AI Magazine Volume 8 Number 2 (1987) (© AAAI)

The Problem of Extracting the Knowledge of Experts from the Perspective of Experimental Psychology

Robert R. Hoffman

The first step in the development of an expert system is the extraction and characterization of the knowledge and skills of an expert. This step is widely regarded as the major bottleneck in the system development process To assist knowledge engineers and others who might be interested in the development of an expert system, I offer (1) a working classification of methods for extracting an expert's knowledge, (2) some ideas about the types of data that the methods yield, and (3) a set of criteria by which the methods can be compared relative to the needs of the system developer The discussion highlights certain issues, including the contrast between the empirical approach taken by experimental psychologists and the formalism-oriented approach that is generally taken by cognitive scientists

For perceptual and conceptual problems requiring the skills of an expert, expertise is rare, the expert's knowledge is extremely detailed and interconnected, and our scientific understanding of the expert's perceptual and conceptual processes is limited. Research on the skills of experts in any domain affords an excellent opportunity for both basic and practical experimentation. My investigations fall on the experimental psychology side of expert system engineering, specifically the problem of generating methods for extracting the knowledge of experts. I do not review the relevant literature on the cognition of experts.[1] I want to share a few ideas about research methods that I found worthwhile as I worked with expert interpreters of aerial photographs and other remotely sensed data (Hoffman 1984) and on a project involving expert planners of airlift operations (Hoffman 1986). These ideas should be useful to knowledge engineers and others who might be interested in developing an expert system.

The Bottleneck

In generating expert systems, one must begin by characterizing the knowledge of an expert. As far as I can tell from the literature on expert systems, getting this characterization is the significant bottleneck in the system development process. For example, from their experience in the development of PROSPECTOR, Duda and Gashnig (1981) concluded that "... something must be done to shorten the time needed to interview experts and represent their special knowledge . . . [It] may take several months of the expert's time and even more of the system builder's" (p. 264). Three years later, Duda and Shortliffe (1983) echoed this lament: "The identification and encoding of knowledge is one of the most complex and arduous tasks encountered in the construction of an expert system" (p. 265).

Some common phrases that occur in the literature are "knowledge acquisition is the time-critical component" (Freiling et al. 1985), and "extracting and articulating the knowledge is the most important phase" (Gevarter 1984), and "the process of extracting knowledge can be laborious" (Quinlan 1984). I encountered passages of this type in every major review article or textbook on expert systems that has come to my attention (for example, Bramer 1982; Denning 1986; Hayes-Roth, Waterman, and Lenat 1983; Mittal and Dym 1985; Raulefs 1985; Weiss and Kulikowski 1984). Such articles dutifully assert that "the first step is to get the knowledge" (Freiling et al. 1985, p. 152). What follows, however, is typically a discussion of abstract inference strategies, details of Lisp code, and descriptions of system architecture rather than an answer to the question of exactly how to get the knowledge.

Some papers have titles that explicitly suggest they deal with the nuts and bolts of how to extract an expert's knowledge (for example, "Acquisition of Knowledge from Domain Experts," Friedland 1981; see also Nii and Aiello 1979; Politakis and Weiss 1984). However, such papers deal mostly with the

METHOD CATEGORY	DESCRIPTION
METHOD OF "FAMILIAR" TASKS	Analysis of the tasks that the expert usually performs.
STRUCTURED AND UNSTRUCTURED INTERVIEWS	The expert is queried with regard to knowledge of facts and procedures.
LIMITED INFORMATION TASKS	A familiar task is performed, but the expert is not given certain information that is typically available.
CONSTRAINED PROCESSING TASKS	A familiar task is performed, but the expert must do so under time or other constraints.
METHOD OF "TOUGH CASES"	Analysis of a familiar task that is conducted for a set of data that presents a "tough case" for the expert.

Table 1. Types of Methods That Can Be Used to Extract the Knowledge of an Expert.

representation of knowledge. In short, apparently little or no systematic research has been conducted on the question of how to elicit an expert's knowledge and inference strategies (Duda and Shortliffe 1983; Hartley 1981).[2] How can one find good tasks for extracting the knowledge of an expert? How can various tasks be compared? Can tasks be tailored to extract specific subdomains of knowledge within the expert's broader domains of knowledge? My research has addressed these questions.

Methods for extracting expert knowledge seem to fall neatly into a handful of categories, as shown in table 1. One obvious methodological category involves observing the performance of the expert at the kinds of tasks the expert is familiar with or usually engages in. A second category of methods is the interview. Artificial tasks can be devised that depart from what the expert usually does by limiting the information that is available to the expert or by constraining the problem the expert is to work on. Another type of method involves studying the expert's performance on the "tough cases" that sometimes occur. The categorization shown in table 1 forms the topical organization of this paper. Following a brief discussion of each of these method types is an examination of some ways in which the data they yield can be analyzed and the various methods compared.

The Method of Familiar Tasks

The method of familiar tasks involves studying the expert while he or she is engaged in the kinds of tasks that are usually or typically engaged in. Looking across a set of experts' specific tactics and procedures, one should see commonalities in terms of goals, the information the experts like to have available, and the data or records that are produced (Mittal and Dym 1985). In a number of reviews (for example, Duda and Shortliffe 1983; Stefik et al. 1982), the various tasks that experts engage in have been analyzed and categorized into basic types, such as diagnosing (interpretation of data), planning, designing, and explaining.

Psychologically, the tasks that an expert typically performs involve at least the following: (1) the analysis of complex stimuli into relevant features or cues based on a process psychologists call "perceptual learning," (2) the analysis of conceptual categories in terms of the relevant features (the perception of similarities and differences), (3) the analysis of the features and the categories in terms of relevant underlying causal laws (involving "concept-formation processes"), and (4) abilities to infer and test hypotheses.

