worth of each consequence is calculated by multiplying worth by probability. The expected worth of the decision is a total of the expected worths of its consequences. The expert then selects the decision that maximizes the expected worth. (Hart, 1986)

TABLE 3. KNOWLEDGE ACQUISITION TOOLS BASED ON PERSONAL CONSTRUCT THEORY

TOOL	SOURCE
ETS	Boose (1986)
PLANET	Shaw (1982)
AQUINAS	Kitto and Boose (1989)
FMS Aid	Garg-Janardan and Salvendy (1987)
KITTEN	Shaw and Gaines (1987a)
KRITON	Diederich, Ruhmann, and May (1987)
KSSO	Gaines (1987)
PEGASUS	Shaw and Gaines (1987b)

This may be a very quick way of capturing an expert's heuristics in some circumstances, but it has a serious drawback. It relies on the expert's estimates of worth and probability. Describing their conclusion in terms of probability theory is not an intuitive or natural way of thinking (von Winterfeldt, 1988). Various methods have been developed for eliciting probabilities from people but none has proven universally acceptable (Hart 1986).

14. Machine Induction

There has been a good deal of controversy over whether machine induction will prove to be a useful source of knowledge or not. Some believe that the problem of extracting deeply compiled procedural and semantic knowledge can be dodged with this technology. This procedure has the expert provide a set (training set) of examples of different types of decisions from the domain. Also provided are the relevant attributes that affect the decision. All this is fed as data into a software inductive algorithm that produces the simplest set of rules that can generate the examples. This allows for an explanation of the decision process and provides predictions of decisions for examples not in the training set. (Berry, 1987)

Automatic induction can produce rule bases very quickly. It may draw out deeply embedded or compiled knowledge because the expert need not have a clearly formulated explicit rule that is used when carrying out a task. Indeed, the expert need not even be present, as the training set may be drawn from documentation. (Berry, 1987)

There are several disadvantages to machine induction. Some domains do not have a base of documentation or examples that can be easily drawn upon. Additionally, what comes out of the induction algorithm is only as good as what goes in. The training set must represent the domain, and it must contain the unusual or rare cases as well. A random sample will not provide this. Induction algorithms (i.e., ID3) cannot cope with uncertain or noisy domains. Further, the rules that an expert uses will not be like those produced by the algorithm. Machine induction produces rules that tend to be more complex and difficult to understand and thus less desirable for coding into expert systems. (Berry, 1987)

F. SUMMARY

The acquisition of knowledge from experts will remain a "bottleneck" for some time to come. Yet, many of the steps involved in the knowledge acquisition phase have benefitted greatly from research over the last two decades. There are frameworks for classifying knowledge and tasks, matching problem-solving techniques to tasks, and representing knowledge. But, the greatest problem remains the extraction of knowledge from an expert. A plethora of techniques have been focused upon this problem, but as this survey has shown, none are wholly satisfactory and all have serious drawbacks. See Tables 4 and 5 for a summarization of all the techniques.

TABLE 4. DIRECT KNOWLEDGE ELICITATION TECHNIQUES

TECHNIQUE	KNOWLEDGE TYPES ACCESSED	DRAWBACKS
Interviews	Declarative	1. Subject to articulation. 2. No deep knowledge uncovered.
Questionnaires	Declarative	1. Subject to articulation. 2. No deep knowledge uncovered.
On-site Observation	Declarative	1. Observer bias, time pressure. 2. Privacy limitations. 3. Expert may not recall all the reasoning
Interruption Analysis	Declarative	1. Difficult to resume process. 2. Subject to articulation.
Protocol Analysis	Declarative	1. Subject to articulation. 2. Affects task.
Drawing Closed Curves	Declarative	1. Requires expert introspection.
Inferential Flow Analysis	Declarative	1. Requires expert introspection. 2. Only for that which can be expressed as relationships.

TABLE 5. INDIRECT KNOWLEDGE ELICITATION TECHNIQUES

TECHNIQUE	KNOWLEDGE TYPES ACCESSED	DRAWBACKS
Multidimensional Scaling	Declarative	1. Tedious. 2. Subject to interpretation. 3. Subject to assumptions.
Hierarchical Clustering	Declarative	1. Tedious. 2. Algorithm choice arbitrary. 3. Subject to assumptions.
General Weighted Networks	Declarative, Some Procedural	1. Assumes an underlying form.
Ordered Trees from Recall	Declarative, Some Procedural	1. Introspection and recall. 2. Based upon assumptions.
Repertory Grid Analysis (Personal Construct Theory)	Declarative	1. No deep knowledge. 2. Use of limited constructs. 3. Difficult to verify.
Decision Analysis	Declarative, Some Procedural	1. Relies on estimates of probability and worth.
Machine Induction	Procedural	1. Training set rarely representative. 2. Only as good as induction algorithm. 3. Complex rules difficult to code.

The direct techniques have as the greatest problem, reliance upon an expert's ability to articulate what he or she really knows. Five of the seven methods require the expert to verbalize, either in interviews, or at some point during or after a problem-solving task. The remaining two methods still require articulation. Questionnaires are, for the most part verbalization on paper, and the drawing of closed curves forces the expert to explicitly indicate relationships where, in some cases, none may exist. All of the direct methods require the expert's introspection and mental search for the correct data.

Research has indicated many problems in extracting the knowledge of individuals. Knowledge may not be available to awareness, and even if it is, it may not be expressible in language. If it is expressible in a language, it may not be understandable, for example, to a novice. Further, it is entirely possible that expressed knowledge may be irrelevant, incomplete, or incorrect. (Gaines, 1987)

The indirect methods of knowledge elicitation attempt to circumvent reliance upon an expert's introspection and articulation. Though the indirect techniques presented here have demonstrated psychological validity in controlled settings, they all have serious drawbacks to implementation. These methods cause the knowledge engineer to become the weakest link in the extraction process because it is he who must infer an underlying structure among the data obtained from the expert. This means that the knowledge engineer must be highly trained in psychological modelling techniques. Further, if the underlying structure is not inferred "*a priori*" so that the "correct" technique can be selected, the data may not support the process.

Many elicitation methods have proven useful for tapping into the declarative knowledge of an expert. Yet, there is no satisfactory way of extracting procedural or semantic knowledge. The indirect techniques present a "hit-or-miss" proposition based on the validity of the underlying assumptions. Empirical research into comparisons of elicitation methods is still in its infancy, and the need clearly exists for a good deal more. Until an empirically derived data base of domains, tasks, knowledge extraction techniques, and their interactions exist, the claims and counter-claims for each technique will remain just that.

This survey indicates a need for an automated technique based on a sound psychological model of expert thought processes, that can extract deeply embedded procedural and semantic knowledge. This technique should be natural to the expert so that it does not force him or her into different modes of reasoning. It should eliminate the knowledge engineer, that translation through another person, so the expert is enabled to interact directly with the process to create a knowledge base. A new technique should allow for a graphic interface and graphic manipulation, as this is how much of long term, deeply embedded memory may be stored. Many automated knowledge acquisition tools have been developed, but they are based on one or a combination of the techniques discussed above, with all of the attendant shortcomings.

III. COGNITIVE FEEDBACK

A. INTRODUCTION

Cognitive feedback is regarded as an effective tool for the capture and representation of a person's mental model (Doherty and Balzer, 1988). This chapter will provide the justification for that assertion and explain how the mental model can be captured. The theoretical foundations of cognitive feedback, Brunswik's probabilistic functionalism and its representation through the lens model are introduced and summarized first (Hursch, Hammond, and Hursch, 1964; Tucker, 1964). Cognitive tasks and cognitive systems are defined and their relation to the lens model illustrated.

Cognitive feedback is explained in detail and shown to be superior to the other form of decision feedback, outcome feedback. The different types of cognitive feedback will be described as well as the various formats in which it can be presented. Cognitive feedback has been put to many different uses, and there is a moderate corpus of research on the effectiveness of cognitive feedback. The results of these studies will be summarized, and some issues and applications for the future will outlined.

B. PROBABILISTIC FUNCTIONALISM

Probabilistic functionalism is a description of the process by which an organism adapts to an uncertain, probabilistic environment, and how the tradeoffs it must make to survive play a central role. Brunswik (1943) believed that to understand the underlying forces guiding an organism's behavior one must focus on the organism's achievement in adapting to the environment. This involves a thorough study and characterization of the environment, the organism in the environment, and the means by which the adaptation occurs. These ideas are salient to practitioners of knowledge acquisition because their goals are to capture an expert's mental model. The mental model has developed as a means of adapting to, and imposing some order upon the environment.

A detailed analysis of the environment is necessary to any explanation of an individual's judgmental processes. Tolman and Brunswik (1935) have stressed that an organism, in its normal interaction with the environment, must deal with many, interdependent, diverse relations among variables (cues), which may be partly relevant or irrelevant to its goals. The cues are limited in their dependability and may be organized in a variety of ways. There is considerable redundancy and interchangeability among the cues. In short, the environment is probabilistic.

Given the many complex situations that can arise from a probabilistic environment, the organism must adjust by bringing to bear a variety of cognitive processes, such as perception, thinking, and learning if it is to survive. Crucial to this process of adjustment is feedback (Hogarth, 1981). If feedback is received and acted upon continuously, the organism has access to a greater number of cues and responses. The cues become more intersubstitutable. The effect that a redundant environment has upon behavior was stressed by Brunswik through the principle of vicarious mediation:

> Since there is no perceptual cue which would be available under all circumstances or is completely trustworthy... the perceptual system of higher organisms must for most types of perceptual attainment develop what the present writer has suggested calling an "or-collective" or an "or-assemblage"... of mutually interchangeable cues vicariously mediating distance or other situational circumstances to the organism... Since cues form a hierarchy just as do means, we may also speak of a "cue-family-hierarchy"... (1955, p. 677)

The importance of this principle is apparent:

> We may add that vicariousness of psychological cues and means may be viewed as a special case of receiving or sending messages through redundant, repetitive channels, thus reducing the probability of errors, that is, the set of possible causes, or effects, that could result in, or be produced by, the type of event in question. Vicarious functioning is thus indeed of the essence of behavior. (Brunswik, 1955, p. 750)

Substitutions between interchangeable cues lead to equivalent results. The organism orders the cues into a system that enables it to make judgments about some object or the future.

The vicarious functioning occurs within a certain environmental context. As a result, Tolman and Brunswik (1935) argued that more emphasis should be placed upon studying the environment and the organism within that environment, than the organism in isolation. The first step toward this understanding must be to study the texture of the relationships among cues in the tasks that require judgment.

Brunswik (1955) stated that the organism's (cognitive) system and the environmental system should be described symmetrically. This is represented in the lens model of behavior (Figure 1). Brunswik described the lens model with a principle of parallel concepts. Each concept on one side is paralleled by an equivalent concept on the opposite side. The cues on the task or ecological (environmental) side vary in *ecological validity* and the cues on the organism's (cognitive) side vary in *cue utilization.* The relations between cues and distal variables (the criterion or object of interest) on the ecological side may assume various forms, just as the relations between cues and judgment on the cognitive side may assume various forms.

C. THE LENS MODEL

Brunswik's lens model is a general construct that graphically embodies the principle of parallel concepts and stresses many important aspects of the decision making process

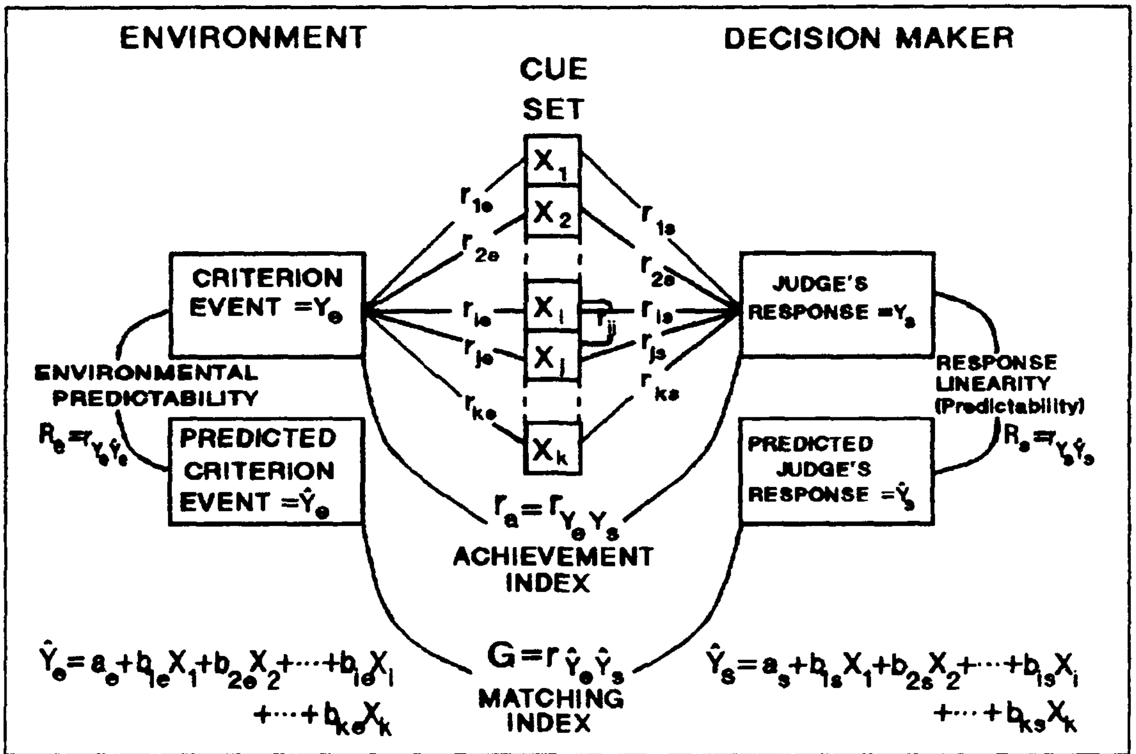


Figure 1. The lens model (Libby, 1981).

under uncertainty. The model can be viewed as an individual judging an event or object (criterion), which cannot be directly perceived, through a lens of cues. The relationship of the cues to the criterion event and to the judge are uncertain. The individual's interaction with the environment may be described by several relationships such as those among cues, those between the cues and the criterion event, those between the cues and the individual's judgment, and those between the criterion event and the individual's judgment. (Brehmer, 1979)

1. The Lens Model Equation

The version of the lens model presented here is the regression formulation and is the one most widely used. The model was quantified with regression analysis by Hursch, Hammond, and Hursch (1964), Hammond, Hursch, and Todd (1964), and Tucker (1964). The material presented here is drawn from Libby (1981) who has further refined the model.

There are three elements to the model. The first is the *task environment* defined by the cue set (X_1, X_2, ..., X_k). The second element is the *criterion event*, also called the focal variable, on the left side of the model and denoted by Y_e. The third element is the *judge's estimate* of the event and is denoted by Y_s on the right side of the model. The relationships among these elements (see Figure 1) are summarized in the lens model equation.

The task environment is defined by the cue set (X_1, X_2, ..., X_k) and the matrix of intercorrelations between the cues, r_{ij}. The relationships of the cues to the criterion and of the cues to the judgment are measured by both univariate and multivariate correlations. The *ecological validity* of a cue is measured by the univariate relationship between each cue (X_i) and the criterion event (Y_e) and is denoted by r_{ie} on the left side of the model. This measures the relevance of the i^{th} cue to predicting the criterion event and is independent of the other cues. The multivariate relationship between all the cues and the criterion event is determined by the following linear regression model:

$$\hat{Y}_e = a_e + b_{1e}X_1 + b_{2e}X_2 + \ldots + b_{ke}X_k, \quad Y_e = \hat{Y}_e + u_e.$$

The cues-criterion multivariate relationship is assessed by the correlation of the criterion event (Y_e) and the prediction of the criterion event ($\hat{Y}_e$) from the above model. The measure is known as *environmental predictability* ($R_e = r_{Y_e \hat{Y}_e}$) and shows the relevance of the cue set to predicting the event.

The right side of the model, the cognitive system, is described in terms similar to that used for the left side, as required by the principle of parallel concepts. The reliance of the judge upon individual cues is measured by the univariate relationship between the cue (X_i) and the response or judgment (Y_s). This is called the *utilization coefficient* (r_{is}) and may take a positive or negative value between zero and one. An ignored cue is given a zero weight. The multivariate relationship between all the cues and the response is defined by the linear regression model:

$$\hat{Y}_s = a_s + b_{1s}X_1 + b_{2s}X_2 + \ldots + b_{ks}X_k, \quad Y_s = \hat{Y}_s + u_s.$$

The cue-response multivariate relationship is assessed by the correlation of the actual judgment (Y_s) with the model's *prediction* of the judgment ($\hat{Y}_s$). This measure is known as the *response linearity* ($R_s = r_{Y_s \hat{Y}_s}$) and may indicate predictability or consistency of judgment.

When the two regression models are compared, the similarity of the decision maker's weightings of cues to the environmental relationships can be assessed. This is done by correlating the predictions of the two equations to form the *matching index* ($G=r_{\hat{Y}_e\hat{Y}_s}$). If each linear model captures all reliable variance in each system, the index can be regarded as an overall measure of the accuracy of cue weighting or utilization. This is because the effects of human inconsistency and environmental unpredictability are eliminated in the regressions.

The *achievement index* ($r_a=r_{Y_eY_s}$) summarizes the judge's performance and shows the correspondence between the judge's response and the environmental event. This measure provides a direct *ex post* indicator of judgment accuracy. Achievement can be explained in terms of the other components of the lens model with the following equation: $r_a=GR_eR_s$. See Table 6 for a summary of lens model components. Achievement depends on three factors: (1) the weighting of cues relative to their weighting in the environment (G is usually less than one because most decision makers fail to use an optimal weighting strategy, which is implicit in the environmental model); (2) the predictability of the environment (R_e is less than one because the environment is not perfectly predictable); (3) the predictability of the individual (R_s is less than one because decision makers are not perfectly consistent). When combined multiplicatively it is apparent that judgmental achievement will not be high, which is consistent with empirical results.