Although these (and probably other) psychological factors are involved, the products that result from familiar tasks bear on these tasks and might not actually be very informative about the expert's reasoning. For example, the standard method of aerial-photo interpretation *(terrain analysis)* provides lots of information about the land that appears in an aerial photo but says precious little about how the expert arrived at the description (cf. Mintzer and Messmore 1984; Way 1978). Such is also probably true for the kinds of diagnostic notations made by radiologists when they interpret X rays (Feltovich 1981). Nevertheless, an analysis of familiar tasks can be very beneficial because it can give the knowledge engineer a feel for the kinds of knowledge and skill involved in the domain.

An analysis of familiar tasks (including an analysis of available texts and technical manuals) can be used to generate a "first pass" at a data base. What the expert knows is represented as a categorized listing of statements cast in some sort of formal language (such as propositions) using terms and categories that are meaningful and related to the domain at hand (Freiling et al. 1985). Such propositions can express observation statements or facts as well as implications or potential if-then rules of inference.[3] Table 2 presents example excerpts from the data base of the aerial-photo-interpretation project.

The Unstructured Interview

As far as I can tell, the development of most existing expert systems started with unstructured interviews of the

ROCK FORMS #3--DOME
Raised circular, linear or ellipsoid rock
Can be small, compound, or clustered
Can have radiating fractures
Can be salt, gypsum, or intrusive bedrock
Radial drainage pattern, annular pattern at base of the slope

ROCK TYPE #1--FLAT SHALE
Gently rolling, irregular plain
Symmetrical finger ridges
Branching rounded hills with saddle ridges
Scalloped hill bases
V- and U-shaped stream gullies
Uniform slope gradients imply homogeneous rock
Compound slope gradients imply thick bedding
Escarpments, very sharp ridges, steep slopes, and steep pinnacles imply sandy soils

HUMID CLIMATE
Implies rounded hills
Implies fine dendritic drainage pattern

ARID CLIMATE
Implies steep, rounded hills and ridges with asymmetrical slopes
Implies intermittent drainage
Implies barren land or shrub land
Implies light or mottled soil tones

FLUVIAL LANDFORMS #17--PLAYAS
Dry, very low relief lakebeds in arid regions
Can include beach ridges
Few drainage features
Irrigation and intense cultivation
Scrabbled surface implies alkaline deposits

SOIL TYPE #2--SILT
Light tones
U-shaped stream gullies

DRAINAGE PATTERNS #5--THERMOKARST
Stream gullies form polygons and hexagons, linked by meandering streams
Implies permafrost

GULLY SHAPES #2--U-SHAPED GULLIES
Moderately steep slopes with curved channel bottom
Implies loess soil type

AGRICULTURE TYPE #4--ORCHARDS
Vegetation in lattice pattern or repeated uniform rows
Porous, well-drained soils
Level terrain plus trees in a rectangular pattern imply nuts or citrus
Rolling, uneven terrain plus trees in a contoured pattern imply fruits

Table 2. An Example Set of Excerpts from the Aerial-Photo-Interpretation Data Base (Hoffman 1984). Within each category and subcategory there can appear potential if-then rules ("X implies Y"), as well as facts (observation statements). Stream gully shapes refer to their cross sections.

experts. Indeed, most system developers have apparently relied exclusively on the unstructured interview method. Some developers have even apparently taken it for granted that an unstructured interview is the only way to extract expert knowledge (for example, Weiss and Kulikowski 1984, p. 105).

In an unstructured interview, the knowledge engineer asks more-or-less spontaneous questions of the expert while the expert is performing (or talking about) a familiar task (see Freiling et al. 1985). For instance, the interviewer might ask the expert a question such as "How do you know that?" whenever the expert seems to tap into knowledge or make an inference, or "Do you want a cup of coffee?" whenever they get tired.

Presumably, the knowledge engineer's prior analysis of the familiar tasks has turned the engineer from a novice into an apprentice (or even a master). Thus, the engineer is trained in what to look for during the interview. Should the training or preparation proceed so far as to turn the knowledge engineer into an expert? I do not know if there can be any principled answer to this question. It has been claimed (Davis 1986) that the novice interviewer might be better able to ask questions about ideas or procedures which an expert tends to leave implicit or take for granted (see also Hartley 1981).

Table 3 presents an example excerpt from an unstructured interview conducted during the airlift-planning project (Hoffman 1986). This excerpt is prototypical in that there are inquiries from the interviewer, speech pauses and hesitations, ungrammatical segments, and unintelligible segments. The expert's monologues in this excerpt are brief. Occasionally, they can be much longer.

Knowledge engineers sometimes make an audiotape of the expert's ruminations; this recording is called a *verbal protocol* (Ericcson and Simon 1984). The trick is to train the expert into thinking out loud, which requires practice because the act of speaking one's thoughts can actually interfere with or inhibit trains of thought for some individuals. Other individuals, those who habitually think out loud, might not actually leave a very good record of their reasoning.

Audiotapes are necessarily a partial representation of the key information. An expert's facial expressions and gestures can also reveal inference-making processes (McNeill and Levy 1982). For instance, in the analysis of 10 hours of unstructured interviews (Hoffman 1986), I found that a majority of the 957 questions asked by the interviewer were not marked by clearly rising intonation. Typically, the interviewer was repeating some bit of knowledge that had just been asserted by the expert. The intention was that the expert would either confirm, deny, or qualify the interviewer's statement. The pragmatics of such utterances as "ah" and "um" (and their gestural counterparts) are only captured as ephemera on audiotape.[4]

The moral here is that the interviewer should always take copious notes and not rely passively on audiotapes. Furthermore, it is prudent to distinguish between interviews, which are one-on-one, and discussion groups. In general, discussion group sessions need not be recorded. An important exception to this rule is the recording of the discussions of teams of experts who are working on tough cases (to be discussed later).

I: What information are you given about airports?
E: Now, just some rudimentary information comes with it. Common name, latitude and longitude, ah . . . no information comes with it about the ah . . . maximum number of airplanes on the ground or the port capability that is at the field.None of that comes along with it.
I: What's the difference between MOG and airport capability?
E: Ah . . . MOG maximum on the ground is parking spots . . . on the ramp. Airport capability is how many passengers and tons of cargo per day it can handle at the facilities.
I: Throughout . . . ah . . . throughout as a function of . . .
E: It all sorta goes together as throughput. If you've only got . . . if you can only have ah . . . if you've only got one parking ramp with the ability to handle 10,000 tons a day, then your . . . your throughput is gonna be limited by your parking ramp. Or, the problem could be vice versa.
I: Yeah . . .
E: So it's a (unintelligible phrase).
I: What if you only had one loader, so that you could only unload one wide-body cargo airplane at a time? You wouldn't want to schedule five airplanes on the ground simultaneously. How would you restrict that?
E: We know we're not gonna get all the error out of it. We're gonna try and minimize the error. And then we'll say that . . . ah . . . we'll say an arrival-departure interval of one hour, so that means that the probability of having two wide-bodies on the ground tryin' to get support from the loader is cut . . .

Table 3. Example Excerpt from an Unstructured Interview.

The Structured Interview

A powerful and yet conceptually simple alternative to unstructured interviews is the structured interview. In a sense, the structured interview combines an analysis of familiar tasks with an unstructured interview. In order to add structure to an interview, the knowledge engineer initially makes a first pass at a data base by analyzing the available texts and technical manuals, or by conducting an unstructured interview. The expert then goes over the first-pass data base one entry at a time, making comments on each one. Recording this process is not necessary because the knowledge engineer can write changes and notes on a copy of the printout of the first-pass data base.

A structured interview in one way or another forces the expert to systematically go back over the knowledge. Any given comment can have a number of effects on the data base. It can lead to (1) the addition or deletion of entries, (2) the qualification of entries, (3) the reorganization of the hierarchical or categorical structure of the data base, or (4) the addition or deletion of categories. The result is a second pass at the data base.

Limited-Information Tasks

Limited-information tasks represent the application of basic scientific method: to understand how something works in nature, we tinker with it. Limited-information tasks are similar to the familiar tasks, but the amount or kind of information that is available to the expert is somehow restricted. For example, an expert radiologist might like to have available all sorts of information about a patient's medical history before interpreting an X ray. Many expert aerial-photo interpreters like to have all sorts of maps available during the interpretation of photos. In the limited-information task, such contextual information can be withheld, forcing the expert to rely heavily upon (and hence provide additional evidence about) their knowledge and reasoning skills.

In general, experts do not like it when you limit the information that is available to them. It is commonly assumed in familiar tasks that all the relevant information is available and that the expert's judgments are correct. Some experts would rather give no judgment at all than give one that is based on incomplete information. It is important when instructing the expert to drive home the point that the limited-information task is not a challenge of their ego or of their expertise: The goal of the task is not to determine how "clever" the expert is.

Once adapted to the task, experts can provide a wealth of information. The limited-information task is especially useful for revealing an expert's strategies (as opposed to factual knowledge). The incompleteness of the information affords the formulation of hypotheses (rather than final judgments), strategic thinking (What if . . . ?), and the use of heuristics (rules of thumb).

The limited-information task can be used to provide information about subdomains of the expert's knowledge, to fill in any gaps in the data base or a set of inference rules. For example, in the aerial-photo-interpretation project, the structured interview with one particular expert did not yield much information about certain geological forms. Hence, other experts could be run through a limited-information task that involved photos of precisely these geological forms.

Constrained-Processing Tasks

Constrained-processing tasks are like limited-information tasks in that both involve tinkering with the familiar task. Constrained-processing tasks involve deliberate attempts to constrain or alter the reasoning strategies that the expert uses. One simple way to achieve this goal is to limit the amount of time that the expert has in which to absorb information or make judgments. For example, the interpretation of aerial photos typically takes hours, but in a constrained-processing task the expert might be allowed, say, two minutes to inspect a photo and five minutes to make judgments about it.[5]

Another way to constrain the processing is to ask the expert a specific question rather than to require the full analysis that is conducted during the familiar task. Two subtypes of a constrained-processing task that involve this single-question processing are what can be called the method of simulated familiar tasks and the method of scenarios.

The Method of Simulated Familiar Tasks

Here, a familiar task is performed using archival data. The best example of this method I've seen is used in probing the knowledge of expert weather forecasters (Moninger and Stewart 1987). The experts have to do a forecast but in displaced real time. In this simulated familiar task, the clock can be stopped at any point and the expert queried with regard to reasoning strategies or subdomains of knowledge as incoming data are monitored and predictions are made.

The Method of Scenarios

While analyzing cases, experts often draw analogies to previously encountered situations or cases. A given case is explored in terms of any relevant or salient similarities and differences (at either perceptual or conceptual levels) relative to a previously encountered case. For example, Klein (1987) examined how expert mechanical and electrical engineers design new components by analogy to other components that perform similar functions. Moninger and Stewart (1987) documented how expert weather forecasters make predictions about the growth of storms based on analogies to previously experienced storms.[6]

One type of constrained-processing task involves having the interviewer deliberately encourage the use of scenarios during the performance of a familiar task. Such a practice should evoke evidence about the expert's reasoning for the kinds of scenarios involved in the data at hand.

Combined Constraints

A task can involve combining limited-information constraints with processing constraints. For example, the expert aerial-photo interpreter could be asked to interpret a photograph without the benefit of maps and with only two minutes in which to view the photo (Hoffman 1984).

If an expert does not like limited-information tasks, chances are the opinion will be the same about constrained-processing tasks. In general, the more constrained the task--or the more altered it is relative to the familiar tasks--the more uncomfortable the expert is doing the task. However, in my experience, once an expert begins to open up and becomes less hesitant about giving uncertain or qualified judgments, the limited or constrained tasks can be good sources of information about the expert's reasoning strategies.