TABLE 6. LENS MODEL STATISITICS

Symbol	Name	Definition
r_{ie}	Ecological validity	$r_{X_iY_e}$
R_e	Environmental predictability	$R_{Y_e\hat{Y}_e}$
r_{is}	Utilization coefficient	$r_{X_iY_s}$
R_s	Response linearity (predictability)	$R_{Y_s\hat{Y}_s}$
G	Matching index	$r_{\hat{Y}_e\hat{Y}_s}$
r_a	Achievement	$r_{Y_eY_s}$

2. Single Systems

Figure 1 illustrates the *double systems paradigm*, so called because it involves analyses of the relations between two systems. The cognitive system (right side of lens model) is compared to the task system (left side of lens model). Standing in contrast to the double system paradigm is the *single system paradigm*. This involves analysis of only the right side of the lens model as depicted in Figure 2. Studies conducted within this framework analyze the relations between a set of cues and a set of judgments with multiple regression or analysis of variance procedures. (Brehmer, 1979)

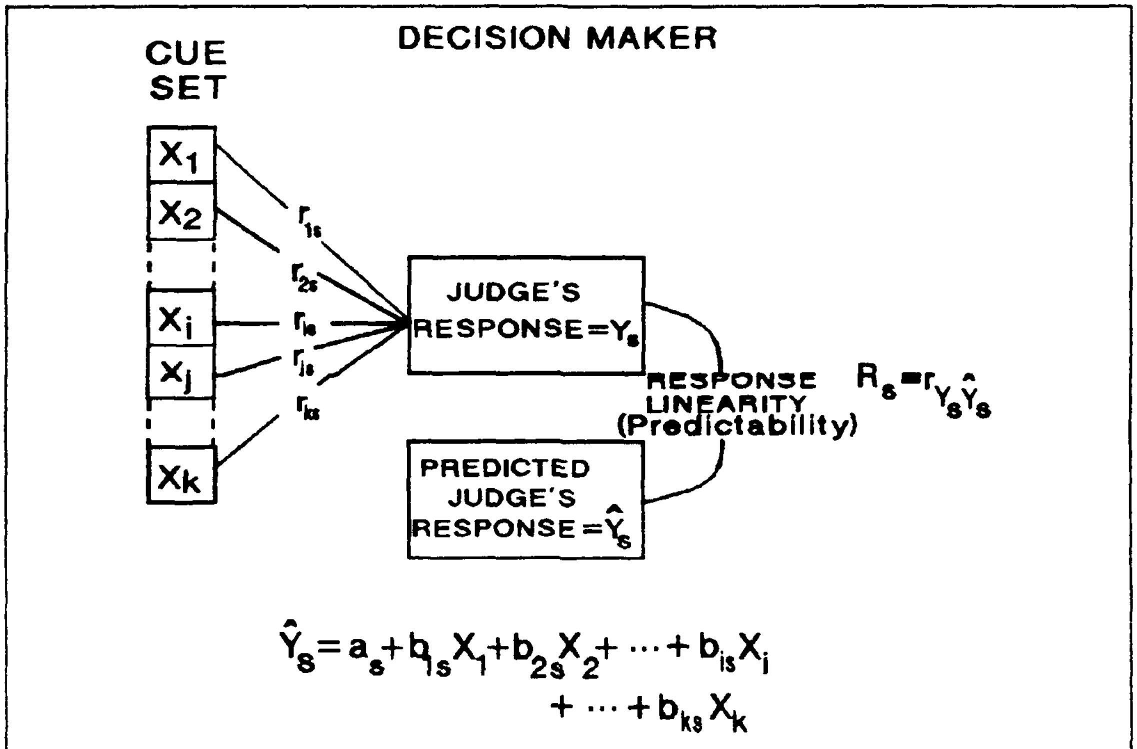


Figure 2. The single system case (Libby, 1981).

D. A FRAMEWORK FOR COGNITIVE SYSTEMS AND COGNITIVE TASKS

Brunswik's probabilistic functionalism provides for a distinction between proximal variables (cues) and a distal variable (the criterion), and is particularly well suited to tasks involving inference. Indeed, the proximal-distal separation is the definition of an inference task. The ability to make inferences is the ability to go beyond the information given (cues) and make a conclusion about what cannot be directly perceived (criterion). Brunswik assumes this proximal-distal variable relation is probabilistic in nature, so inferences cannot be made with complete certainty regarding the distal variable. There are many important tasks containing uncertainty and all inference tasks are probabilistic in nature. (Brunswik, 1955)

The lens model and statistical concepts can be used to describe cognitive tasks and cognitive systems. This makes it possible to handle the problem of uncertainty and to express both the regularities and irregularities of each system. Formal logic proves inadequate as a representation for this. The principle of parallel concepts holds that both systems be described with similar concepts (Brehmer, 1979).

1. Cognitive Tasks

A cognitive task is the process by which an individual selects a focus and then obtains information about that focus. Cognitive tasks do not exist independently of a person, they arise from the person's desire to know something. The focal variable (criterion) is most often not directly perceived by the person who must find a set of cues (proximal variables) upon which to base an inference of the state of the focal variable. The person must conform to the structure of the environment, as it applies to the focal variable, when choosing the set of cues and learning to use them. Cognitive tasks are therefore dependent on both the individual and the environment. The implication of this is that cognitive tasks will differ among individuals because each person will select different foci and cues. (Brunswik, 1955)

A cognitive task, once defined, may be described in terms of its formal characteristics, of which there are seven dimensions divided into two classes. The first class groups together the surface characteristics that relate to the nature of the proximal variables (cues), as opposed to relations between cues and focal variables. Surface characteristics include the *number of cues*, their *metric characteristics* (i.e., nominal or quantitative), and the *intercorrelations among the cues* (i.e., the extent to which cues tend to go together). (Brehmer, 1979)

The second class groups together the four system characteristics. The first is the *relative weights for the cues* (i.e., some may be more important than others). The second is the *functional relations between each cue and the distal variable*. The third refers to the *integration rule* for integrating information from the cues into a single judgment (i.e., additive, averaging, or configural). The fourth system characteristic is the *predictability* of the system. System predictability may be low, as when not all the cues are available, the system is inherently unstable, or the criterion event is far into the future. Alternatively, system predictability may be high, as when all relevant cues are available or there is little time lag. (Brehmer, 1979)

2. Cognitive Systems

A cognitive system is a representation or a model of what a person perceives in the environment. It is the judgment process by which the person copes with their surroundings and it is depicted by the right side of the lens model. A description of a cognitive system details what cues are used, the weights assigned to each cue, the functional relations between the cues and the judgment, what principle is used to combine the cues, and the system's predictability. Also included in the definition is an account of the metric level at which the cues are used. This may not be the same as in the cognitive task because a person may be assigning a quantitative interpretation to nonquantitative cues or vice versa. (Brehmer, 1979)

It is important to note that the judgment process itself does not function according to multiple regression or analysis of variance. There is a good deal of empirical evidence against this. But, considerable research suggests that a simple linear model will often adequately explain the judgments made by an individual. Hoffman (1960) referred to the use of linear models as *paramorphic representations* of judges. This means that the cognitive processes of individuals do not actually compute weighted averages of cues or variables, but rather these processes can be simulated or described through the use of such weightings.

E. FEEDBACK

An understanding of cognitive feedback (CFB) first requires a definition of the term "feedback". As defined by Doherty and Balzer:

> While the term feedback (FB) has been used in a variety of ways in different disciplines, by definition it involves an environment that returns some measure of the output of a system back to the system that produced that output. The FB then allows the system to compare its present state with an ideal state, to adjust itself in light of that comparison, and bring itself closer to that ideal state. (1988, p. 163)

For the purposes of this thesis that system is a person.

Hogarth (1981) notes the importance of feedback to judgmental accuracy. Judgment is essentially a continuous process that is predominately exerted to facilitate action. The actions normally produce feedback that is immediately available. This gives rise to a series of incremental judgment-action-feedback loops that monitor progress during activity. Feedback is therefore central to behavior. It enhances an individual's ability to adapt because it reduces any particular action's implied commitment.

1. Two Types of Decision Feedback: CFB and OFB

In decision and judgment literature, two types of feedback have been identified: cognitive feedback and outcome feedback (Hammond, Stewart, Brehmer, and Steinman, 1986). Cognitive feedback returns some measure of a person's cognitive output to help that person come to grips with the environment. In particular, an individual receives information describing the relationships defined in the lens model.

Outcome feedback, had in the past, always been assumed to improve the accuracy of an individual's judgments. Outcome Feedback (OFB) simply describes the accuracy or correctness of a judgment. It is the presentation of Y_e to an individual immediately after that person produces Y_s. Cognitive feedback, on the other hand, returns information on the how and why that supports the accuracy of a judgment. Given the definition of feedback presented earlier, OFB is really not a form of feedback at all, because it does not return information that a system can use in adjusting its response to the environment. (Doherty and Balzer, 1988)

Outcome feedback's effects have been studied extensively, especially with *multiple cue probability learning* (MCPL) experiments. In MCPL studies subjects are given sets of cues and asked to make overall judgments. After making the judgment they are presented with its accuracy but not with information regarding relations between cues and criterion. (Hammond et al, 1975)

2. CFB Versus OFB

Many studies have directly contrasted OFB with CFB. For example, a study of security analysts participating in a security analysis decision simulation by Jacoby, Mazursky, Troutman, and Kuss (1984), led to two important conclusions. First, in an environment that permits decision makers to be selective in the information they choose, not all will access feedback information if it possesses only outcome value, which fails to possess predictive or explanatory aspects. Second, better performing decision makers are less likely to access OFB than are poorer decision makers. This led to the conclusion that OFB may be especially dysfunctional in a complex, dynamic environment.

Hammond and Summers (1972) have proposed a theory stating that performance in cognitive tasks depends upon acquisition of knowledge and cognitive control over knowledge already acquired. Their studies suggest that OFB is an impediment to the learning of complex inference tasks, especially when the relations are complex and under conditions of uncertainty. When OFB was removed, an increase in response consistency ($\hat{}Y_s$) was typical. When compared, under the same conditions, to individuals receiving CFB, the CFB group performed most accurately. Hammond and Summers' conclusion:

> Furthermore, the evidence which suggests that traditional, response oriented outcome feedback is an impediment to cognitive control (and thus to performance) also points to the facilitating effect of *cognitive* material as feedback. This shift in conception of the notion of feedback carries considerable practical as well as theoretical significance, for it is now evident that computer technology can be used to produce such facilitating feedback. (1972, p. 66)

There are some researchers who disagree with the above conclusion. Klayman (1984) contends that the focus of research has been on the perception of shapes and magnitudes of cue-criterion functions and that use of these may not be how people actually learn. The model building process itself may be how people acquire knowledge and outcome feedback may be an effective tool when applied to this. He argues for a greater research effort directed toward the model building process.

Despite some favorable reports on the effectiveness of OFB, most of the current literature shows that OFB provides little value and may even be detrimental to learning in cognitively complex, uncertain, probabilistic environments. Doherty and Balzer states, "The superiority of relational information, or what has been loosely called CFB, over 'OFB' has been confirmed many times" (1988, p. 176).

F. COGNITIVE FEEDBACK

1. Definition

Cognitive feedback is the feedback that contributes to the exercise of control. It consists of cognitive material rather than response-oriented material (OFB). OFB only enables individuals to see that their decision was in error, but not why it was in error. If a person is to discover why they were in error, they must have feedback that allows them to compare the properties of their cognitive system with the properties of the task system that is being dealt with. (Doherty and Balzer, 1988)

2. Types of CFB

Three types of information can be returned in the CFB process: Task information (TI), cognitive information (CI), and functional validity information (FVI). Task information refers to the relationships between cues and criterion. It represents the left side of the lens model (ecological or environmental side). The TI that can be returned is R_e (the multiple correlation indicating overall task uncertainty), r_{ie} (correlations between individual cues and criterion), and r_{ij} (cue intercorrelations). (Balzer, Doherty, and O'Connor, 1989)

Cognitive information refers to the relationships between cues and the person's judgments. This is the right side of the lens model (decision maker side). As implied by the principle of parallel concepts, CI largely mirrors TI except that there is no equivalent to r_{ij}. CI that can be returned is r_{is} (utilization coefficient) and Rs (response linearity or predictability). Also, the conceptual interpretations of consistency (right side) are very different from predictability (left side). (Balzer et al, 1989)

Functional validity information are the relationships between judgments and criterion. This is the achievement index r_a and the matching index G. (Balzer et al, 1989)

TI, CI, and FVI can all be returned to an individual, either alone or in combination with the others. However, for an operation to be labeled cognitive feedback it must include a cognitive component, that is, either CI, FVI, or both. Each CFB measure can be returned to the individual in various formats: verbally, graphically with bar-graphs or cue-criterion function forms for example, or statistically by correlation measures. See Table 7 for a summary of the types of CFB. (Balzer et al, 1989)

Only CI is provided in the single system paradigm; the concepts of TI and FVI are irrelevant. When CI is returned to the decision maker it can be used as a cognitive aid to heighten insight into one's own system of values as it applies to a given environment. Most studies have suggested that individuals may lack a high degree of insight into their policies. Individual's descriptions of their policies are often inaccurate and difficult to verbalize. This is one of the reasons that CFB indexes and procedures were originally developed. If the hypothesis that individuals lack insight into their policies is true, it would justify development of systems to provide them with CI. On the other hand, if experts' insight into their policies is imperfect but not totally absent, as one study relates, then presentation of CI, though redundant, would provide an externalization of their policy. (Balzer et al 1989)

TABLE 7. TYPES OF COGNITIVE FEEDBACK.

TYPE	MEASURE	INTERPRETATION
Matching Index (G)	$r_{\hat{Y}e\hat{Y}s}$	Extent to which task properties correctly identified. Accuracy of cue utilization.
Achievement (r_a)	r_{YeYs}	Correspondence between judge's decision and the environmental event.
Response Linearity (R_s)	$R_{Ys\hat{Y}s}$	Predictability/consistency of judgment. Extent to which judge controls execution of knowledge.
Environmental Predictability (R_e)	$R_{Ye\hat{Y}e}$	Relevance of the cue set to predicting the criterion event. Overall task uncertainty.
Ecological Validity (r_{ie})	r_{XiYe}	Relevance of the i^{th} cue to predicting the criterion event, independent of other cues.

3. Presentation of CFB

Cognitive feedback can be presented in a variety of forms. Brehmer and Svensson (1976) plotted a judge's last block of judgments on the same graph as the true function forms. Function forms (CI) relate the cues to the judgment. Todd and Hammond (1965) did the same thing and included the means of the criterion and judgment values. Schmitt and Levine (1977) presented transformations of beta values from the regression equation. Many researchers have given solely verbal descriptions (Deane, Hammond, and Summers, 1972) or verbal and graphical descriptions of function forms. All the above methods have demonstrated effectiveness, but the most frequently employed method is graphical.

Few studies exist that have directly compared the variety of feedback formats. This may not be very important because human beings appear well adapted to receiving input either verbally or graphically, but people may differ on what format works best. Therefore, if a cognitive aid is developed for the purposes of feeding back lens model indexes, it would be best to build in as much redundancy as possible. (Doherty and Balzer, 1988)

Presentation of a particular CFB index (i.e., CI, TI, FVI) or presentation in some format rather than another, may have the effect of returning slightly different information to individuals. Since CFB may be used to change a person's judgment policy, or mental model, different combinations of CFB may produce different mental models. (Doherty and Balzer, 1988)

The computer is, of course, the perfect tool for analyzing an individual's judgment policy and for instantaneously generating graphical feedback. Hammond et al describes the use of a *cognitive aid*, "Persons exercising their judgment can discover, immediately and in pictorial form (by means of computer graphics), the properties of their own judgmental system, as well as the properties of another person's judgmental system, and *change* those properties, if they desire, with complete control" (1986, p. 67). Not only can representations of cognitive systems be compared, cognitive systems can also be compared with pictorial representations of task systems as well. This enables one to study the degree of match between the two systems, the right and left sides of the lens model.

Time is an important consideration in the presentation of feedback. Wickens (1984) notes that if feedback is delayed, many salient factors that went into the decision making process will have been forgotten. If the judge is preoccupied with something else, there will probably be scant attention paid to the feedback. Fischoff (1977) states that this can be exacerbated by *cognitive conceit.* By this, individuals underestimate the information gained from observing the effects of their decision and will overestimate, in hindsight, the extent of prior knowledge. If the discrepancy between what is known after a decision, and what was thought to be known before the decision is slight, there appears to be nothing wrong with the original decision making process.

4. Applications of Cognitive Feedback

Much of the evidence for the power and usefulness of CFB has come from non-laboratory settings. This suggests that there are a good many practical applications for CFB, many of which are now being realized. Cognitive feedback is also being used in laboratory settings as a basic research tool.

Cognitive feedback has been used for training purposes within the medical fields and may be useful to any practitioner who must make multiple-cue judgments of distal, incompletely discernable objects. As an example, TI + CI has been presented by computer program to improve the diagnostic accuracy of medical students evaluating urinary tract infections. The same program has been used to improve the achievement and tuning of diagnoses in streptococcal infections. Cognitive feedback, as CI, has also been used to increase agreement between specialists in a medical field where there is much disagreement over proper treatment (rheumatoid arthritis). (Balzer et al, 1989)

Cognitive feedback has been used in performance rating training programs. Here, information (i.e., TI) is provided to raters about which cues or behaviors should be regarded during a rating session. The student's patterns of ratings are compared to ratings provided by experts. This has been shown to reinforce and improve multiple measures of rating effectiveness. A technique such as this could conceivably be used to prepare personnel screening boards for their task, or for the training of portfolio managers, inspection and auditing personnel, and even battlefield situation assessment or conflict prediction/resolution tasks. (Balzer et al, 1989)

Further examples for the application of CFB as TI come from Balzer et al:

> CFB as TI may be used to teach (a) selection interviewers to differentially weight dimensions of interviewees' performances; (b) stockbroker trainees to use various indexes of company performance when making sell or buy recommendations; (c) clinical trainees to focus on certain aspects of a client's personal history, test performance, or interview behaviors when making diagnoses; and (d) medical students to examine and integrate particular pieces of current and previous medical history. (1989, p. 430)

More applications for cognitive feedback involve the return of FVI in a manner that informs individuals about the validity of their judgments. Those engaged in personnel selection could be provided with some measure of the relationship between their selection decisions and the subsequent performance of the personnel. This information could lead them to retain, change, or discard their personal selection policies. This can be applied to the organizational level as well. (Balzer et al, 1989)

5. Research on Cognitive Feedback

Many studies have been conducted to determine whether CFB really "works". A problem with comparing all the studies is that several measures have been used to assess CFB. Some have used the common lens model statistics of R_s, r_a, and G as dependent variables. Others have used variants on these statistics or have added self reports from users of CFB.

Three criteria are used by Balzer et al (1989), to integrate all the results from studies on CFB. The first is *reaction criteria* that includes self reports, informal testimonials, or formal scale responses. Reaction criteria is primarily qualitative in nature and most studies in this area have reported favorable reactions to CFB. The second, *behavioral criteria*, appraises change in some specific feature of a person's performance. The changes may include insight into one's own policy, consistency of policy usage, or task learning improvement. When used as dependent variables R_s, r_a, and G belong in this second category. The third, *results criteria*, measures whether CFB resulted in improvement beyond the CFB task and as such has not received much examination. Most studies have dealt with behavioral criteria.

Three behavioral criteria are represented in the lens model: knowledge (G, the matching index), control (R_s, response linearity or predictability), and achievement (r_a). Studies have looked at each of these behavioral criteria as if they were separate and others have explored the interactions between knowledge and control as it affects achievement.

Several studies assessing the impact of CFB on knowledge, have found that providing individuals with TI + CI have led to significant increases in linear matching. In one project r_c ($r_c = Gr_s$), which assesses the extent to which an individual can predict linear variance in the environment, was significantly higher for those receiving CFB than OFB. Still, some other experiments providing feedback as TI and extensive CFB (r_{is}, r_{ie}, R_s, R_e) led to little or no increase in knowledge. Most studies however, show that CFB will lead to increases in knowledge. See Table 8 for a summary of studies on the impact of feedback on knowledge (Balzer et al, 1989).

TABLE 8. STUDIES ON THE IMPACT OF CFB UPON KNOWLEDGE.

Type of Feedback	Results	Study
CFB + OFB	TI+CI+OFB > TI+OFB ≈ OFB > CI+OFB CFB < CFB+OFB ≈ No FB	Schmitt, Coyle, and King (1976) Schmitt, Coyle, and Saari (1977)
CFB, OFB	TI ≈ TI+CI+FVI > OFB	Nystedt and Magnusson (1973)
CFB, No FB	CFB > No FB	Fero (1975)
TI + CI	CFB > OFB No increase CFB > OFB CFB > OFB	Adelman (1981) Clover (1979) Lindell (1976) Hoffman, Earle and Slovic (1981)
TI, TI + CI + FVI	TI+CI+FVI ≈ TI	Galbraith (1984)
CI, TI + CI	TI+CI > TI > CI	Schmitt et al (1976)
TI	TI > No TI	Schmitt et al (1976)
FVI	No studies	
CI	No studies	
TI + FVI	Increased	Newton (1965)
TI + CI + FVI	Increased	Newton (1965)
TI + CI + FVI, TI + FVI	TI+CI+FVI ≈ TI+FVI > OFB	Steinmann (1974)

CFB has been evaluated with respect to its impact on control (R_s). One of the methods employed was to examine the effect of CFB on a person's ability to employ policies in a consistent fashion. Results are mixed, but generally CFB has produced more significant increases in control than not. In another study, CFB has significantly improved control at a Veteran's Administration drug dependency unit. Similar results were reported in a study of learning and training in undergraduate students. OFB has also been shown to decrease R_s when presented with CFB. See Table 9 for a summary of research studies on the impact of CFB upon control (Balzer et al 1989).