The Method of Tough Cases

Research on an expert's reasoning encounters special difficulty during its later phases when some of the expert's knowledge and methods of reasoning have already been described. The task that now confronts the knowledge engineer is to get evidence about the subtle or refined aspects of the expert's reasoning. One needs to know something about the expert's reasoning that is not already known.

Subtle or refined aspects of an expert's reasoning are often manifested when an expert encounters a tough case, a case with unusual, unfamiliar, or challenging features. It is usually intuitively obvious to an expert when a case is a tough one. Almost by definition, such cases are rare. As a consequence, the knowledge engineer must adopt special methods of study. The limited-information and constrained-processing tasks work by making routine cases challenging; they do not turn routine cases into tough cases.

Methodologically, the method of tough cases is quite simple. The knowledge engineer is not likely to be present when an expert encounters a tough case; so, the expert is equipped with a small tape recorder and is instructed how to make a verbal protocol of personal ruminations when encountering a tough case.

It seems to me that for all important domains of expertise--those which take years of practice--all experts and teams of experts should routinely make recordings of their tough cases. Expertise should never be wasted. Although the analysis of protocols can be time consuming (more on this point later), benefits will no doubt accrue from recording the knowledge that took someone a lifetime to acquire. Ultimately, recording of knowledge might be the crowning achievement of the modern work on expert systems. As Doyle (1984) put it, "It may be fruitful to separate training in articulate apprenticeship from

METHOD	ADVANTAGES	DISADVANTAGES
ANALYSIS OF FAMILIAR TASKS	The expert feels comfortable comfortable	Can be fairly time-consuming
INTERVIEWS	For a first- and second-pass at a data base, it can generate much information.	Typically very time-consuming.
LIMITED INFORMATION TASKS	Can be tailored to extract information on selected subdomains of knowledge.	Expert feels uncomfortable and is hesitant to make judgments.
CONSTRAINED PROCESSING TASKS	Can be tailored to extract information on selected subdomains of knowledge, or on the expert's strategies	Expert feels uncomfortable and is hesitant to make judgments.
ANALYSIS OF "TOUGH CASES"	Can yield information about refined reasoning	Occur unpredictably, the knowledge engineer may not be present.

Table 4. Some Salient Advantages and Disadvantages of the Various Methods for Extracting an Expert's Knowledge.

training in current computer systems, for the former will be useful today and tomorrow, while the latter will continually become obsolete" (p. 61).

Comparison of the Methods

For an independent example of how all these various methods crop up in work on expert systems, see the section "Step by Step" in a review article by Patton (1985). Looking through her discussion, one can ferret out an example of a structured interview, an analysis of tough cases, a simulated familiar task, and a constrained-processing task. Given all these various tasks, how can they be compared? One way to compare them is in terms of their relative advantages and disadvantages, some of which I have already alluded to for the various tasks. Table 4 summarizes these points.

Apart from the qualitative advantages and disadvantages, the experimentalist would like to have some quantifiable criteria for use in a comparative analysis. Some possible criteria are presented in table 5.

Task and Material Simplicity

The method of familiar tasks and the method of tough cases are essentially equal in terms of the simplicity of the tasks and the materials, as are the limited-information and constrained-processing tasks. For all these methods, the instructions involved are about a page or so long, and the materials will be one or a few pages of information, displays, graphs, and so on. The structured interview stands out because it requires a first-pass data base, which can itself consist of a relatively large and complex set of materials.

Task Brevity

Ideally, one wants to disclose the expert's reasoning as quickly as possible. Familiar tasks can take anywhere from a few minutes (X-ray diagnosis) to an hour (weather forecasting) to an entire day (aerial-photo interpretation). Interviews take on the order of days or even weeks. Although a time-consuming process, interviews can yield a great deal of information, especially when used in the initial phase of developing a data base. Limited-information and constrained-processing tasks can be intensive and should be designed to take somewhere between 15 and 45 minutes. Although they are not very time consuming, these tasks do require time for transcribing and coding the audiotapes (more on this point later).

Task Flexibility

Ideally, the task should be flexible: it should work with different sets of materials and with variations in instructions. For some experts, abundant information about reasoning can be evoked by the simplest of questions. For other experts, the verbalizations might be less discursive. For some experts, a tape recorder might be absolutely necessary; for others, shorthand notes might suffice.

Task Artificiality

The task should not depart too much from the familiar tasks. After all, one does not want to build a model of reasoning based on research evidence gained from tasks that never occur in the expert's experience. The further the task departs from the usual problem-solving situation, the less it tells the knowledge engineer about the usual sequences of mental operations and judgments. However, deliberate violations of the constraints that are involved in the familiar tasks (that is, tinkering with the familiar tasks) can be used to systematically expose the expert's reasoning.

None of the tasks that I have described in this paper depart radically from the familiar tasks. They do differ in the degree to which they relax the constraints that are involved in the familiar tasks in order to evoke instances of reasoning.

Data Format

Ideally, the data that result from any method should be in a format ready to be input into the data base. Only the structured interview has this characteristic. The other methods can result in verbal protocol data, which need to be transcribed and analyzed.[7]

Data Validity

The data that result from any method should contain valid information, valid in that it discloses truths about the expert's perceptions and observations, the expert's knowledge and reasoning, and the methods of testing hypotheses. In addition to validity in the sense of truth value, one would like to feel that a data base is complete--that it covers all the relevant subdomains of knowledge. One would also like to know that different experts agree about the facts. This agreement is validity in the sense of reliability or consensus across experts. Finally, one would like to have some assurance that a set of facts includes the most important ones.

Without such valid information, an expert system cannot be built. How can one be sure that a data base represents valid information?

CRITERION	OPERATIONAL DEFINITION
SIMPLICITY OF THE TASK	Brevity of the instructions that are necessary to specify exactly what the expert is expected to do, and in what order.
SIMPLICITY OF MATERIALS	The number of stimuli or other materials that are needed, and their complexity, relative to the familiar task.
BREVITY OF TASK	Total time taken by the task, including the reading of the instructions and the analysis of the data.
FLEXIBILITY OF TASK	Is the task adaptable to different materials, to variations in the instructions, or to different experts?
ARTIFICIALITY OF TASK	Does the task depart much from the familiar task?
DATA FORMAT	Are the data in a format ready for inputting into a data base?
DATA VALIDITY	Do the data records provide correct evidence about important knowledge and reasoning strategies?
METHOD EFFICIENCY	How many informative propositions are produced per task minute?

Table 5. Some Criteria by Which Methods Can Be Analyzed and Compared.

Validity: The Importance of the Data Some of an expert's statements will obviously be irrelevant (Do you want a cup of coffee?). Excepting these statements, the knowledge engineer generally takes it for granted that a given statement by the expert is relevant to the domain of expertise being studied. However, just because a given statement (or datum) is relevant does not mean that it is at all important.

Validity in the sense of the relative importance of a given fact or rule can be assessed in a number of ways. One way is the direct approach--ask the experts some more-or-less structured questions to get their judgments of the relative importance of a data-base entry. Another way to assess importance is to see how early or how often a given fact crops up in the data from various experts.

Validity: Reliability or Consensus The running of additional experts in a knowledge-extraction task presumably has as its major effect the generation of lots of information which is redundant in that it repeats data which are already in the data base (having come from earlier studies of some other expert). This redundancy can be taken as evidence of the validity of the data, validity in the sense of agreement or reliability across different experts.

How many experts is enough? I am not sure there can be a principled answer to this question. Some domains have only one expert, leaving one with little choice. At the other extreme is the "shotgun" approach of interviewing as many different experts as possible (for example, Mittal and Dym 1985) in order to get a complete data base.

Validity: The Completeness of the Data Base Having more than one expert perform various tasks can also help assure validity--in the sense of the completeness of the data base. Indeed, a good rule of thumb is to "pick the brains" of more than one expert for any

TYPES OF PROPOSITIONS

		OBSERVATION STATEMENTS	INFERENCES
STATUS	OLD	Quite easy to obtain. Requires no special methods	Relatively easy to obtain Requires no special methods
	NEW	Relatively difficult to obtain Requires special methods	Relatively very difficult to obtain Requires special methods

Table 6. A Matrix for Classifying the Data Generated by a Task That Is Intended to Extract the Knowledge of an Expert. The frequency of propositions of each of two types (facts or observation statements, and inferences or potential if-then rules) can be used to assess the relative difficulty of extracting the two types of knowledge· "old" or "new" relative to the current data base.

domain in which different experts have widely different degrees of experience with different subdomains. In addition to their idiosyncratic knowledge, they might have idiosyncratic strategies (Mittal and Dym 1985). Thus, having more than one expert go through the various knowledge-extraction tasks should have the effect of "pushing" the data base toward completeness by filling in the gaps, which occurred in the aerial-photo-interpretation project. One expert knew a lot about desert forms, another expert knew a lot about glacial and arctic forms.

Validity: The Truth Value of the Data

The question of how many experts is enough cuts two ways. The study of more than one expert is bound to generate some disagreements, and disagreements are indigestible contradictions as far as any logic-based machine system is concerned. Nevertheless, it might become inevitable that contradictions arise as one refines a data base, adds details, and fleshes out the experience-dependent subdomains of knowledge.

It has been proposed that expert system development projects should only use one expert precisely in order to avoid contradictions (for example, Prerau 1985). This strategy assumes that disagreements are pervasive, but it also seems to assume an underlying reason for the pervasiveness--it is the expert knowledge itself that is plagued by uncertainty. If this statement is true in a given domain, one obviously runs the risk of generating a system that is wholly idiosyncratic at best and based on trivial knowledge at worst. In any event, it could be that the domain at hand is not really well-suited to the application of expert system tools. The advice of trying to avoid contradictions seems to the experimental psychologist to miss the point of expert system work. Disagreements should be used as clues about the basic research that might be needed to fill in the knowledge gaps, perhaps even before any expert system work can be done.

Less extreme than the assumption of pervasive uncertainty and disagreement is the assumption that experts in a given domain agree about most of the basic facts and relevant causal laws, but might disagree about how to go about applying them. An example is the project on the effects of sunspots on the weather (McIntosh 1986). Although the relevant laws are known (that is, electromagnetism, particle physics, and so on), the dynamics of the sun's atmosphere are only partially understood. Thus, the solar weather experts sometimes disagree on how to go about predicting solar phenomena.

Another realistic possibility is the experts agreeing on the relevant laws and about how to go about applying these laws but disagreeing about hypotheses for some specific cases. Such occurrences routinely happen in knowledge-extraction tasks that involve more than one expert at a time. For example, weather forecasters can be paired in the simulated familiar task (Moninger and Stewart 1987) and they often enter into debates as hypotheses are hashed about. Aerial-photo interpreters sometimes work in teams of research groups. For cases such as these team efforts, the experts' discussions, debates, and exchanges of hypotheses can be very informative, especially when the team encounters a tough case.

Whether disagreements among experts are pervasive, are due to uncertainty in the application of relevant knowledge, or are due to uncertainty about specific hypotheses, it is ultimately the responsibility of the knowledge engineer to resolve any unreliable or contradictory rules or facts. Suppose that each of a set of expert aerial-photo interpreters produces a rule about the identification of limestone sinkholes but that the rules are somehow mutually incompatible. The knowledge engineer really has only two choices: either combine the rules into a new logically coherent rule, or select one of them for use. In any event, only one rule for identifying limestone sinkholes can appear in the expert system (that is, there can be no contradictions in its logic). Formally, the machine can only model one expert at a time.

Method Efficiency

Perhaps the most important criterion in a practical sense is overall efficiency. For the sake of people out there "in the trenches," I thought I'd try to apply some efficiency metrics to the various methods. In the case of knowledge acquisition, efficiency involves the number of propositions generated per unit of time. Time is expressed as total task time, that is, the time it takes the knowledge engineer to prepare to run the expert at the task plus the time it takes to run the task plus the time it takes to analyze the data.