When CFB is examined for its effect on achievement several studies show significant improvement in r_a. It has been demonstrated that r_a is highest when TI + CI is received, second highest when only TI is received, and lowest when OFB is presented. One study showed that r_a was significantly lower for OFB individuals than for either CFB or CFB + OFB. See Table 10 for a summary of research studies on the impact of CFB on achievement (Balzer et al, 1989).

Most studies indicate that CFB improves behavioral criteria. It reinforces linear matching of an individual's policy with the linear environment, linear consistency, and achievement. Knowledge (G) and cognitive control (R_s) are both significantly increased by CFB, and in an environment of

TABLE 9. STUDIES ON THE IMPACT OF CFB UPON CONTROL.

Type of Feedback	Results	Study
CFB + OFB	No effects CFB > CFB + OFB	Schmitt et al (1976) Schmitt et al (1977)
CFB, OFB	Decreased TI ≈ TI+CI+FVI ≈ OFB	Balke, Hammond and Meyer (1973) Nystedt and Magnusson (1973)
CFB, No FB	CFB > No FB	Fero (1975)
TI + CI	CFB > OFB No increase CFB ≈ PDF > OFB CFB > OFB	Adelman (1981) Clover (1979) Hoffman et al (1981) Lindell (1976)
TI, TI + CI + FVI	TI+CI+FVI ≈ TI	Galbraith (1984)
CI, TI + CI	No studies	
CI	No studies	
TI	No studies	
FVI	Increased	Newton (1965)
TI + CI + FVI	Increased	Newton (1965)
TI + CI + FVI, TI + FVI	TI+CI+FVI ≈ TI+FVI > OFB	Steinmann (1974)

given predictability, achievement ($r_a = GR_eR_s$) depends on these two components. This is because R_e has little effect: it appears that as environmental predictability increases so does an individual's predictive ability. To summarize the effect upon behavioral criteria Balzer et al states:

> ...the lens model equation is not only a statistically correct decomposition of achievement, it is also an analytical tool that gives us insight into the dynamics of

achievement. People are capable of improving their achievement by increasing both knowledge and control; CFB is a means of enhancing both. (1989, p. 422)

TABLE 10. STUDIES ON THE IMPACT OF CFB UPON ACHIEVEMENT.

Type of Feedback	Results	Study
CFB + OFB	No effects CFB > CFB+OFB TI+CI+FVI ≈ TI+CI+ FVI+OFB > OFB	Schmitt et al (1976) Schmitt et al (1977) Todd and Hammond (1965)
CFB, OFB	TI≈TI+CI+FVI>OFB	Nystedt and Magnusson (1973)
CFB, No FB	CFB > No FB CFB > No FB	Balke et al(1973) Fero (1975)
TI + CI	CFB > OFB No increase CFB > OFB CFB > OFB	Adelman (1981) Clover (1979) Hoffman et al (1981) Lindell (1976)
TI, TI + CI + FVI	TI+CI+FVI ≈ TI	Galbraith (1984)
CI, TI + CI	TI+CI > TI > OFB	Hammond and Boyle (1971)
TI	Increased	Deane(1970, cited in Hammond and Boyle, 1971)
FVI	No studies	
CI	Decreased Increased	Flack and Summers (1971) Stang (1985)
TI + FVI	Increased	Newton (1965)
TI + CI + FVI	Increased Increased	Newton (1965) Stang (1985)
TI + CI + FVI, TI + FVI	TI+CI+FVI ≈ TI+FVI > OFB	Steinmann (1974)

6. Issues

a. Contributions of Individual CFB Components

Only a few experiments or studies have examined the individual effects of TI, CI, and FVI on improved performance. The results so far are not conclusive enough to show which component contributes most to judgment policy learning and understanding. Trends however, suggest that TI may be the primary contributing component and that CI may be of lesser value. It must be noted that most experiments are of limited cognitive complexity and that as complexity increases so may the contribution of CI. (Balzer et al, 1989)

b. The Role of CI

The greatest future growth within the field of CFB may come from the use of CI. According to Doherty and Balzer:

> We believe that the future growth of CFB applications will be in the measurement of utilities rather than in modelling environments. We see more people benefitting from CFB, specifically the CI component, by an increased understanding of, and ability to communicate, what they personally value ... rather than from an improved prediction of an uncertain environment ... If the promise of wide availability of CFB software packages for general purposes is fulfilled, there may be another benefit to users. Since the user will have to decide upon the dimensions (or cues) and their levels, the user will not only have the benefit of having to make trade-offs, receiving CFB, etc., but will also have the benefit of the insights gained from the decomposition of the problem common to the early stages of decision analysis. (1988, p. 189)

c. *Policy PC*

Policy PC: Judgment Analysis Software[1] is a program written for IBM and compatible personal computers. This is an example of software developed on the lens model single system paradigm, using statistical methods to make models of human judgment. The program enables the user to construct a series of problem characterizations or scenarios and to then extract judgments about them. It calculates the regression measures and returns cognitive feedback as graphic displays. It can analyze the judgments of up to eight decision makers, use up to eight cues (text or numeric) in 100 cases, and compute the statistics for each task, judge, and policy.

Policy PC and similar programs were not intended for use in research into expert system development. Though it has some features that would be desirable in such an effort, it lacks some properties necessary for expert system development. For example, this program will not allow an expert to iteratively refine his or her model. It does not have a method for defining a cue in terms of other "subcues" in a hierarchical manner. Cues cannot be temporarily modified, added, or deleted so that a model can be quickly refined in response to changing feedback. The feedback (a

---

[1]Executive Decision Services, Inc., Albany, NY

combination of TI + CI), as presented in simple text mode, is not readily assimilable or intuitive. Additionally, it is not accompanied by much textual explanation, an important form of feedback. *Policy PC* is also incapable of capturing an expert's initial impressions of cue correlations, as a "starting point", should the expert already have a feel for what his or her policy is.

G. SUMMARY

This chapter presented a modern psychological theory that explains the manner in which information is used in decision making. Brunswik's probabilistic functionalism and the quantification methods provided by subsequent researchers, provide a mathematical means of *modeling mental processes*. This simple and straightforward way of constructing a linear model provides a technique for testing hypotheses about the way individuals combine information and exercise judgment.

A major contribution of this theory is to stress representativeness in research design. The experiments and investigations must have an ecological validity or "true-to-life" modeling if they are to be at all successful in discovering or representing judgmental processes. With respect to representative design, Hoffman observed, "In focusing upon the individual as the unit of research while at the same time preserving methodological rigor it becomes possible to achieve a level of psychological description which would otherwise be quite difficult" (1960, p. 131).

The effects of cognitive feedback have been extensively studied for almost 30 years. The majority of the research findings demonstrate that when cognitive feedback is given to an individual it will improve performance with respect to knowledge, control, and achievement. This indicates that cognitive feedback is a useful tool for representing a person's mental model and for altering that mental model as well. This is only effectively accomplished when information regarding relationships is returned to an individual rather than information concerning outcomes.

The wide variety of settings in which cognitive feedback has been successively used show that broad applicability is possible. It can be used to resolve interpersonal or interorganizational conflicts. It also can allow for the resolution of *intra*personal conflicts, that is, it can clarify and enhance an individual's view into his or her own value system. Cognitive feedback can improve a person's judgment and it can improve learning.

The computer is the logical device for the employment of cognitive feedback in analyzing judgment policy and providing the feedback in a timely manner. The variety of possible formats and methods allow the feedback process to be tailored to an individual. The following chapter provides a high level description of a proposed cognitive feedback system for use in knowledge acquisition for expert systems.

IV. A PROPOSAL FOR AN AUTOMATED KNOWLEDGE ACQUISITION TOOL USING COGNITIVE FEEDBACK

A. INTRODUCTION

The previous chapter established the validity of cognitive feedback as a tool for enhancing the learning process and for understanding how an individual may perceive his or her environment. More importantly, empirical studies have shown cognitive feedback to be useful in capturing a person's policy or mental model. Since this is based upon Brunswik's probabilistic functionalism, depicted in the lens model, the expert's policy or knowledge can be represented as a simple numerical model (linear equation). (Hursch et al, 1964; Hammond et al 1964; Tucker, 1964)

This chapter proposes an automated knowledge acquisition tool known as KARCOF, *Knowledge Acquisition and Representation with COgnitive Feedback*. The tool is described with state transition diagrams and computer screens, set within the context of a task that evaluates personnel performance. The program interacts with an expert personnel evaluator (user) to elicit a policy that will determine the promotability of individuals within a particular field of expertise. KARCOF uses the single system paradigm and through presentation of cognitive feedback in graphical form, allows the expert to iteratively refine their policy. When the expert is satisfied that the program has successfully captured their policy or knowledge, it is stored or represented as a numerical model in a linear regression equation.

KARCOF is more than a knowledge acquisition tool, it also can be considered as the developing core of a Knowledge Support System (KSS). KSS's encompass tools for knowledge engineering and support for human knowledge processes. KARCOF can be used toward this end by returning an individual's policies, or by enabling them to clarify a decision making process. KARCOF has several other characteristics of a KSS. These include domain independence (cognitive feedback has shown broad applicability with respect to tasks (Balzer et al, 1989)), direct interaction with the expert, provision for validation, a sound theoretical foundation, and the ability to incorporate different forms of knowledge and relationships between knowledge. (Shaw and Gaines, 1987a)

B. SPECIFICATION

A top level view of how the KARCOF program operates is illustrated by Figure 3. This is a state transition diagram that uses Wasserman's (1985) methodology for specifying and implementing interactive information systems. The following is a brief listing and definition of the diagram components:

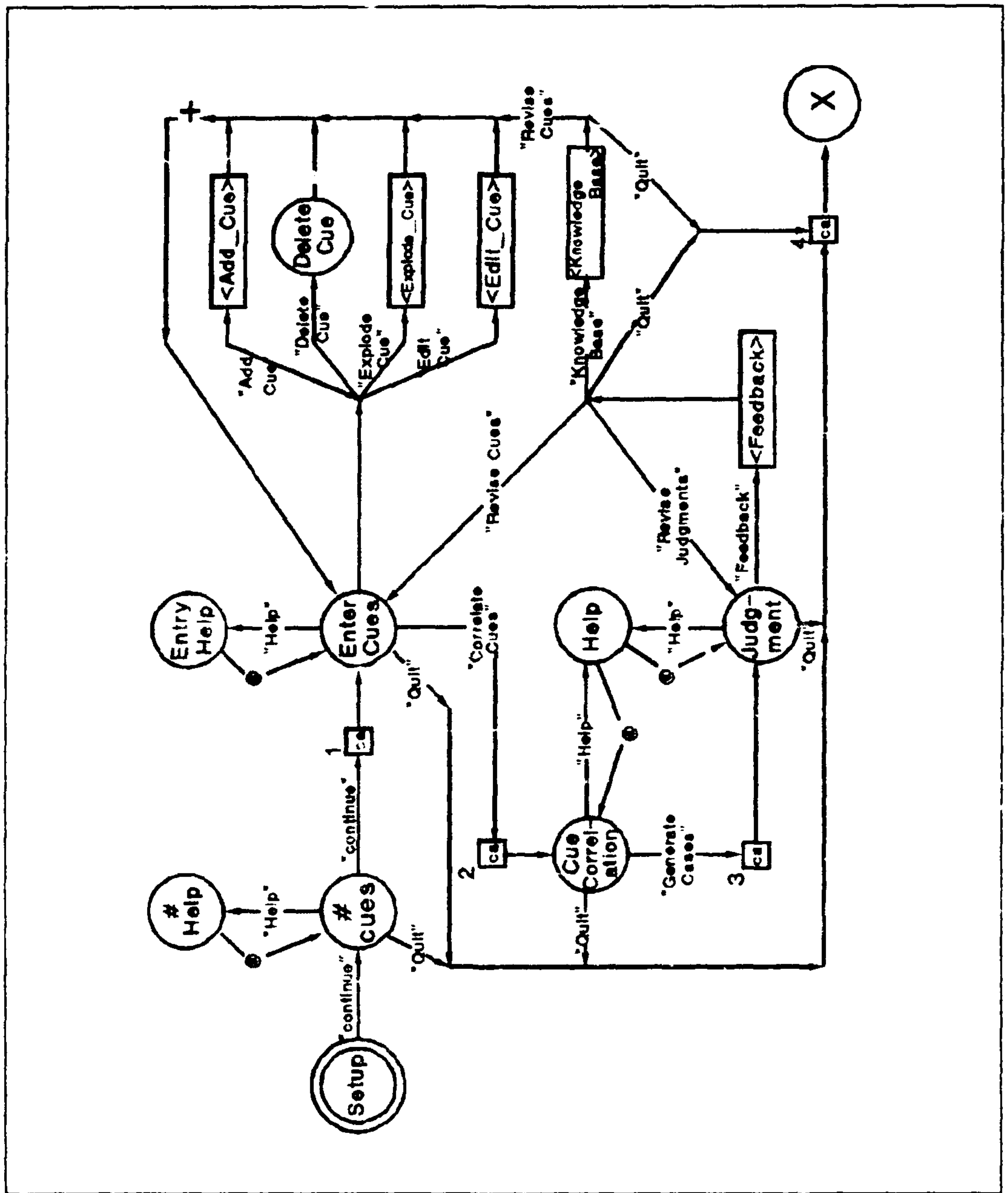


Figure 3. Top level state transition diagram.

--*Nodes* are shown by a circle and represent a stable state awaiting some user input. A *node* displays a message.
--*Arcs* are shown by arrows that connect the nodes to one another. The arc is a state transition caused by some user input. The arc is labeled by the input or variable assignment that causes the transition.
--An *operation* is shown by a small square with "ca" (call action) and is associated with an integer that differentiates the actions. It may be associated with a transition to show a particular action that is taken when an arc is traversed. The action also may be associated with more than one arc.
--A *subconversation* is shown by a rectangular box with an associated diagram name. It is a lower level diagram or module to which control is passed. The new diagram is traversed to the exit and control is then returned to the top level.
--The "+" denotes return to a previous node without intervening user input.
--The "●" denotes resumption of a program after it has paused for access to the help feature.

Each node in Figure 3 represents a computer screen and each rectangle represents another diagram. All are included in subsequent figures to illustrate how an expert would use this tool.

1. Setup and Initial Steps

The session begins with the user entering the data that will identify this knowledge base when the session is complete. The first node in Figure 3 is the "Setup" phase and it is here that the purpose or name of the policy, name of the expert, and name of the file in which the data is to be stored, is entered. The elicitation begins with the user specifying the number of cues, up to a maximum of eight, that will be used in the model, as shown in Figure 4. Input is via the keyboard and the functions at the bottom of the screen are

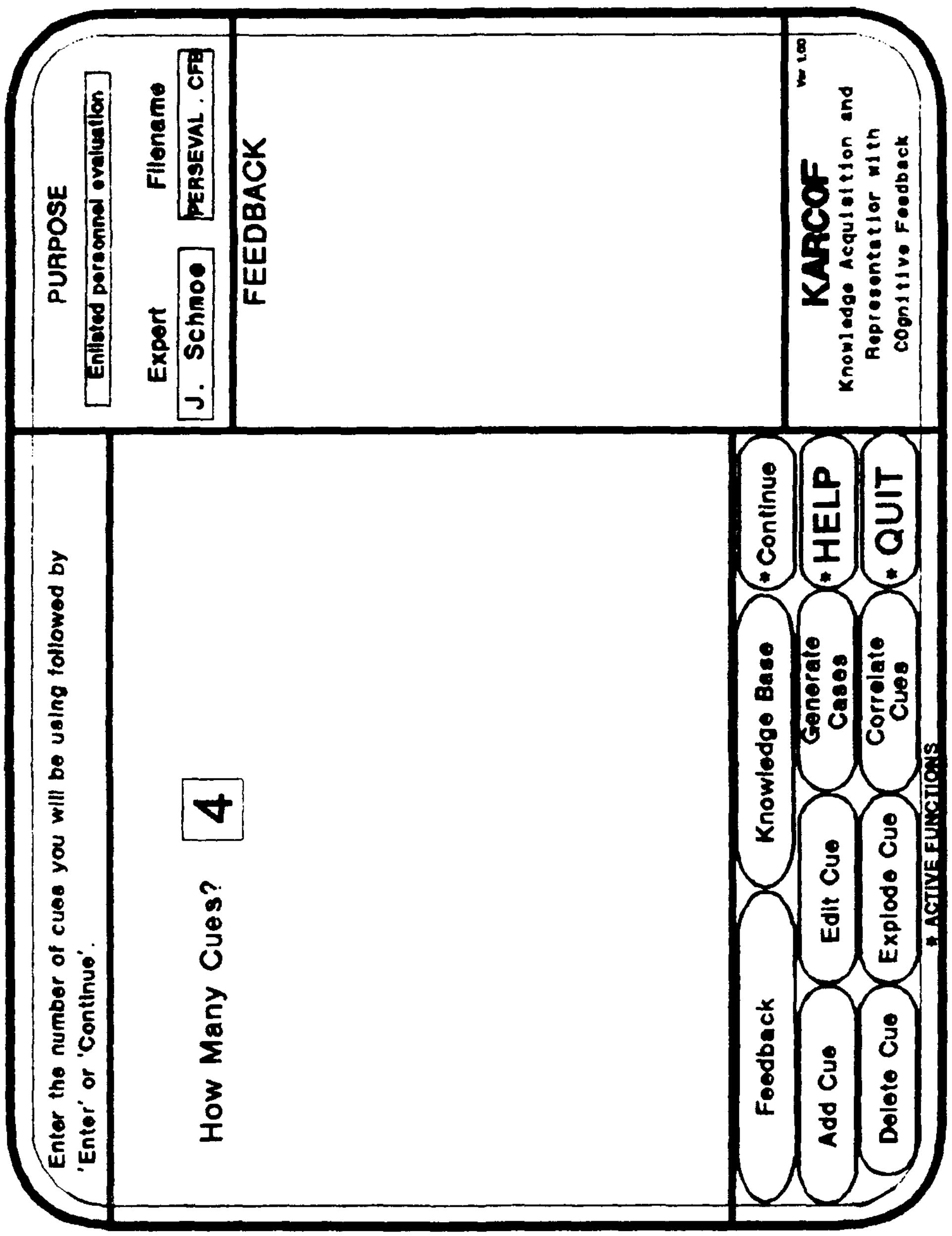


Figure 4. Screen for specifying the number of cues.

accessed by mouse or cursor keys. Available functions, at any particular step, are denoted by an asterisk. At the top of the screen is a message bar showing what is required for the present step and how to move to the next phase.

The next step, Figure 5, requires the user to specify the nature of the cues in a popup window. There are two available choices: cardinal (for a discontinuous scale), and numeric (for continuous values). For each cue the user enters a name and a value range. In this example, each cue can assume integer values between one and ten inclusive, though all the cues need not have the same scale. As an anchor point for the scale, one means "worst" or "least desirable", and ten means "best" or "most desirable". If the user desires additional information on any step, help is always available through a popup, scrollable, context sensitive window, as in Figure 6.

When determining the number of cues to be used, the expert should always be mindful that only the minimum necessary for the judgment should be entered. Judges will most often include too many cues: the important ones and some unimportant ones (Stewart, 1988). Feedback after the first iteration or so should suggest to the user which cues are used the least so that these can be deleted.