One can assess the production rates of propositions of different types. One seeks not only facts (predications or observation statements), one also seeks information about potential if-then rules (the expert's inference strategies and heuristics). Furthermore, one seeks not just propositions of these types but new ones, informative ones because they are not already in the data base. Table 6 presents this classification of propositional data.

Table 7 presents some efficiency

analyses for the projects I've been involved with. I present these analyses not to imply that knowledge engineers should routinely perform such analyses, but hopefully to save them the trouble. I suspect that the rate statistics presented here are representative of what people can expect in work on expert systems.

Let me focus on unstructured interviews because this knowledge-acquisition method is the most commonly used. For unstructured interviews, one can assess efficiency in a number of ways. Total time can be divided into the number of propositions obtained (for both observation statements and potential if-then rules). Also, the number of propositions obtained can be divided into the number of questions asked by the interviewer (or the number of pages of transcript).

For the project on expert airlift planners (Hoffman 1986), the analysis of the 10 hours of taped unstructured interviews yielded a total task time of 100 hours, a total of 957 questions asked by the interviewer, and a total of 753 propositions (including both observation statements and potential if-then rules). Thus, the unstructured interview generated only approximately 0.13 propositions per task minute and approximately 0.8 propositions for each question asked by the knowledge engineer.

The unstructured interview produced lots of propositions in the first hours of the interview (over 5 propositions per page of transcript or over 200 propositions per hour), but the rate trailed off rapidly (to less than 1 proposition per page of transcript or about 40 propositions per hour). In part, this result was due to the fact that in the later sessions, the interviewer and expert spent time sitting at a computer terminal and informally discussed user needs and system design. In just the first 5 of the 10 hours, the unstructured interview yielded approximately 1.6 propositions per task minute.

In general, if a knowledge engineer finds that knowledge is being extracted at a rate of about two new propositions per task minute, the engineer can be assured of probably being on the right track. If the rate is closer to one proposition per task minute, chances are there is some inefficiency somewhere. If the rate is closer to three per minute, the knowledge engineer is golden.

METHOD	RESULTS
UNSTRUCTURED INTERVIEW	For each question the interviewer asked, the method generated 0.8 propositions. The method generated 0.13 propositions per task minute.
STRUCTURED INTERVIEW	Of the 1,400 propositions in the first-pass data base, 30 percent were modified. The first-pass data base was increased in size by 15 percent. The method generated about one new proposition per task minute.
LIMITED INFORMATION AND CONSTRAINED PROCESSING TASK	The method generated between one and two new observation-related propositions and between one and two new inference-related propositions per task minute.
METHOD OF "TOUGH CASES"	The method generated between one and two new observation-related propositions and between one and two new inference-related propositions per task minute.

Table 7. A Comparison of the Results for Four Methods in Terms of Some Measures of Efficiency The unstructured interview results are from the project on expert airlift planners (Hoffman 1986). The other results are from the aerial photo interpretation project (Hoffman 1984) The third task listed here is one that combined limited information with processing constraints. All rate computations are based on total task time.

Recommendations

Overall, the unstructured interview is not too terribly efficient. This finding deserves emphasis because most expert systems have apparently been developed using unstructured interviews Preliminary unstructured interviews can be critical for obtaining information on three key questions: (1) Who are the experts? (2) Is this problem well-suited to the expert system approach? and (3) What are the needs of the people who will use the system? However, if the knowledge engineer needs to become familiar with the domain, then an analysis of familiar tasks should be conducted and a first-pass data base gleaned from this analysis. Once a first-pass data base has been produced, then unstructured interviews (using note taking rather than tape recording) can be used to determine user needs. If the knowledge engineer wishes to build a refined or second-pass data base, structured interviews are more efficient. In fact, they can be many times more efficient overall, according to my results.

As data bases become very large, it becomes necessary to use special tasks. Limited-information or constrained-processing tasks or the method of tough cases can be used in order to focus on specific subdomains of knowledge or to disclose aspects of reasoning that the data base might lack. Otherwise, the knowledge engineer might spend hours working at an inefficient task, inefficient in that it produces lots of propositions which happen to already reside in the data base.

All experts differ, and all domains of expertise differ; so, too, will all expert system-development projects differ. Some methods for extracting an expert's knowledge will fit some projects and not others. For example, an analysis of familiar tasks might not be very fruitful for domains in which there are no texts or in which the available technical manuals provide little information about what the experts actually do. Such was the case for the airlift-planning project (Hoffman 1986). It made sense in that project to begin with an unstructured interview to learn not only about user needs but to simultaneously get some very basic information about what airlift planners actually do in their planning tasks (that is, the familiar task).

Despite such domain specificity, I can propose a generalized series of steps for extracting the knowledge of experts, as shown in table 8. Situations probably exist for which my recommended series of steps is not quite fitting. However, I'd wager that one or another slight variation on this theme would fit.

The figures in table 8 for "Effort Required" are approximate ranges based on my experience. According to my estimates, it can take at least three months to get to the point of having a refined or second-pass data base. How does this time frame compare with the experiences of other artificial intelligence (AI) researchers? Actually, it is impossible to tell from reading the expert system literature exactly how much effort went into building the data base in any expert system project. (It is for this reason that I feel compelled to present my own "ballpark" estimates.) Typically, authors do not even state how the knowledge was obtained--sometimes from texts and manuals, sometimes from archived data, and usually from unstructured interviews. Typically, research reports jump right into a discussion of system architecture.

Some reports on expert system projects state how long it took to develop a prototype, but one can only guess how long it took just to acquire (or extract) the data. The development of MYCIN (Davis, Buchanan, and Shortliffe 1977), a system for diagnosing infectious diseases, took many years. The development of INTERNIST, another medical-diagnosis system, took 10 years with the help of a full-time specialist in internal medicine (Buchanan 1982). R1, which configures the VAX computer (McDermott 1980), took two man-years to develop by a team of about a dozen researchers and is still being refined. In general, it takes one to two years to develop a prototype and about five years to develop a full-scale system (Duda and Shortliffe 1983; Gevarter 1984).

In contrast, the PUFF system for the diagnosis of pulmonary disorders was reported to have been developed in less than 10 weeks (Bramer 1982). The likely reason for this brevity was that most of the rules were easily gleaned from archived data (Feigenbaum 1977), and only one week was spent interviewing the experts (Bramer 1982).

Apparently, if one is relying on unstructured interviews, is interviewing a large number of experts, or is working on a system that requires hundreds or thousands of rules and facts in its data base (as in the INTERNIST system), then many years of effort are required to develop the system to the prototype stage.

In this light, a minimum of three months to build a second-pass data base seems like an underestimate. However, it cannot be a gross underestimate. The second-pass data base for the aerial-photo-interpretation project (Hoffman 1984) contained over 1400 rules and facts, and it took me about 45 days to develop it by means of structured interviews and special tasks.

AI researchers will not be eager to use all the methods and carry out all the efficiency computations presented here, but I do not propose that they should. I do not want to shoehorn the "tricks of the trade" of building expert systems into the rigid statistics-oriented structure of a psychological experiment. However, I do want to strongly recommend that developers of expert systems routinely report the methods used to extract experts' knowledge and the efficiency of the methods (that is, the amount of effort taken to construct the data base).

Some Lingering Issues

None of the methodological ideas that are presented here is intended to be logically airtight. My categorization of methods is intended to be pragmatic rather than complete. The criteria for analyzing methods are probably insufficient. I welcome reactions and suggestions and hope that these ideas are helpful to others.

To those who have already developed an expert system, many of the research ideas I have referred to might seem detailed (if not trivial). I wanted to spill out the gory details for the sake of those who have not yet attempted to generate an expert system. As I said at the outset, I address the expert system engineering process from a nuts-and-bolts perspective. To those who are not familiar with experimental psychology, it might be surprising that experimentalists deal with minutiae such as better ways of tape recording or the variety of ways to assess the validity of a data set. However, such minutiae are precisely the sorts of things that experimental psychologists like to attend to. Thus, my discussion focused on methods and methodology (the analysis of methods). Hidden in the discussion are a few broader issues that merit explicit consideration and a wee bit of feather ruffling.

The Social Psychology of Expertise

Should the knowledge engineer be the expert? In other words, should the expert become a knowledge engineer? I do not know if there can be a principled answer to this question. Some researchers claim that the knowledge engineer should be the expert (Friedland 1981), that "expert systems are best conceived by the experts themselves" (Taylor 1985, p. 62). Some researchers claim that the situation in which experts interview other experts can be disastrous. Other researchers say the more the expert knows about AI, the more likely it is that the expert has biases (for example, about which knowledge representation format to use) (McIntosh 1986).

STEP	ACTIVITY	PURPOSE	EFFORT REQUIRED
1	Analysis of Familiar Task	Familiarizes the knowledge engineer with the domain. Yields information for the first-pass data base.	One month
2	(Optional) Unstructured Interview	Familiarizes the knowledge engineer with the domain. Yields information for the first-pass data base. Helps in determining user needs and system design.	One to two weeks
3	Preparation of First-pass Data base	Beginnings of a computer file.	One month or more
4	(Optional) Unstructured Interview	Familiarizes the knowledge engineer with the domain. Yields information for the second-pass data base. Helps in determining user needs and system design.	One to two weeks
5.	Structured Interviews	Yields information for the second-pass data base.	One to two weeks
6.	Preparation of Second-pass Data Base	Refinements of the computer file	Two weeks or more
7.	Special Tasks	Yields further refinements of the data base.	One week or more per experiment

Table 8 Some Steps for Extracting and Characterizing the Knowledge of an Expert Prior to the Construction of an Expert System The analysis of familiar tasks includes the analysis of texts and technical manuals. The activity designated "Special tasks" includes limited information tasks, constrained processing tasks and analyses of "tough cases " All figures for "effort required" are approximations based on the author's experience.

The answer to this question probably lies in the social-psychological dynamics of particular working groups. Certainly, the expert should know enough about AI to be skeptical and enough about expert systems to appreciate the ways in which the system can aid, rather than replace them.

Are Expert Systems AI, or Are They Cognitive Simulation?

Some expert systems have an "explanation" component; that is, one can query the system about a given conclusion or inference. Typically, the system returns with what is essentially a list or a printout of the rules involved. Such an explanation is not very adequate. Until expert systems deal with reasoning and knowledge at a semantic or conceptual level, their artificiality will remain painfully obvious, and they might not even merit being called intelligent.

Concepts are typically defined in AI using logically necessary and sufficient conditions. However, can an expert system work (or work well) without also being a "cognitive simulation?" To what degree does a given expert system need to represent conceptual knowledge? The jury is still out. We cannot be certain the purely rule-based approach will always effectively or efficiently solve the various problems that people would like their expert systems to solve.

Most concepts do not exist in human cognition in any pristine form. Experts, unlike expert systems, do not always reason in terms of a logical set of rules (Dreyfus and Dreyfus 1986). What is clear is that in many domains, experts reason in terms of their perceptual and conceptual understanding, such as their "mental models" of causal laws (Gentner and Stevens 1983; Sridharan 1985).

When the expert aerial-photo interpreter looks at a photo, the complex geo-biological events that led to the formation of the terrain are perceived (for example, mountain building, glaciation, and so on). When the expert cardiologist listens to heart sounds over a stethoscope, the complex biomechanical events that occur in the cardiovascular system are perceived (Jenkins 1985; Johnson et al. 1982). Such perceptual and conceptual understanding, the result of perceptual-learning and concept-formation processes, might be direct (Gibson 1979); that is, it might not be mediated by any analysis of naked if-then rules, serial or otherwise.