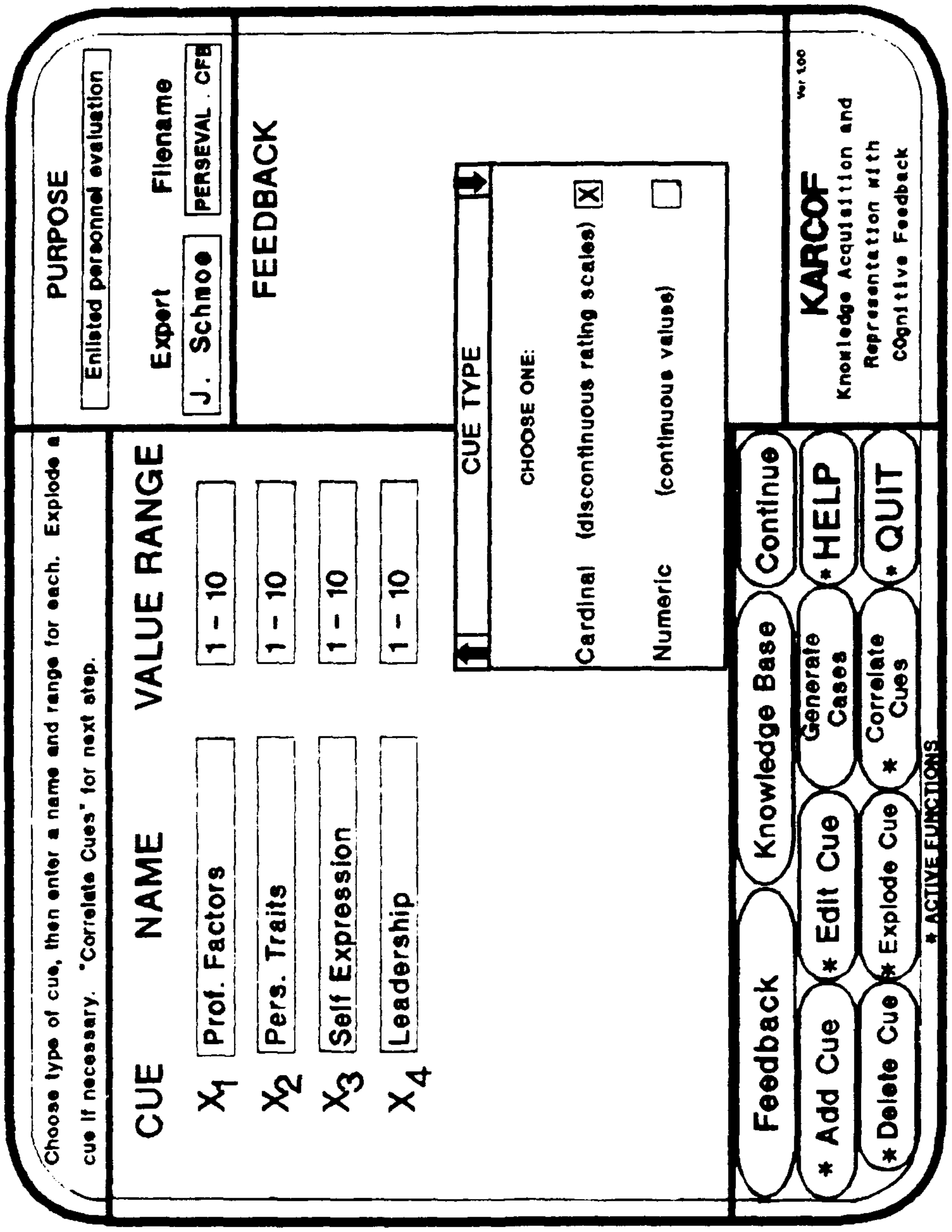


Figure 5. Screen entering the number of cues.

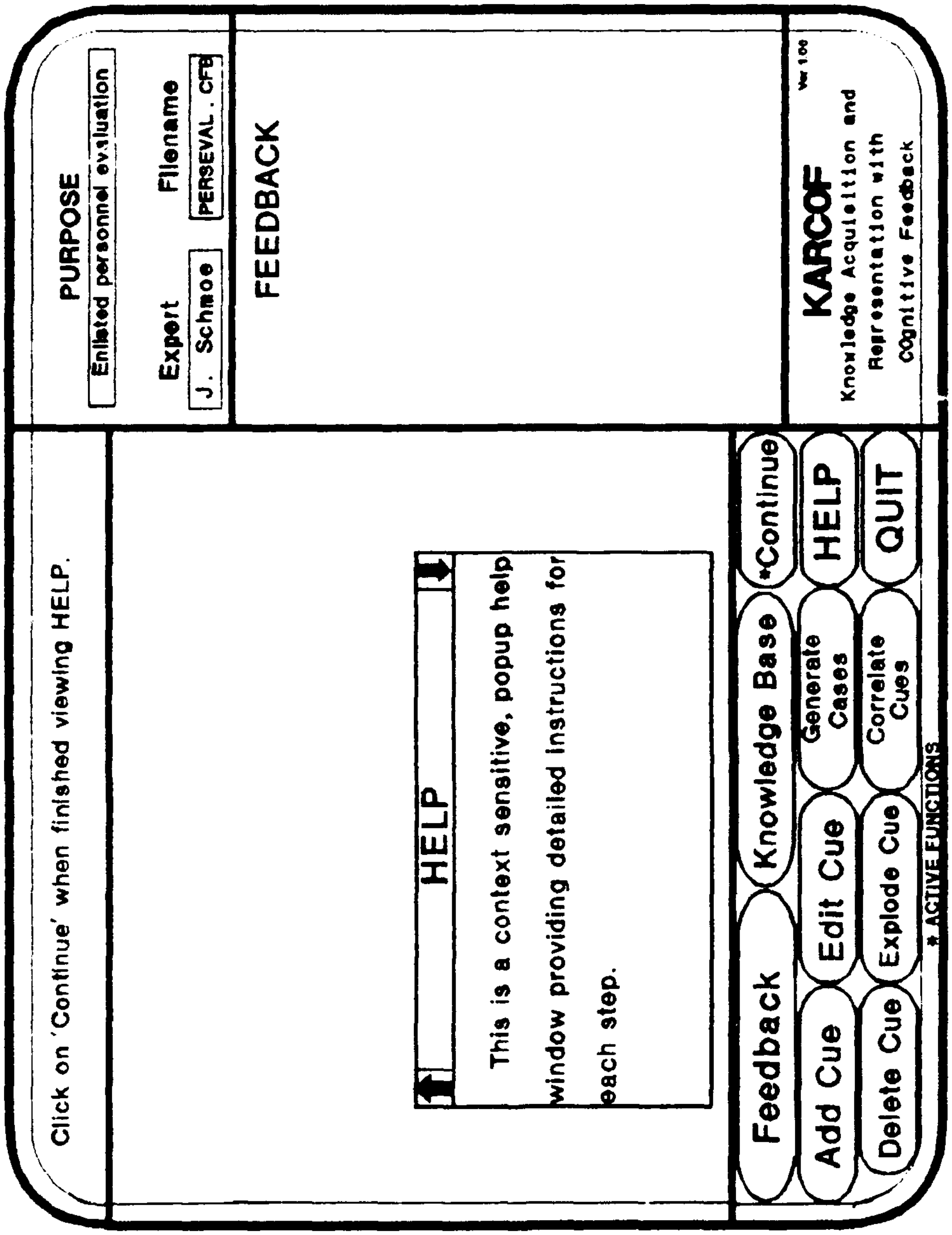


Figure 6. Context sensitive, popup help window.

At the next step, Figure 7, the user enters his or her estimate of the correlation between each combination of cues. Since humans deal poorly with statistical estimations, a graphical representation as a number line, is provided for the entry of estimates. Figure 7 shows the cue pair X_1-X_2 to have a weak positive correlation, X_1-X_3 to have a strong positive correlation, and X_2-X_3 to have a weak negative correlation. This step is necessary because the principle of parallel concepts states that cues in the cognitive task should match cues in the cognitive system. The environmental cues are unknown, of course, so they must be estimated by the expert. These subjective cue intercorrelations may be reasonably accurate if the expert has observed occurrences of similar correlations in real life over many cases (Stewart, 1988). This example uses four cues, so several more screens of correlations would be necessary.

2. **Case Generation and Judgments**

To capture an expert's policy or knowledge, KARCOF must obtain a series of judgments made by the expert over a set of representative cases. Stewart (1988) has set forth some guidelines for creating an algorithm that determines the number of cases necessary to obtain a statistically stable model of the expert's policy. The requirement for stability sets a lower bound on the number of cases required. However, this number cannot be precisely calculated because it depends

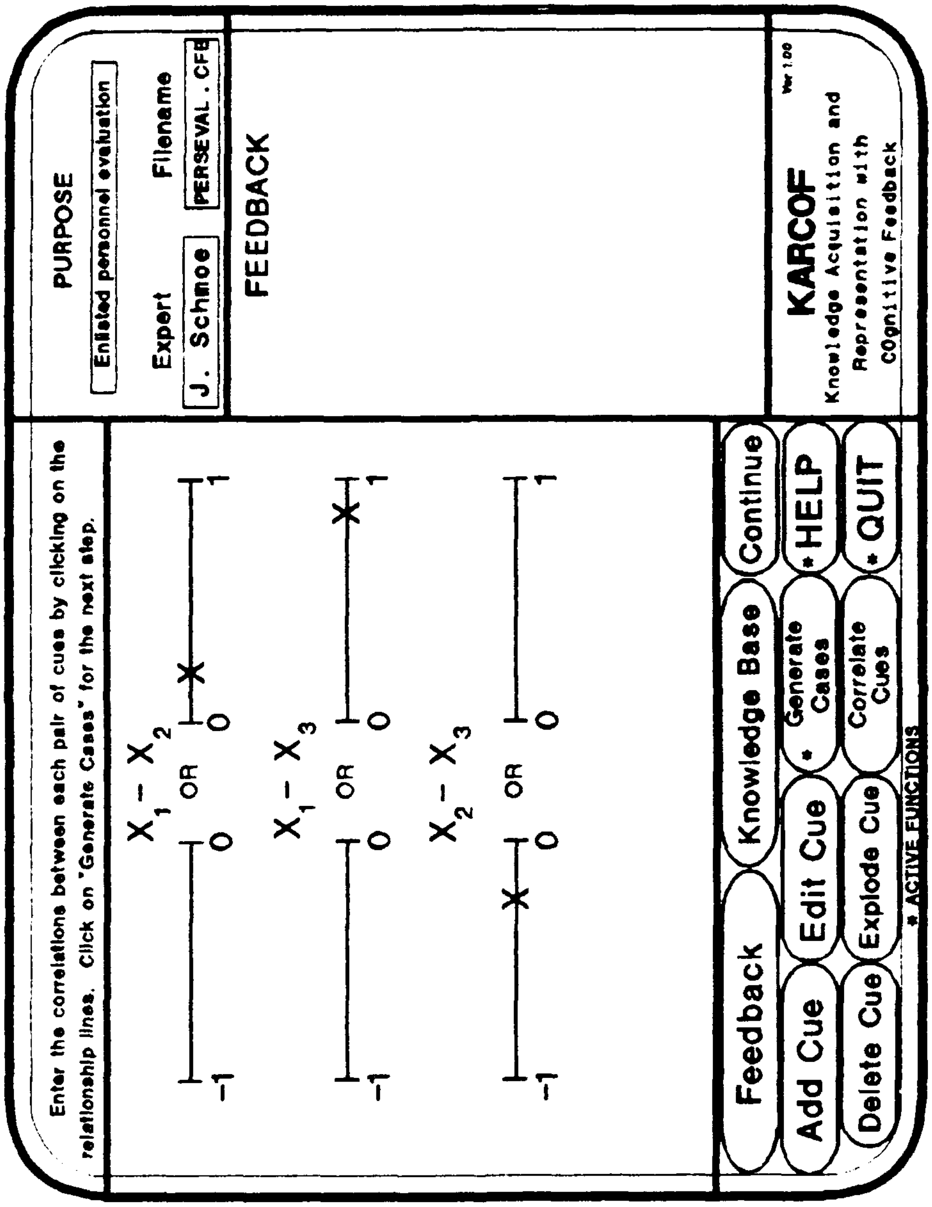


Figure 7. Screen for entering the correlations between combinations of cues.

upon the complex interactions of three factors:

- --The *number of cues*, which when increased requires an increase in the number of cases to maintain stability.
- --The *fit of the model to the judgments* as shown by the multiple correlation. If the fit is good, fewer cases are required.
- --The *Cue intercorrelations*, when zero, result in greatest stability. If correlations exist, then more cases will be required.

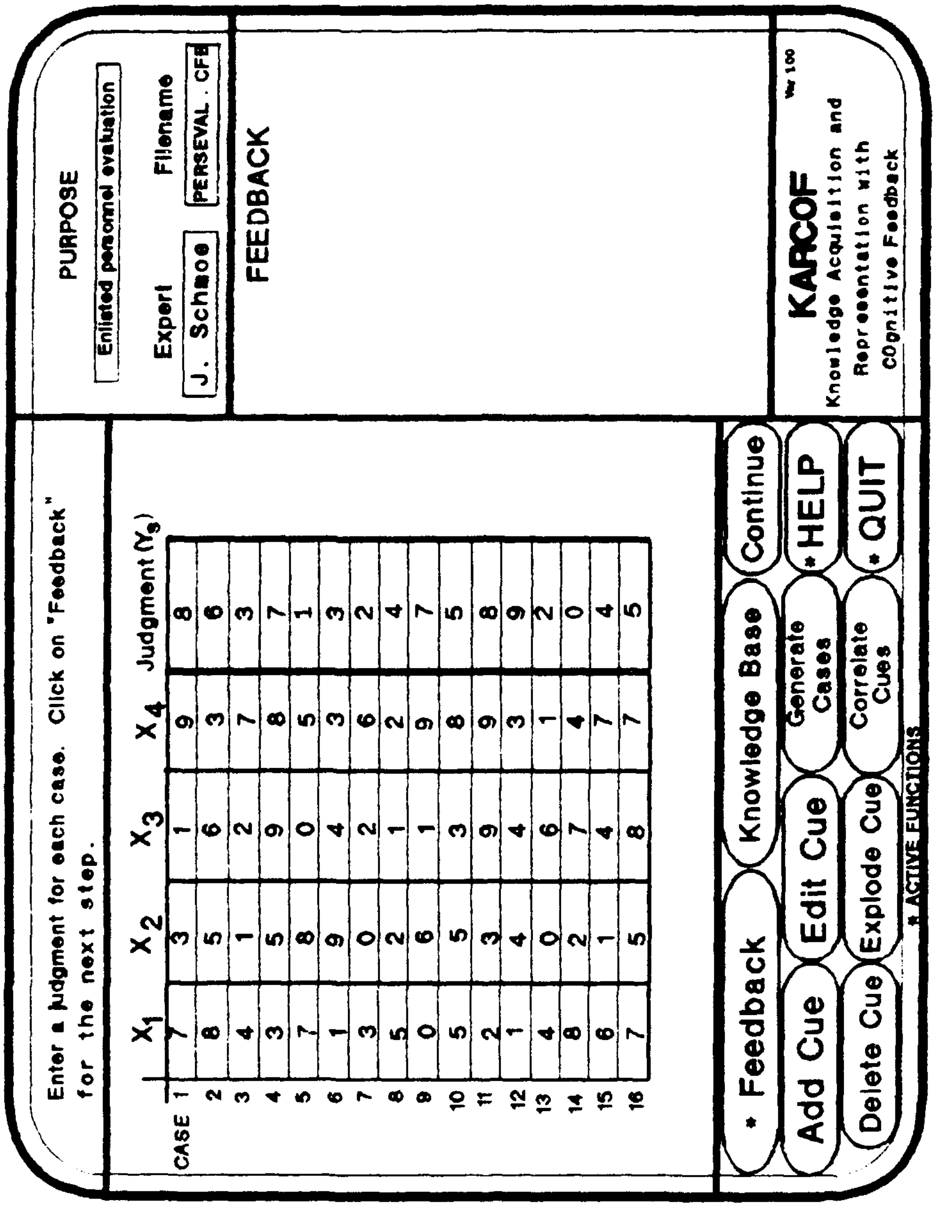
Various levels of each of these factors affects the *standard error of the regression coefficient* (SE), which can be interpreted as an estimate of the instability of the model. A high SE suggests greater instability and unreliability of the analysis.

The knowledge engineer administering KARCOF must choose an appropriate SE before beginning an elicitation session. An SE of .1 is generally relevant for most circumstances, and when combined with a known level of cue intercorrelations, a suitable number of cases can be determined. Past research has shown that judgments over 30 cases will yield a statistically stable model with this example's number of cues and level of intercorrelation. Generally, 30 cases for the first four cues and 5 cases for each additional cue will yield a stable model. (Stewart, 1988)

Cook (1976) as cited in Stewart (1988), submits that the standard statistical assumptions may not apply to this application (judgment analysis). He found that stability could be achieved with fewer cases than is suggested by statistical theory. If true, then the above procedure will be too conservative. Cook's findings have yet to be duplicated, but the above algorithm should yield the appropriate accuracy.

KARCOF's algorithm, after determining the number of cases, generates a random number matrix from a clock time seed. The matrix is an assignment of a random number to each cue over the set of required cases. The algorithm then performs the appropriate statistical operations upon the matrix to compute the standard error and the cue intercorrelations. These two measures are then compared against the SE specified by the knowledge engineer and the cue intercorrelations specified by the expert. If there is not a resulting match, within a certain tolerance range (say, 10%), to prevent an inordinate number of iterations, the above steps are repeated. When a match of SE and cue intercorrelations is finally achieved, the cue values are rescaled according to the range prescribed by the expert.

The final matrix of cue values over the prescribed set of cases is then presented to the expert, as in Figure 8. The expert performs a judgment for each case and enters it in the "Judgment (Y_s)" column of the matrix. The example in Figure 8 would require another screen of 14 judgments to be executed before this phase is complete. An expert can generally make 40-75 such judgments in one hour (Stewart, 1988).



CASE	X_1	X_2	X_3	X_4	Judgment (Y_s)
1	7	3	1	9	8
2	8	5	6	3	6
3	4	1	2	7	3
4	3	5	9	8	7
5	7	8	0	5	1
6	1	9	4	3	3
7	3	0	2	6	2
8	5	2	1	2	4
9	0	6	1	9	7
10	5	5	3	8	5
11	2	3	9	9	8
12	1	4	4	3	9
13	4	0	6	1	2
14	8	2	7	4	0
15	6	1	4	7	4
16	7	5	8	7	5

Figure 8. The expert enters a judgment for each generated case.

3. Feedback

When the expert has completed all the necessary judgments, control of the program is passed to the "Feedback" subconversation or module, Figure 9 (also see Figure 3). All the statistical measures required by the single system paradigm are computed and the expert is then given a choice of how he or she wishes to view the feedback, as shown in Figure 10. Three choices, with a sample of each are shown in this step:

--The *relationship (lens) model* is a graphical representation of all cues and subcues. Strength of cue correlation to judgment and of subcue to parent cue is shown by the thickness of the connecting line.

--*Function forms* represent the relation between a single cue and the judgment. It is presented on a simple X-Y graph.

--*Decision weights* are the standard regression coefficients, or beta weights, of each cue expressed in percentages. This eliminates differences due to units of measure between each cue and estimates the direct impact a cue has on a judgment if the other cues are held constant. The data is presented on a simple vertical bar chart. A negative decision weight is shown below the X-axis.

The expert views the feedback in Figure 11, and has the option of seeing each type in succession. A detailed verbal explanation of each form of feedback is available by accessing the Help facility. When finished viewing the feedback, the user may revise the cues in any manner, revise the judgments, or view the knowledge base (validation). (Stewart, 1988)

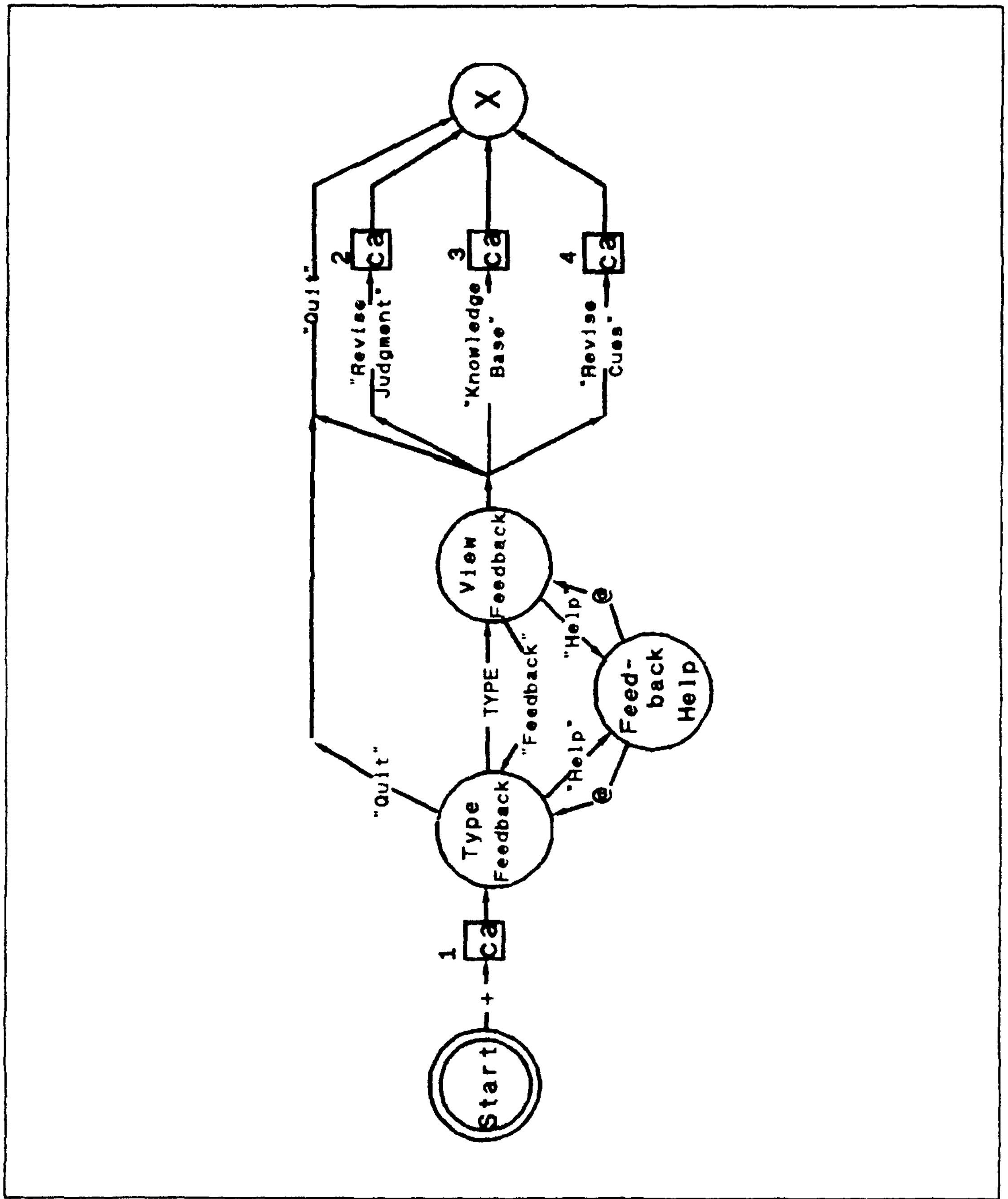


Figure 9. State transition diagram for the Feedback module.

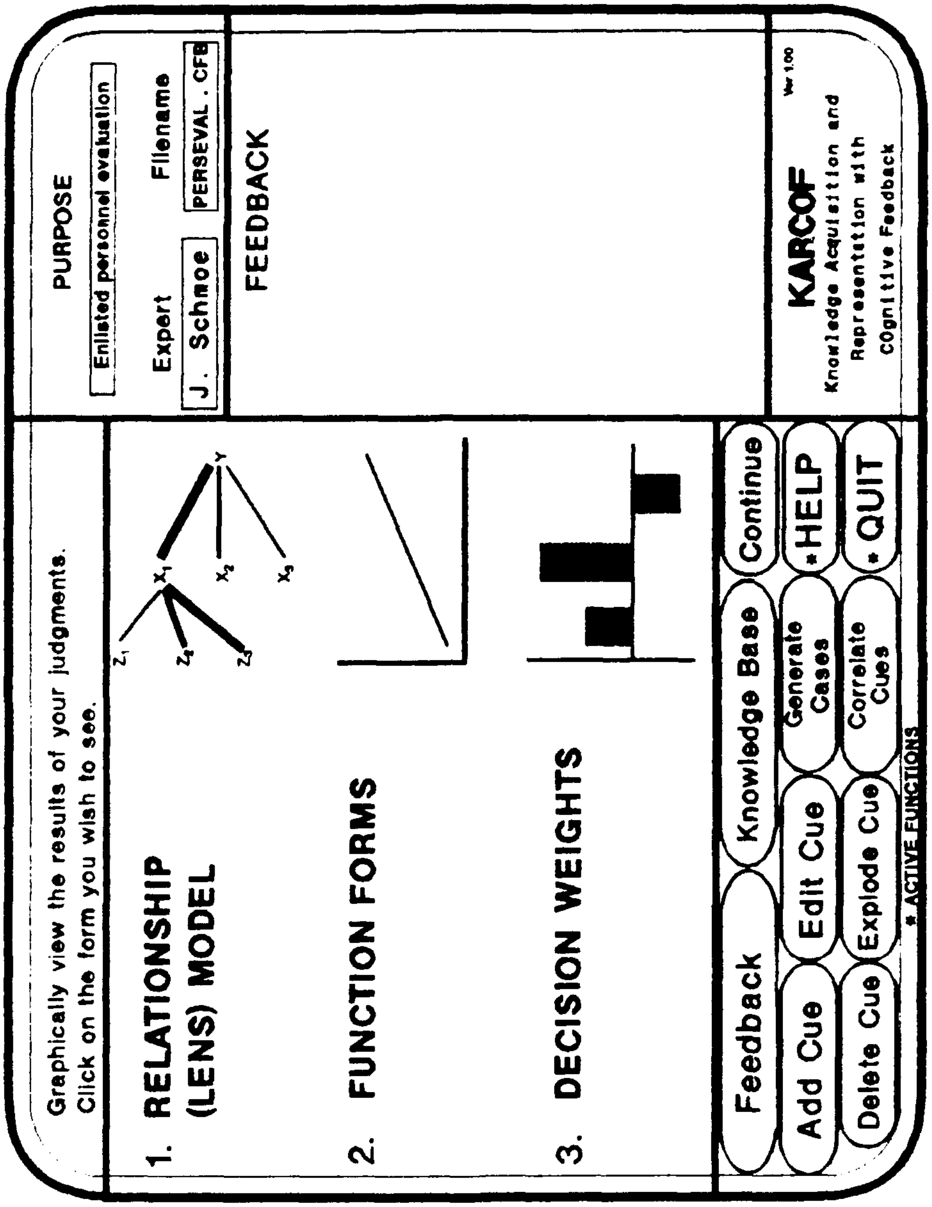


Figure 10. Screen for selecting the type of feedback.

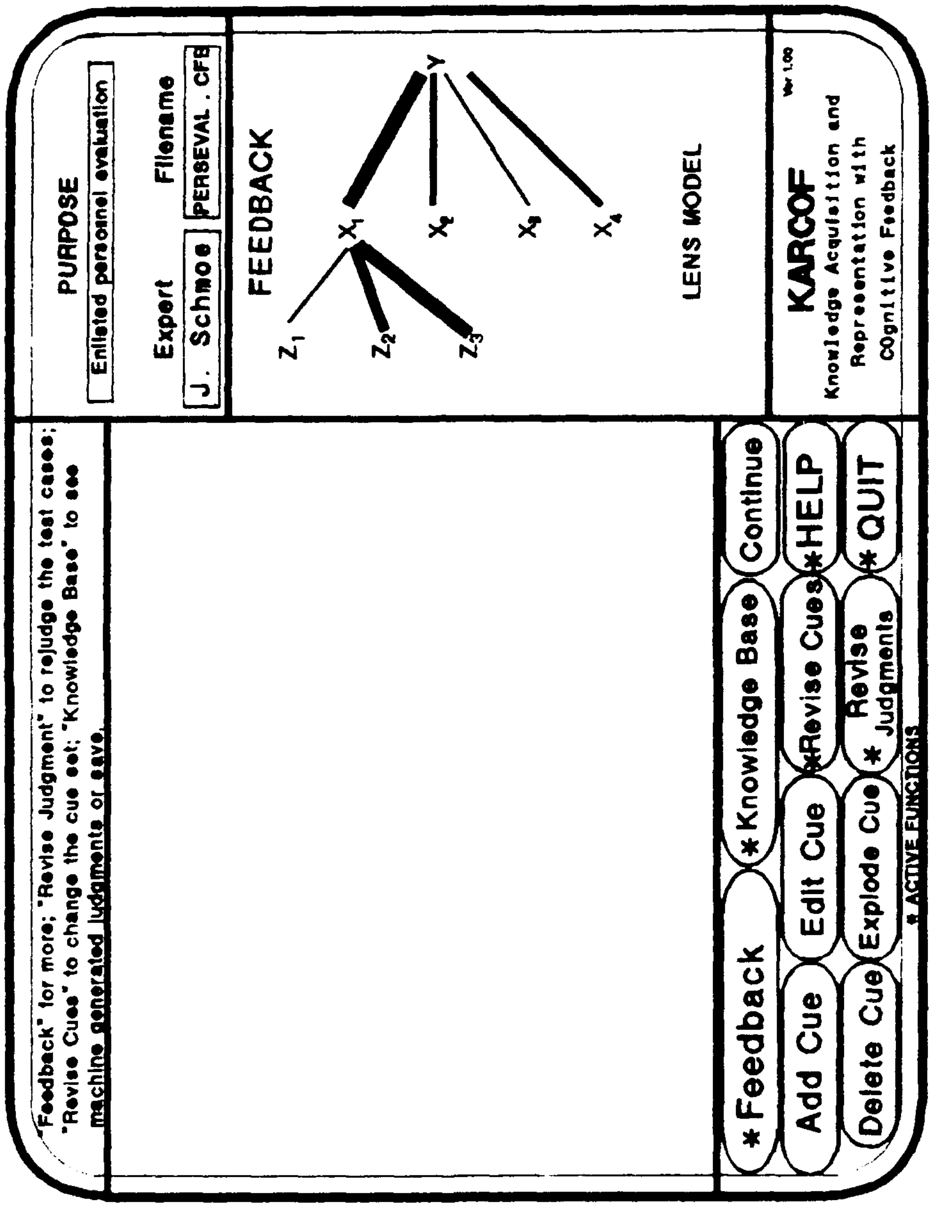


Figure 11. Screen for viewing feedback.

4. Validation

Should the expert feel satisfied the feedback indicates that his or her policy may be effectively captured, the next step is to select "Knowledge Base", an option in Figure 11. Control of the program is then passed to the "Knowledge Base" subconversation or module, Figure 12. KARCOF generates another random number matrix for each cue and case, and the requirement for matching cue intercorrelations and a matching SE is again satisfied. KARCOF applies the expert's captured policy, in a multiple regression equation, to predict a judgment for each case, Figure 13.

The expert inspects the machine generated judgments and if satisfied, elects to save the knowledge base and terminate the session, Figure 14. However, KARCOF may not have produced satisfactory judgments, indicating to the expert that his or her policy has not yet been faithfully captured. If this is the case the expert may elect to save the knowledge base and continue the session at another time, or revise the cue set and subsequently do another iteration of cue correlations, judgments, and feedback.

5. Refinement

After viewing feedback from the first set of judgments, the expert may choose to rejudge the original set of cases. He or she may have under or overemphasized certain cues at the expense of others. This would result in a model

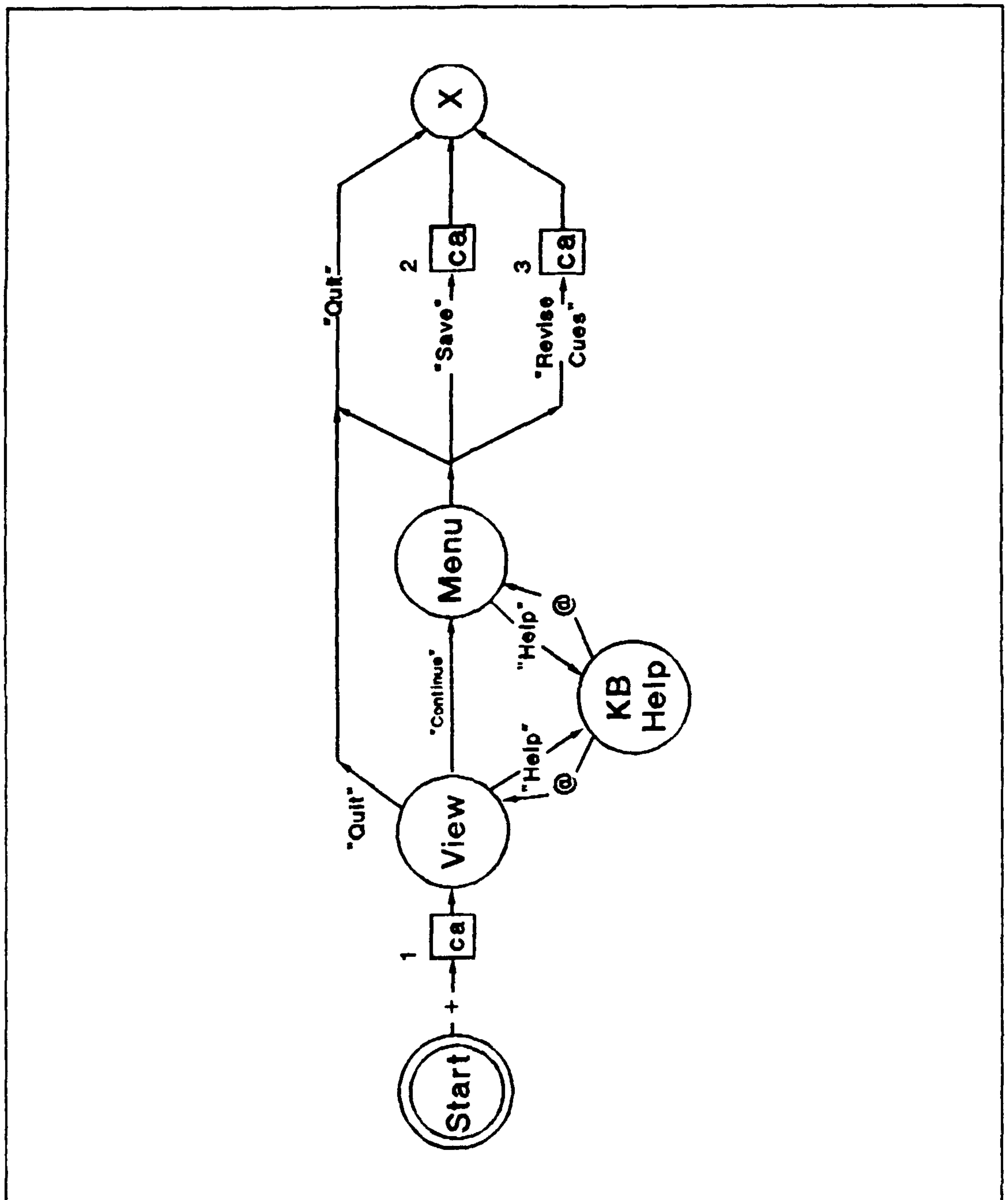


Figure 12. State transition diagram for the Knowledge Base module.

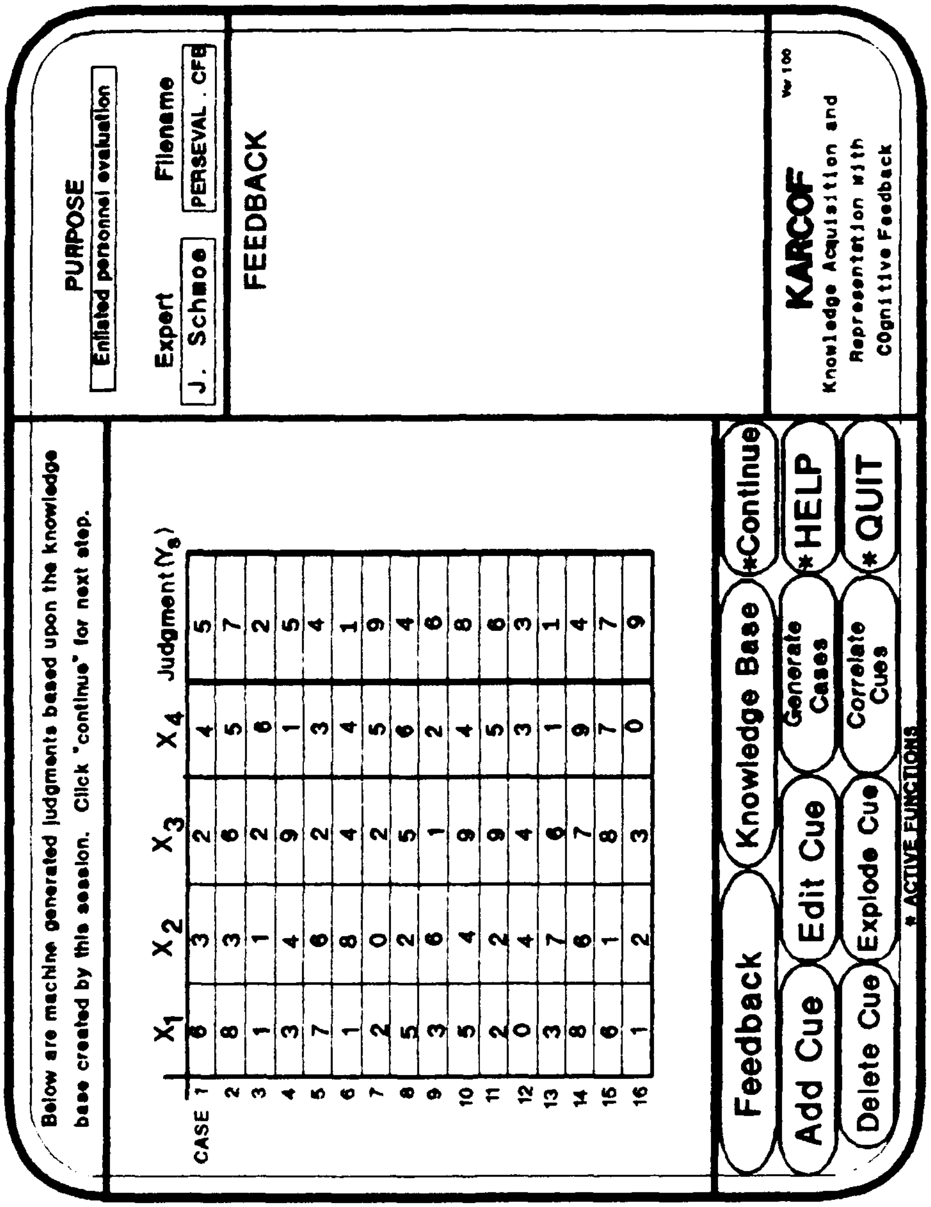


Figure 13. Validation of the knowledge base. Judgments are generated for a new set of cases.

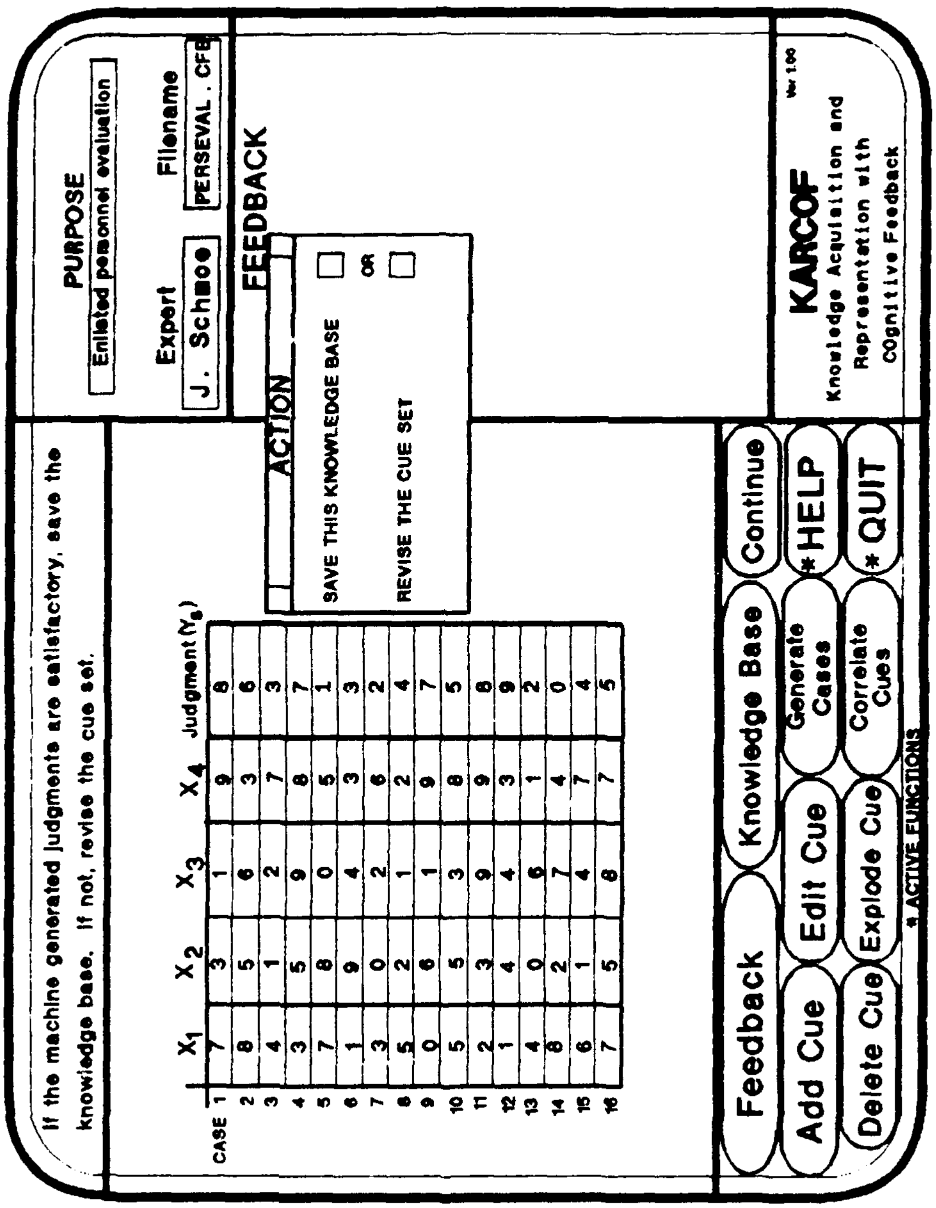


Figure 14. The expert may elect to save or revise the knowledge base.

that does not truly reflect the expert's policy. In Figure 11 the expert selects "Revise Judgments" to return and rejudge the original set of cases. When completed, feedback is accessed again. The user may indefinitely iterate in this fashion until satisfied with the feedback and validation of the model.

An alternative to rejudging the original cue set, which may not produce the desired feedback, is to alter the parameters of the model. When the user selects "Revise Cues" in Figure 11 four options present themselves: add a cue, delete a cue, "explode" a cue, and edit a cue. Should the user choose to add a cue, control of the program is passed to the "Add Cue" subconversation or module, Figure 15. The user is prompted for the number of additional cues to incorporate into the model, Figure 16, and for the names and value ranges of the new cues, Figure 17. After this step the expert may further refine the model by deleting, exploding, or editing cues or may proceed to correlate and judge the cues again.

Another option in the refinement process is to delete a cue. The expert may choose to do this if he or she considers a cue to be unimportant, contributing little or even detracting from the final decision. In Figure 18 the expert is asked which cues to delete. Afterward, the refinement process may continue, or the cue combinations may be recorrelated and judged.

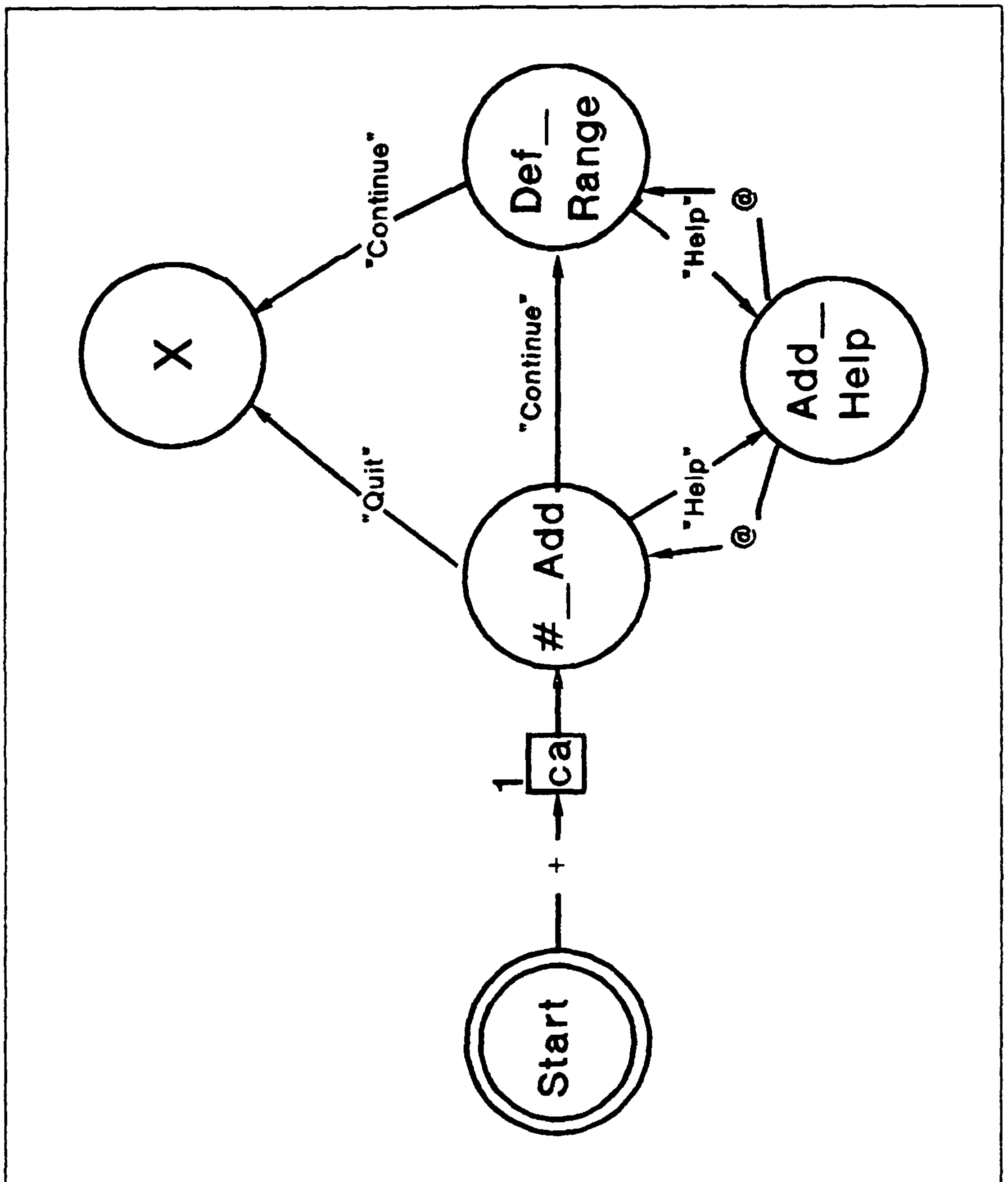


Figure 15. State transition diagram for the Add Cue module.

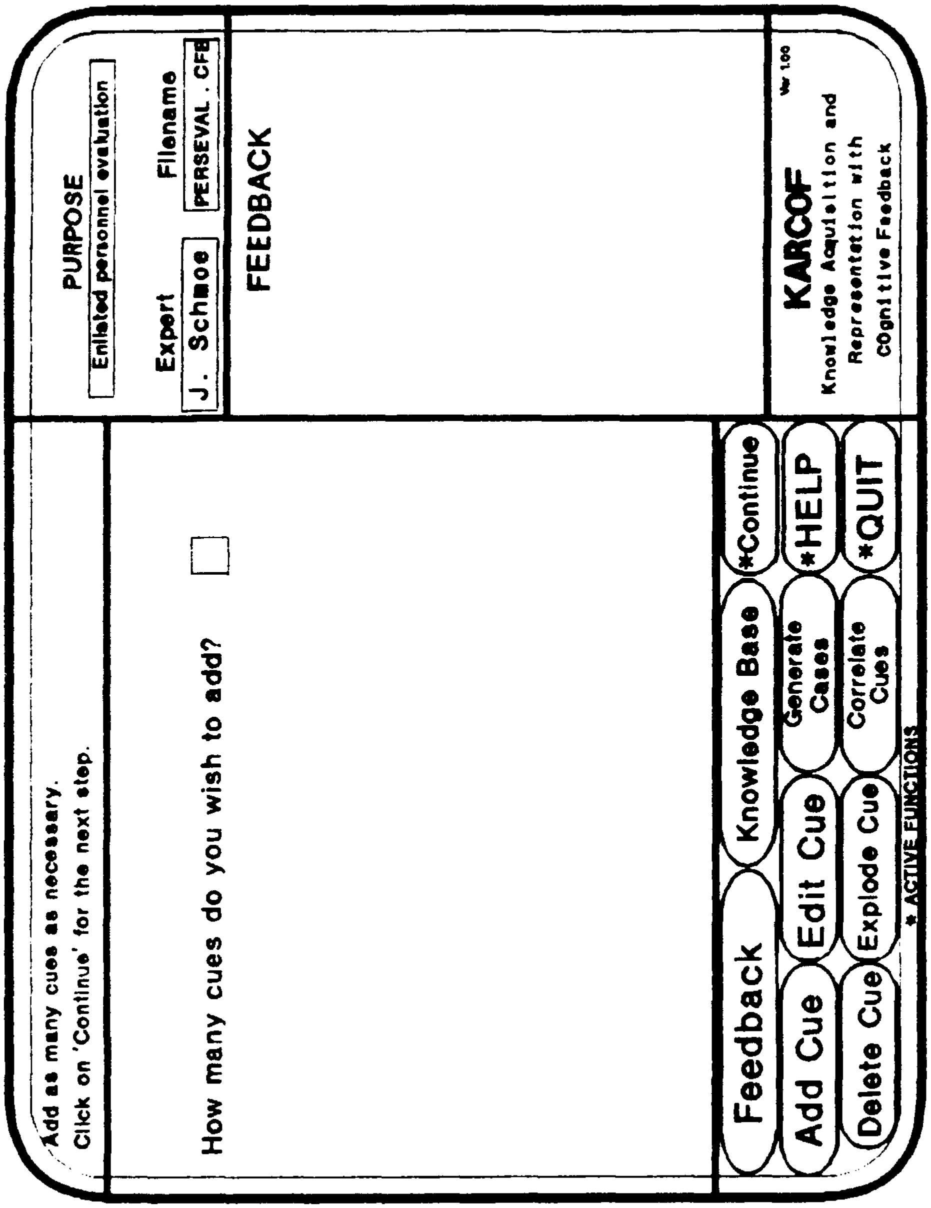


Figure 16. Screen for specifying the number of additional cues to add to the model.

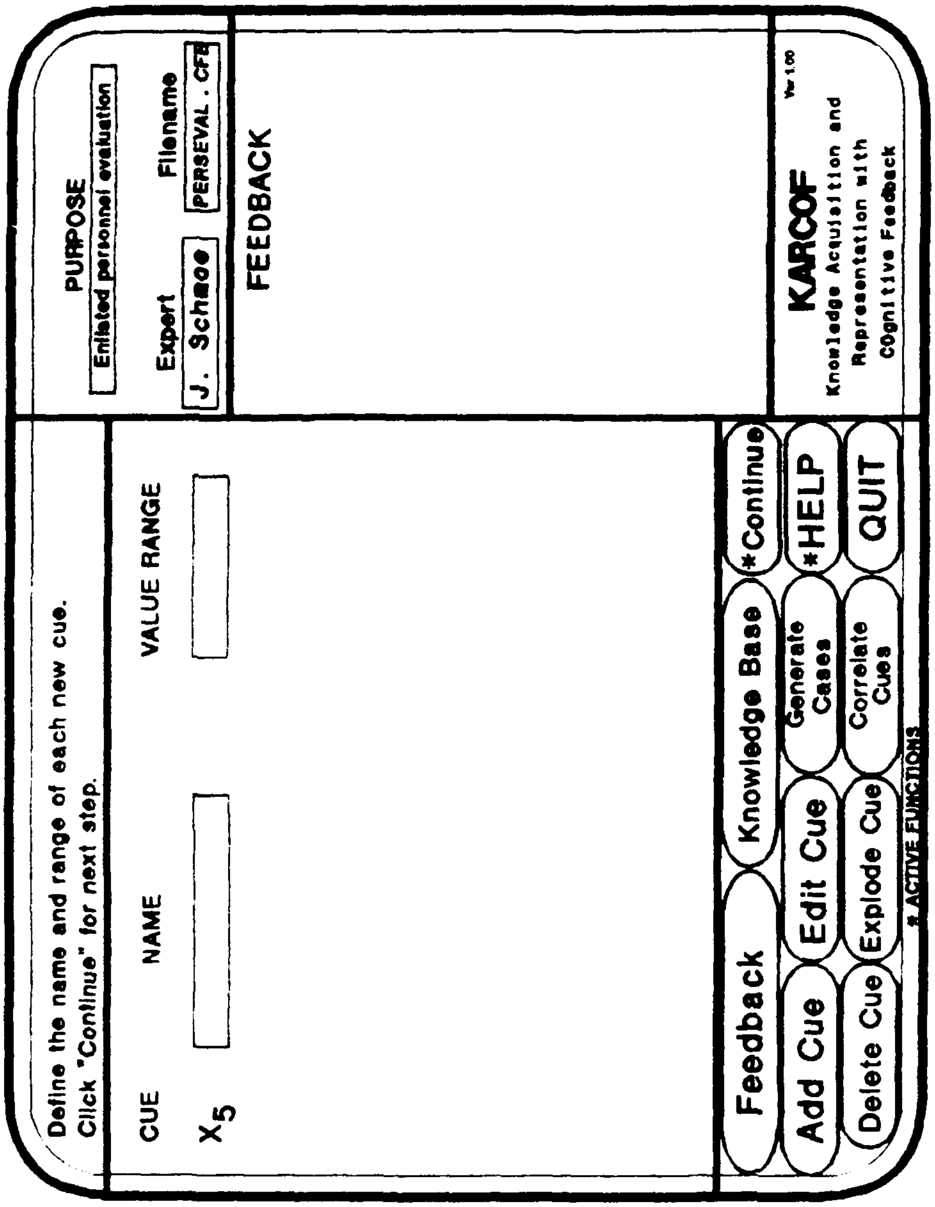


Figure 17. Screen for defining the parameters of an additional cue.

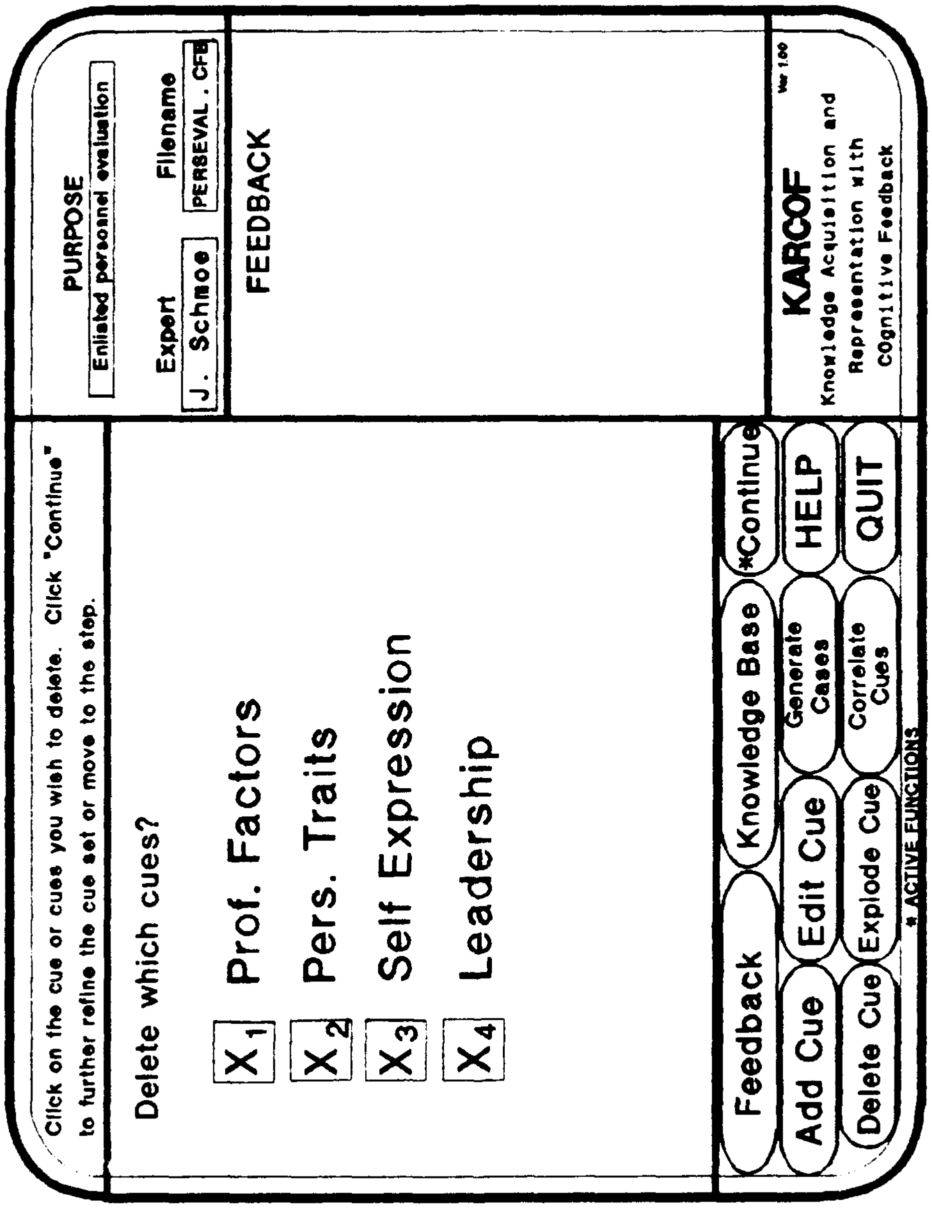


Figure 18. Screen for deleting a cue.

A third option in the refinement process is to "explode" a cue. Control of the program is passed to the "Explode" subconversation or module, then a cue can be hierarchically defined in terms of lower level subcues, as shown in Figure 19. This module contains processes and screens that are virtually identical to that of the main program. The only difference is that a parent cue is now regarded as a judgment or decision. A regression model, formulated in terms of the subcues, is created to define the parent cue through the same operations of cue correlation, case generation, judgment, feedback, refinement, and validation.

The last option in the refinement process is to edit a cue. When "Edit Cue" is selected, control of the program is passed to the "Edit Cue" subconversation or module, Figure 20. The user is given a choice of which cue to modify. The name of the cue, Figure 21, and the value range, Figure 22, may be altered. The expert may continue the refinement process or go to the steps of cue correlation, judgment, feedback, and validation.

C. SUMMARY

KARCOF is a highly interactive program that can provide an expert with instantaneous feedback to his or her decisions. Parameters of the decision model can be altered dynamically and there is little delay between the judgments and the

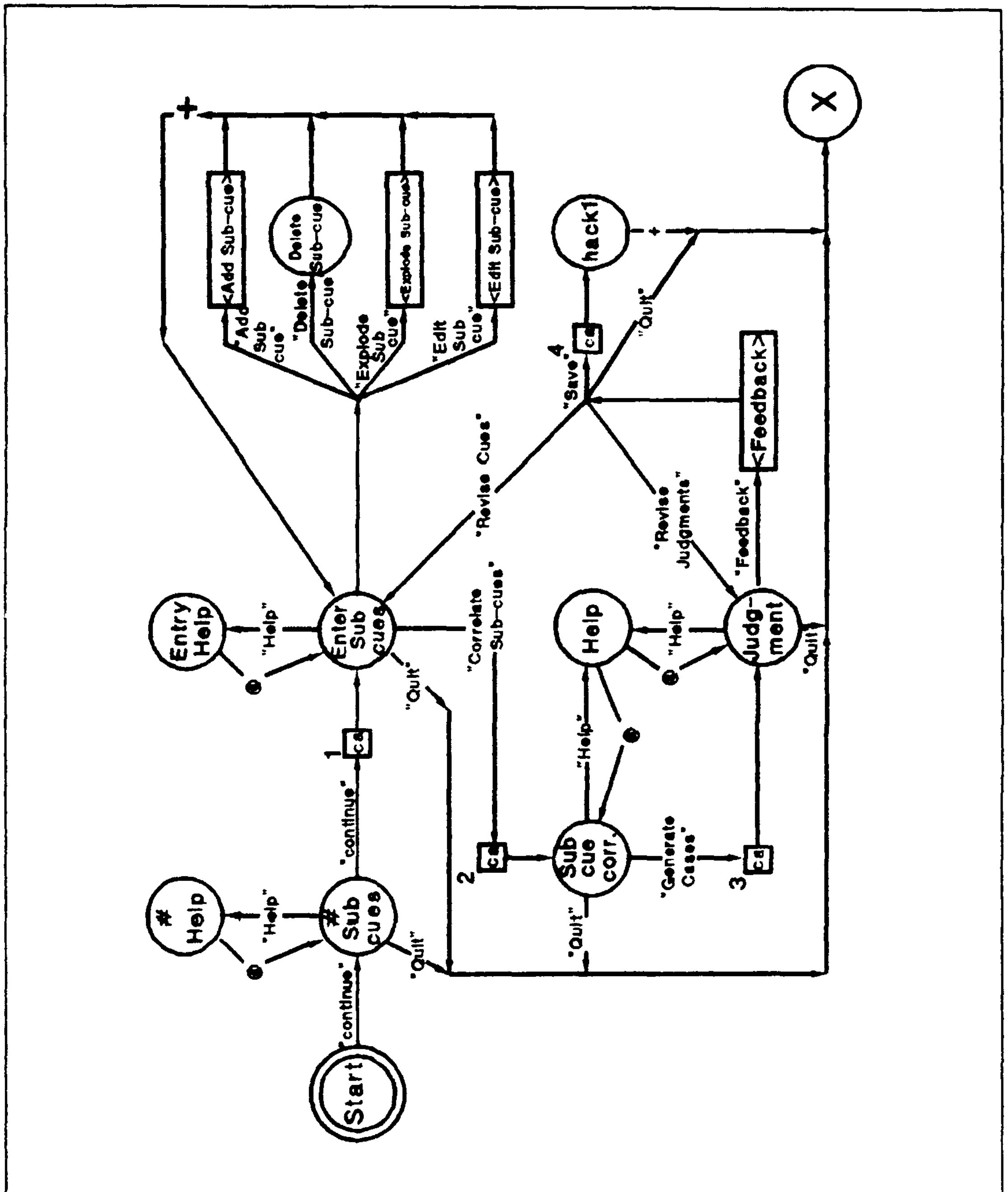


Figure 19. State transition diagram for the explode module.

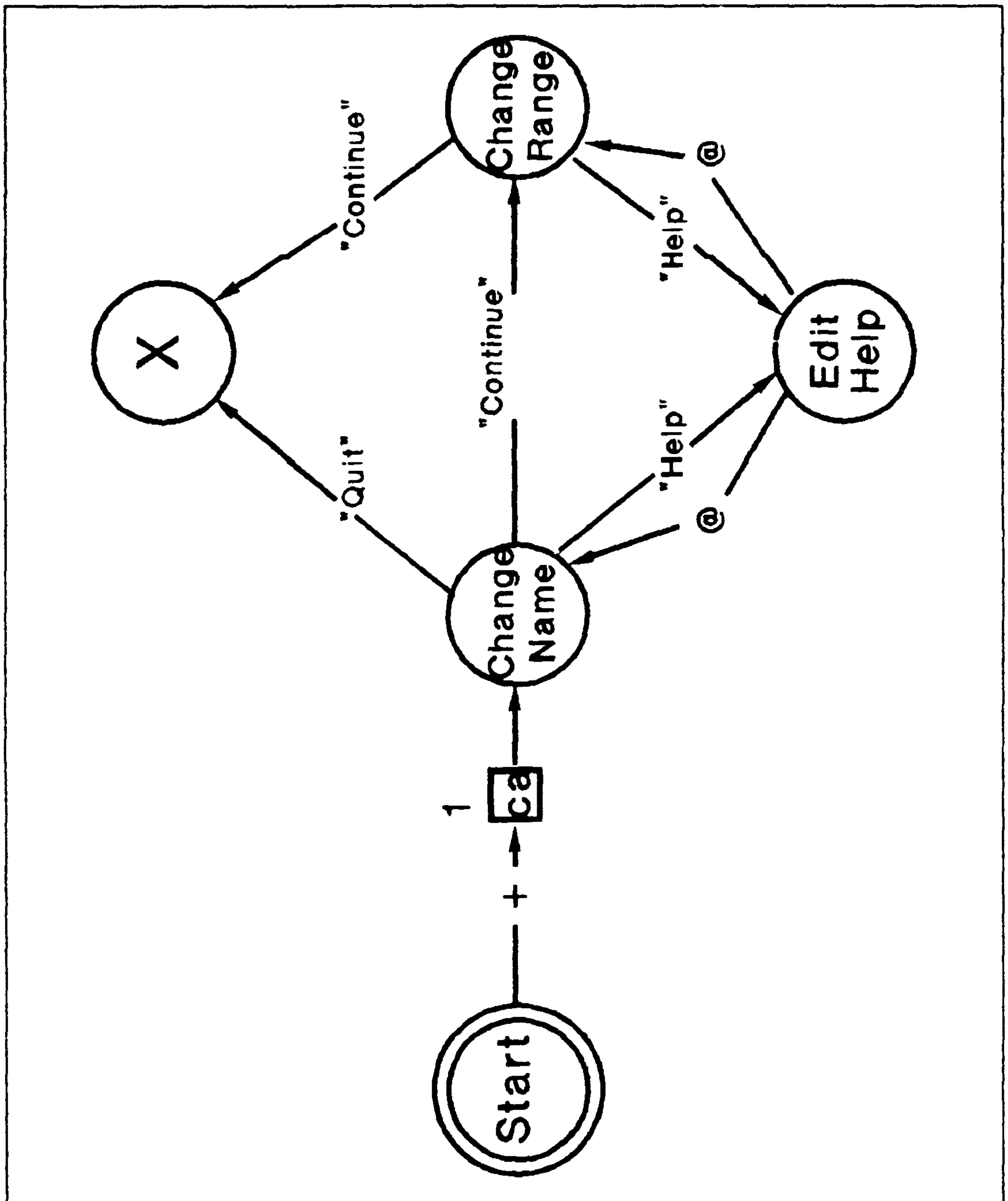


Figure 20. State transition diagram for the edit module.

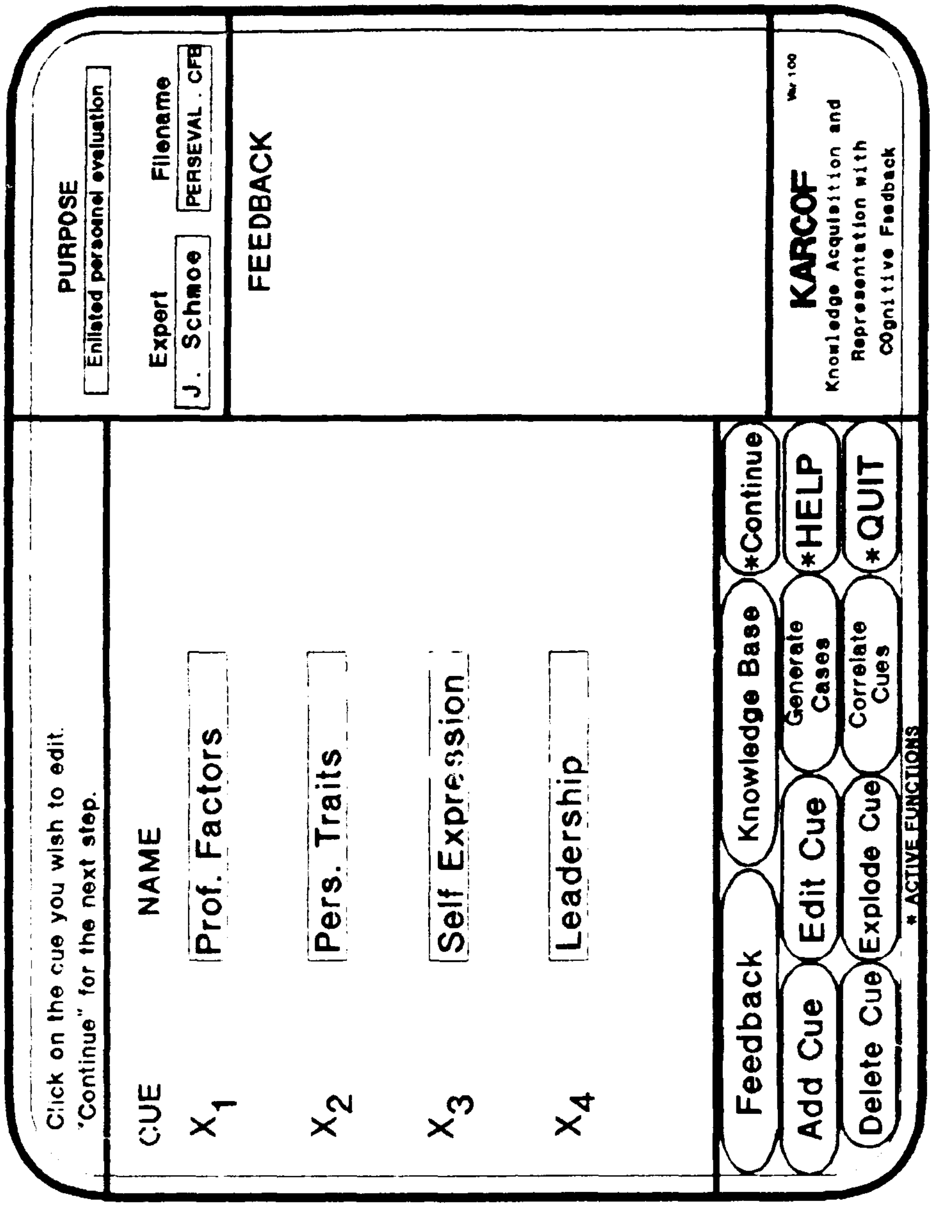


Figure 21. Screen for changing the name of a cue.

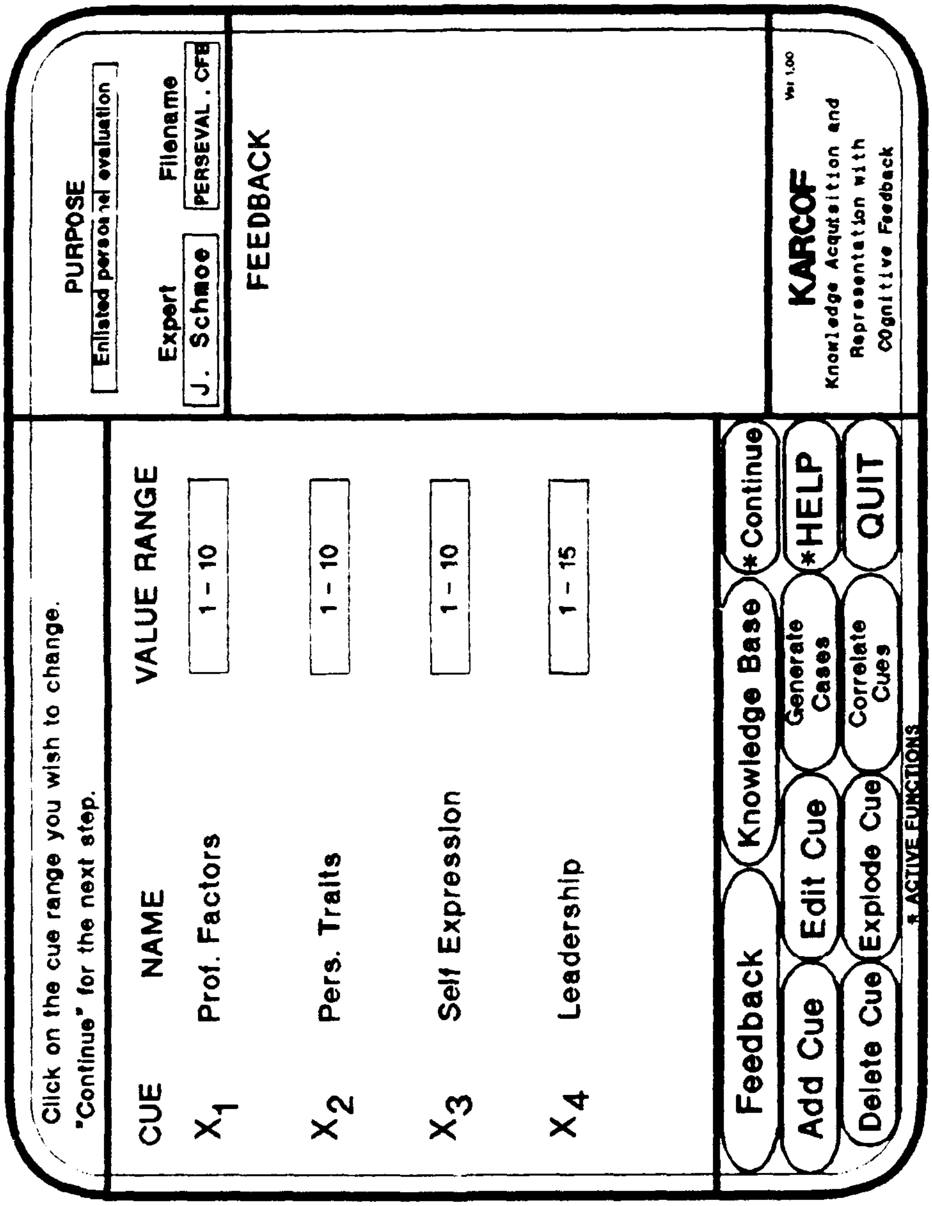


Figure 22. Screen for changing the value range of a cue.

feedback. This is expected to eliminate the negative and counterproductive aspects of delayed feedback. If feedback is not immediate, its effectiveness will be diminished, as many factors that went into the decision making process may be forgotten or attenuated. The expert's mind is invariably preoccupied with something else if the feedback arrives late. (Wickens, 1984)

The ability of KARCOF to define a hierarchical model is more reflective of actual decision making processes. Decisions may not always be based upon a single layer of cues. Many situations may be dependent on several decisions, some of which impinge on others in a hierarchical manner.

A program such as KARCOF, when finally implemented, should allow the domain expert to assume many functions currently performed by a knowledge engineer. Table 11 summarizes the steps in KARCOF. This will result in enhanced quality of the knowledge base and one that is more reflective of the domain expert's mental model. The knowledge engineer's function would be redefined to that of a facilitator with in-depth technical knowledge of the tool. He or she would handle some refinement in the form of the knowledge base and whatever special situations in coding or coordination that invariably arise. The expert would no longer be confined by the focus of the knowledge engineer on implementation and representation problems.

TABLE 11. SUMMARY OF THE STEPS IN KARCOF.

Step	Action	Figure
1	Setup. Name the policy, expert, and file.	3
2	Specify the number of cues.	4
3	Specify nature, name, and range of cues.	5
4	Specify cue correlations.	7
5	Judge a set of generated cases.	8
6	Specify form of feedback.	10
7	View feedback.	11
8	Validate knowledge base.	13
9	Save knowledge base or iterate between steps 2-8: Add Cue, Explode Cue, Delete Cue, or Edit Cue.	14 16 17 18 21 22

V. CONCLUSION

A. SUMMARY

A major goal of the knowledge acquisition process is to determine what a domain expert knows and uses to solve problems. An understanding of the different types of knowledge that exist is an important first step toward this goal. Chapter II surveyed four types of knowledge: declarative, procedural, semantic, and episodic. Of particular interest, when considering expertise, are the procedural and semantic types of knowledge.

Procedural knowledge presents especially difficult problems for the knowledge acquisition process. This is due to its highly compiled and automated nature. Knowledge that was once declarative is combined with other types of knowledge; then refined, tuned, strengthened, and integrated into the expert's overall knowledge base. The facts, rules, and concepts that comprise procedural knowledge are represented in a more abstract and solution-oriented manner. These cognitive changes have a beneficial impact on a person's ability to use the information, but negatively impact an individual's ability to consciously access the information. The inability of experts to verbalize about their cognitive processes is well documented.

Many different techniques have been developed in an effort to capture expert knowledge. Chapter II presented 14 of these, half which can be considered "direct" techniques, and half which can be considered "indirect". Direct techniques rely on some form of verbalization or introspection by the individual. These methods are only effective at illuminating the declarative aspects of knowledge and they work poorly with procedural knowledge. The "indirect" techniques rely less on verbalization and more on methods that attempt to describe the underlying forms or organization (mental models) of the knowledge. A problem with the indirect techniques is that all assume, *a priori*, some underlying form. These methods can only succeed to the extent that the expert's underlying mental model parallels the assumed forms.

Many cognitive scientists hypothesize that human thinking is a multi-representational system (Rouse and Morris, 1986). Each aspect of the represented environment is mapped into the representation best suited to a particular use or domain. It is apparent that since the indirect techniques of Chapter II assume an underlying form, they are capable of representing but *one* model.

Chapter III presented Brunswik's probabilistic functionalism theory that explains how pieces of information or cues, may be used by a person in a decision making process. The cues are viewed as less than dependable, and they exist within a probabilistic environment. Depending on the task at hand, the cues may be combined and used in a multitude of ways. The theory is operationalized through use of the lens model that graphically represents how an expert may combine cues when an inference is made about some object or event in the future. Each different arrangement of cues may be considered a representation of a different mental model. Therefore, the lens model can represent far more than the single models of Chapter II's indirect techniques.

The lens model also provides a convenient way to combine an expert's cognitive system (mental model representation) with representations of the environment and the task, into a single overarching model. Statistical analysis techniques allow for comparisons and predictions between each of the systems depicted in the lens model. Measurements of the decision maker's achievement, predictability, and the match with the environment are possible.

When the lens model is employed in the single system paradigm, the expert's cognitive system, or mental model, is represented. Cognitive feedback, the return of some aspects of the decision output to the decision maker, can be used to capture and even change the expert's working mental model. Chapter III described in detail the types of cognitive feedback that can be returned and a summary of studies that have examined the effectiveness of each.

The utility of cognitive feedback is well documented, as are comparisons with the other type of decision feedback, outcome feedback. Outcome feedback may have a detrimental impact on cognitive processes whereas cognitive feedback has shown a positive and constructive effect on cognitive processes.

Cognitive feedback has been successfully used in a variety of fields that involve training and learning. It has also been used to resolve conflicts among individuals and organizations by providing each party with a view of what they value most. Computers have been used with these examples to capture the individual's policy or knowledge, and display feedback in graphical form.

B. FINAL REMARKS

The above summary addressed the first and second objectives of the thesis. The first was to discover what gaps exist in current knowledge acquisition methodologies. A survey of the techniques revealed that there appears to be no reliable technique for capturing an expert's procedural knowledge or mental model in general.

The second objective was to determine whether cognitive feedback is appropriate as a tool for filling the gaps identified in the current techniques. This study has found ample evidence that cognitive feedback and the lens model may be effective in capturing a variety of mental models, and with that, an expert's procedural knowledge.

The third objective of this study was to determine how, in an operational sense, cognitive feedback could be employed to extract an individual's knowledge. Chapter IV addresses this goal through the description of a proposed automated knowledge acquisition tool that uses cognitive feedback. The program can be used directly by an expert without the intervention of a knowledge engineer. The expert furnishes the program with the number of cues and cue intercorrelations used in a decision making process. Based on this, and a standard error suitable for statistical stability specified by a supervising knowledge engineer, the program generates a number of cases. The expert then enters a numerical judgment for each case. The program calculates the necessary statistical measures and presents cognitive feedback in graphical form. If the expert is not satisfied with the feedback he or she may revise previous judgments or alter the parameters of the problem. This may continue in an iterative manner until the expert is convinced that the computer has captured a workable representation of their knowledge.

C. APPLICABLE TASKS

Cognitive feedback appears to have broad applicability irrespective of domain or problem solving technique. As shown in Chapter II, many knowledge acquisition tools form a specific link between a problem solving method and an application task category. Cognitive feedback however, due to its general nature, may work with a number of problem solving techniques. It is also particularly well suited to inference tasks since the lens model, with its proximal-distal variable separation, is the very definition of an inference task.

Cognitive feedback could be used in personnel selection boards, or in tasks that rate the performance of individuals. It could be used for complex learning situations such as nuclear reactor operation or anti-submarine operations. Many other possible applications present themselves:

--Economic forecasting;
--Practice in battlefield situation assessment;
--Conflict resolution;
--Auditing;
--Security risk assessment;
--Law: Case evaluation, litigation risk;
--Medical diagnosis;
--Hardware diagnosis.

D. AREAS FOR FURTHER RESEARCH

Coding of the automated knowledge acquisition program is the subject of a follow-on study to this thesis. Once the prototype has been developed the logical next step involves empirical studies of its effectiveness in capturing procedural knowledge. Empirical studies should also be conducted to determine what domains or tasks, within the Department of Defense, would benefit from the development of this tool.

The primary goal of a knowledge acquisition tool, whether automated or manual, is to represent knowledge for ultimate inclusion into a working expert system. Follow-on research should take the knowledge captured by cognitive feedback, as linear equations, and combine it with an effective inference mechanism in an expert system. An alternative would be to transform this captured knowledge into a form that can be used by an existing inference engine in an expert system.

Additional areas of research should investigate the intricacies of dealing with uncertainty. There are two aspects to this problem. First, considering the emphasis that probabilistic functionalism places upon the probabilistic nature of the environment, researchers have yet to devise a means of representing uncertainty with the lens model. R_e and R_s represent point estimates of overall error but a means of representing error bands around parameters is lacking. Some parameters that should be treated in this way are the estimates of ecological validities, utilization coefficients, and r_a. Second, methods of handling uncertainty must be worked into the expert system that uses knowledge, in linear equations, captured by cognitive feedback.

LIST OF REFERENCES

Adelman, L., "The Influence of Formal, Substantive and Contextual Task Properties on the Relative Effectiveness of Different Forms of Feedback in Multiple Cue Probability Learning Tasks", *Organizational Behavior and Human Performance*, v. 27, pp. 423-442, 1981

Bainbridge, L., "Verbal Reports as Evidence of the Process Operators Knowledge", *International Journal of Man-machine Studies*, v. 11, pp. 411-436, 1979

Balke, W.M., Hammond, K.R., and Meyer, G.D., "An Alternate Approach to Labor-Management Relations", *Administrative Science Quarterly*, v. 18, pp. 311-327, 1973

Balzer, W.K., Doherty, M.E., and O'Connor Jr.,R., "Effects of Cognitive Feedback on Performance", *Psychological Bulletin*, v. 3, pp. 410-433, 1989

Berry, D.C., "The Problem of Implicit Knowledge", *Expert Systems*, v. 4, pp. 144-151, 1987

Berry, D.C., and Broadbent, D.E., "On the Relationship Between Task Performance and Associated Verbalisable Knowledge", *Quarterly Journal of Experimental Psychology*, v. 36A, pp. 209-231, 1984

Boose, J.H., "A Knowledge Acquisition Program for Expert Systems Based on Personal Construct Theory", *International Journal of Man-Machine Studies*, v. 23, pp. 495-525, 1985

Boose, J.H., *Expertise Transfer for Expert System Design*, pp. 1-50, Elsevier Science Publishers, 1986

Brehmer, B., "Preliminaries to a Psychology of Inference", *Scandinavian Journal of Psychology*, v. 20, pp. 193-210, 1979

Brehmer, B., "In One Word: Not From Experience", *Acta Psychologica*, v. 45, pp. 223-241, 1980

Brehmer, B., and Svensson, C., "Learning to Use Function Rules in Inference Tasks", *Scandinavian Journal of Psychology*, v. 17, pp. 313-319, 1976

Breuker, J.A., and Weilinga, B.J., "Use of Models in the Interpretation of Verbal Data", in Kidd, A.(ed.), *Knowledge Elicitation for Expert Systems: A Practical Handbook*, pp. 17-44, Plenum Press, 1987

Broadbent, D.E., Fitzgerald, P., and Broadbent, M.H.P., "Implicit and Explicit Knowledge in the Control of Complex Systems", *British Journal of Psychology*, v. 77, pp. 33-50, 1986

Brunswik, E., "Organismic Achievement and Environmental Probability", *Psychological Review*, v. 50, pp. 255-272, 1943

Brunswik, E., "The Conceptual Framework of Psychology", in Neurath, O., Carnap, R., and Morris, C. (eds.), *International Encyclopedia of Unified Science*, v. 1, nos. 6-10, pp. 655-760, University of Chicago Press, 1955

Burton, A.M., Shadbolt, N.R., Hedgecock, A.P., and Rugg, G., "A Formal Evaluation of Knowledge Elicitation Techniques for Expert Systems: Domain 1", *Research and Development in Expert Systems IV*, pp. 136-145, Cambridge University Press, 1988

Clancey, W.J., "Heuristic Classification", in Kowalik, J.S. (ed.), *Knowledge Based Problem Solving*, pp. 1-67, Prentice Hall, 1986

Clover, W., *Cognitive Feedback in the Selection Interview: Applying Social Judgment Theory in the Field.* Unpublished doctoral dissertation, Bowling Green State University, Bowling Green, OH, 1979

Cook, R.L., *A Study of Interactive Judgment Analysis and the Representation of Weights in Judgment Policies*, Unpublished doctoral dissertation, University of Colorado, 1976

Davis, R., "Interactive Transfer of Expertise: Acquisition of New Inference Rules", *Artificial Intelligence*, v. 12, pp. 121-157, 1979

Deane, D.H., Hammond, K.R., and Summers, D.A., "Acquisition and Application of Knowledge in Complex Inference Tasks", *Journal of Experimental Psychology*, v. 92, pp. 20-26, 1972

Diederich, J., Ruhmann, I., and May, M., "KRITON: A Knowledge Acquisition Tool for Expert Systems", *International Journal of Man-Machine Studies*, v. 26, pp. 29-40, 1987

Doherty, M.E., and Balzer, W.K., "Cognitive Feedback", in Brehmer, B., and Joyce, C.R.B. (eds.), *Human Judgment: The SJT View*, pp. 163-197, Elsevier Science Publishers, 1988

Einhorn, H., Kleinmuntz, D., and Kleinmuntz, B., "Linear Regression and Process Tracing Models of Judgment", *Psychological Review*, v. 86, pp. 465-485, 1979

Eshelman, L., "MOLE: A Knowledge Acquisition Tool that Buries Certainty Factors", *International Journal of Man-Machine Studies*, v. 29, pp. 563-577, 1988

Fero, D.D., *A Lens Model Analysis of the Effects of the Amount of Information and Mechanical Decision Making Aid on Clinical Judgment and Confidence.* Unpublished doctoral dissertation, Bowling Green State University, OH, 1975

Fischoff, B., "Perceived Informativeness of Facts", *Journal of Experimental Psychology: Human Perception and Performance*, v. 3, pp. 349-358, 1977

Flack, J.E., and Summers, D.A., "Computer Aided Conflict Resolution in Water Resource Planning: An Illustration", *Water Resources Research*, v. 7, pp. 1410-1414, 1971

Gaines, B., "An Overview of Knowledge Acquisition and Transfer", *International Journal of Man-Machine Studies*, v. 26, pp. 453-472, 1987

Galbraith, J.T., *Training Assessment Center Assessors: Applying Principles of Human Judgment.* Unpublished doctoral dissertation, Bowling Green State University, Bowling Green, OH, 1984

Gammack, J.G., "Different Techniques and Different Aspects on Declarative Knowledge", in Kidd, A.L. (ed.), *Knowledge Acquisition for Expert Systems: A Practical Handbook*, pp. 137-163, Plenum Press, 1987

Gammack, J., and Young, R., "Psychological Techniques for Eliciting Expert Knowledge", in Bramer, M. (ed.), *Research and Development in Expert Systems*, pp. 105-112, Cambridge University Press, 1985

Garg-Janardan, C., and Salvendy, G., "A Conceptual Framework for Knowledge Elicitation", *International Journal of Man-Machine Studies*, v. 26, pp. 521-531, 1987

Hammond, K.R., and Boyle, J.R., "Quasi-Rationality, Quarrels, and New Conceptions of Feedback", *Bulletin of the British Psychological Society*, v. 24, pp. 103-113, 1971

Hammond, K.R., Hursch, C.J., and Todd, F.J., "Analyzing the Components of Clinical Inference", *Psychological Review*, v. 71, pp. 438-456, 1964

Hammond, K.R., Stewart, T.R., Brehmer, B., and Steinmann, D.O., "Social Judgment Theory", in Arkes, H.R., and Hammond, K.R. (eds.), *Judgment and Decision Making*, pp. 56-76, Cambridge University Press, 1986

Hammond, K.R., and Summers, D.A., "Cognitive Control", *Psychological Review*, v. 79, pp. 58-67, 1972

Harmon, P., and King, D., *Expert Systems: Artificial Intelligence in Business*, pp. 22-48, John Wiley and Sons, 1985

Hart, A., *Knowledge Acquisition for Expert Systems*, pp. 133-152, McGraw-Hill, 1986

Hawkins, D., "An Analysis of Expert Thinking", *International Journal of Man-Machine Studies*, v. 18, pp. 1-47, 1983

Hayes-Roth, F., Klahr, P., and Mostow, D.J., "Knowledge Acquisition, Knowledge Programming, and Knowledge Refinement", *Expert Systems: Techniques, Tools, and Applications*, pp. 310-349, Addison-Wesley, 1986

Hoffman, P.J., "The Paramorphic Representation of Clinical Judgment", *Psychological Bulletin*, v. 57, pp. 116-131, 1960

Hoffman, P.J., Earle, T.C., and Slovic, P., "Multidimensional Functional Learning (MFL) and Some New Conceptions of Feedback", *Organizational Behavior and Human Performance*, v. 27, pp. 75-102, 1981

Hoffman, R.R., "The Problem of Extracting the Knowledge of Experts from the Perspective of Experimental Psychology", *AI Magazine*, v. 8, pp. 53-67, 1987

Hogarth, R.M., "Process Tracing in Clinical Judgment", *Behavioral Science*, v. 19, pp. 298-313, 1974

Hogarth, R.M., "Beyond Discrete Biases: Functional and Dysfunctional Aspects of Judgmental Heuristics", *Psychological Bulletin*, v. 90, pp. 197-217, 1981

Hursch, C.J., Hammond, K.R., and Hursch, J.L., "Some Methodological Considerations in Multiple-Cue Probability Studies", *Psychological Review*, v. 71, pp. 42-60, 1964

Jacoby, J., Mazursky, D., Troutman, T., and Kuss, A., "WhenFeedback is Ignored: Disutility of Outcome Feedback", *Journal of Applied Psychology*, v. 69, pp. 531-545, 1984

Johnson, P.E., "What Kind of Expert Should a System Be?", *Journal of Medicine and Philosophy*, v. 8, pp. 77-97, 1983

Kahn, G., Nowlan, S., and McDermott, J., "MORE: An Intelligent Knowledge Acquisition Tool", *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pp. 581-584, 1985

Kidd, A.L. (ed.), *Knowledge Acquisition for Expert Systems: A Practical Handbook*, pp. 1-16, Plenum Press, 1987

Kitto, C.M., and Boose, J.H., "Selecting Knowledge Acquisition Tools and Strategies Based on Application Characteristics", *International Journal of Man-Machine Studies*, v. 31, pp. 149-160, 1989

Klayman, J., "Learning From Feedback in Probabilistic Environments", *Acta Psychologica*, v. 56, pp. 81-92, 1984

Klinker, G., Boyd, C., Genetet, S., and McDermott, J., "A KNACK for Knowledge Acquisition", *Proceedings of the AAAI-87*, pp. 488-493, 1987

Libby, R., *Accounting and Human Information Processing: Theory and Applications*, pp. 18-49, Prentice Hall, 1981

Lindell, M.K., "Cognitive and Outcome Feedback in Multiple Cue Probability Learning Tasks", *Journal of Experimental Psychology: Human Learning and Memory*, v. 2, pp. 739-745, 1976

Marcus, M., "Taking Backtracking With a Grain of SALT", *International Journal of Man-Machine Studies*, v. 26, pp. 383-398, 1987

McGraw, K.L., and Riner, A., "Task Analysis: Structuring the Knowledge Acquisition Process", *TI Technical Journal*, pp. 16-21, November-December 1987

McKeithen, K.B., Reitman, J.S., Rueter, H.H., and Hirtle, S.C., "Knowledge Organization and Skill Differences in Computer Programmers", *Cognitive Psychology*, v. 13, pp. 307-325, 1981

Milter, R.G., and Rohrbaugh, J., "Judgment Analysis and Decision Conferencing for Administrative Review: A Case Study of Innovative Policy Making in Government", *Advances in Information Processing in Organizations*, v. 3, pp. 245-262, 1988

Newton, J.R., "Judgment and Feedback in a Quasi-Clinical Situation", *Journal of Personality and Social Psychology*, v. 1, pp. 336-342, 1965

Null, C.H., "Design Considerations for Multidimensional Scaling", *Behavior Research Methods and Instrumentation*, v. 12, pp. 274-280, 1980

Nystedt, L., and Magnusson, D., "Cue Relevance and Feedback in a Clinical Prediction Task", *Organizational Behavior and Human Performance*, v. 9, pp. 100-109, 1973

Olson, J.R., and Rueter, H.H., "Extracting Expertise from Experts: Methods for Knowledge Acquisition", *Expert Systems*, pp. 152-168, August 1987

Reitman, J.S., "Skilled Perception in Go: Deducing Memory Structures from Interresponse Times", *Cognitive Psychology*, v. 8, pp. 336-356, 1976

Reitman, J.S., and Rueter, H.H., "Organization Revealed by Recall Orders and Confirmed by Pauses", *Cognitive Psychology*, v. 12, pp. 554-581, 1980

Richter, M.M., "Some Abstract Problems in Knowledge Representation", *Expert Judgment and Expert Systems*, pp. 17-26, Springer-Verlag, 1986

Riesbeck, C.K., "Knowledge Reorganization and Reasoning Style", *Developments in Expert Systems*, pp. 159-175, Academic Press, 1984

Rouse, W.B., and Morris, N.M., "On Looking Into the Black Box: Prospects and Limits in the Search for Mental Models", *Psychological Bulletin*, v. 100, pp. 349-363, 1986

Schmitt, N., Coyle, B.W., and King, L., "Feedback and Task Predictability as Determinants of Performance in Multiple Cue Probability Learning Tasks", *Organizational Behavior and Human Performance*, v. 16, pp. 388-402, 1976

Schmitt, N., Coyle, B.W., and Saari, B.B., "Types of Task Information Feedback in Multiple Cue Probability Learning" *Organizational Behavior and Human Performance*, v. 18, pp. 316-328, 1977

Schmitt, N., Levine, R., "Statistical and Subjective Weights: Some Problems and Proposals", *Organizational Behavior and Human Performance*, v. 20, pp. 15-30, 1977

Shaw, M.L.G., and Gaines, B.R., "KITTEN: Knowledge Initiation and Transfer Tools for Experts and Novices", *International Journal of Man-Machine Studies*, v. 27, pp. 251-280, 1987a

Shaw, M.L.G., and Gaines, B.R., "An Interactive Knowledge Acquisition Technique Using Personal Construct Psychology", in Kidd, A.L.(ed.), *Knowledge Acquisition for Expert Systems: A Practical Handbook*, pp. 109-136, Plenum Press, 1987b

Shaw, M.L.G., "PLANET: Some Experience in Creating an Integrated System for Repertory Grid Applications on a Microcomputer", *International Journal of Man-Machine Studies*, v. 17, pp. 345-360, 1982

Stang, S.W., "An Interactive Judgment Analysis of Job Worth", Unpublished doctoral dissertation, Bowling Green State University, Bowling Green, OH, 1985

Steinmann, D.O., "Transfer of Lens Model Training", *Organizational Behavior and Human Performance*, v. 12, pp.1-16, 1974

Stewart, T.R., "Judgment Analysis: Procedures", in Brehmer, B., and Joyce, C.R.B. (eds.), *Human Judgment: The SJT View*, pp. 41-74, Elsevier Science Publishers, 1988

Todd, F.J., and Hammond, K.R., "Differential Effects in Two Multiple Cue Probability Learning Tasks", *Behavioral Science, v. 10, pp. 429-435, 1965*

Tolman, E.C., and Brunswik, E., "The Organism and the Causal Texture of the Environment", *Psychological Review*, v. 42, pp. 43-77, 1935

Tucker, L.R., "A Suggested Alternative Formultion in the Developments by Hursch, Hammond, and Hursch, and by Hammond, Hursch, and Todd", *Psychological Review*, v. 71, pp. 528-530, 1964

von Winterfeldt, D., "Expert Systems and Behavioral Decision Research", *Decision Support Systems*, v. 4, pp. 461-471, 1988

Wasserman, A.I., "Extending State Transition Diagrams for the Specification of Human-Computer Interaction", *IEEE Transactions in Software Engineering*, v. 11, pp. 699-713, 1985

Waterman, D.A., *A Guide to Expert Systems*, pp. 49-60, Addison-Wesley, 1986

Wickens, C.D., *Engineering Psychology and Human Performance*, pp 107-109, Charles E. Merrill 1984

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Professor Kishore Sengupta, Code ASSe Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5004	1
4.	Professor Tung X. Bui, Code ASBd Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943-5004	2
5.	Lt. Charles Allen Patterson 240 Remington Loop Danville, California 94526-3732	2