Recently, attempts have been made to incorporate conceptual knowledge into AI systems (Chandrasekaran and Mittal 1983; Kunz 1983; Pople 1982) and to define rule systems that behave like the human reasoner (Sridharan 1985). As instances of AI systems, expert systems might not need to precisely simulate cognition in order to get the job done. However, we do need to learn more about experts' reasoning through basic research if our expert systems are ever to get the job done well, which brings me to the last issue I address.

Does AI Have Its Foundations in Philosophy or Research?

The cognitive-simulation work hints at the fact that some AI workers are really closet psychologists. They rely on psychological terms and mentalistic metaphors, and they work on psychological problems. More to the point, the problems they work on often beg for an empirical answer, an answer based on research findings. However, AI workers tend to turn to computing theories and abstract concepts to produce solutions.

No better example of this tendency exists than that provided by the field of expert systems itself. Having recognized that the knowledge-acquisition process is a bottleneck, what solution do AI workers seek? Let's build another program! Davis and Lenat (1982, p. 348) assert that a system which has knowledge about its own representations can allow for "knowledge acquisition using a high-level dialog." But can't humans use dialog to extract knowledge, or did I miss something? AI really does like to put the logical cart before the empirical horse.

In the literature on expert systems, I've found numerous assertions, such as the claim that experts are better able to build good data bases than computer scientists. Such assertions might seem solid and might indeed be correct, but they are rarely empirically grounded, which causes the experimental psychologist to yank out clumps of hair.

The broader literature on AI and cognitive science is characterized by its penchant for philosophical speculation (Hoffman and Nead 1983). The typical issue of a cognitive science technical journal contains evermore clever theories, theories that combine "schemas," "scripts," "inference nets," and "labeled relations" in all sorts of clever computational ways. However, there is rarely a clear statement about whether the theory has an evidential base. Sometimes, no statement even exists about whether the theory is supposed to have anything at all to do with heads or whether it is just supposed to be about computers, which causes the experimental psychologist to yank out even larger clumps of hair.

While remembering that the roots of AI are in the philosophy of the mind (cf. McCarthy and Hayes 1969), AI practitioners seem to have forgotten about the roots in experimental psychology. It is the rare paper that acknowledges the need for basic research on cognition, that is not put off by the experimental psychologist's stubborn concern with methodological details, or that is sensitive to the difference between AI and cognitive simulation.[8]

If one is at all interested in cognition or cognitive simulation, then experiments are necessary to ensure that the model works like the mind does, or to test hypotheses about an expert's strategies. One cannot simply assert, for instance, that human or computer memories must rely on schemas because schemas are computationally elegant. Such hasty epistemological pronouncements will never do.[9]

If AI is to solve the really tough problems, it would do well to put a bit less effort into short-term solutions (for example, reliance on naked if-then rules) and premature applications (for example, expert systems that rapidly become too complex to be rapidly computable) and a bit more effort into systematic research (for example, how experts really reason). In the past few years, federal funding for basic research on cognition and graduate training in experimental psychology have suffered from much more than their fair share of cuts. Rather than just bemoaning the lack of trained knowledge engineers, anyone who is interested in making a real contribution to our needs in the field of AI might consider investing some resources in the support of basic research on human cognition and perception.

The Benefits of Collaboration

The collaboration of AI researchers and experimental psychologists can be of benefit to work on expert systems. I hope my article has made this point. Benefits can also accrue for psychology, however. For example, a major theory of memory asserts that forgetting is caused by the interference of new learning with old (Reder 1987; Smith, Adams, and Schon 1978). This hypothesis generates what is called the *paradox of expertise:* how can experts deal so adroitly with so many remembered facts? Hopefully, research on the cognition of experts can help clarify this paradox.

Expert system work also bears on theories of perceptual learning. Although it is obvious experts have special knowledge and concepts, it should be kept in mind that they also have special perceptual skills (Shanteau 1984). For example, the expert radiologist "sees" X rays differently than the first-year medical student (Feltovich 1981). Another, more commonplace example is the expert sports commentator who "sees" something that we novices can only pick up when shown the slow-motion replay.

Lifetimes of good research could be carried out to distinguish experts from novices in terms of their perceptual processes, learning processes, and reasoning strategies. What information do experts focus on? What reasoning biases do they have? How can experts be identified?

What I find particularly appealing about such questions is that the research they engender is practical and basic at the same time--practical because it contributes to the solving of important problems, basic because it contributes to the body of knowledge. The collaboration of experimental psychologists and AI researchers can be of great mutual benefit.

Acknowledgments

Most of the ideas reported here were first developed under the aegis of the U S. Army Summer Faculty Research and Engineering Program (Delivery Order No 0550). The work was conducted in the summer of 1983 at the Engineer Topographic Laboratories (ETL), U.S. Army Corps of Engineers, Ft Belvoir, Virginia The author would like to thank Olin Mintzer, master aerial-photo interpreter, for his help and guidance The author would also like to thank all the people at the Center for Physical Science and the Center for Artificial Intelligence at ETL for their many stimulating discussions

Clarifications and amplifications arose in discussions at the Artificial Intelligence Research in Environmental Science Conference held at the Environmental Research Laboratories of the National Oceanic and Atmospheric Administration (NOAA), Boulder, Colorado, 28-29 May 1986 The author would especially like to thank William Moninger of NOAA for his encouragement

Many others deserve thanks for their encouragement and support of the research reported here: Gary Grann, Lieutenant Colonel John Whitcomb, Captain Eric Goepper, Captain Thomas Moran, Captain Jeffrey Valiton, 2d Lieutenant Joseph Besselman, and Lieutenant Ray Harris of the Strategic Planning Directorate, Electronics System Division, Hanscom Air Force Base, Massachusetts; Murray Daniels of MITRE Corporation; Rodney Darrah, Susan Espy and the people at Universal Energy Systems, Inc ; Karen Thompson and Pat Carey at Adelphi University.

Preparation of this paper was supported by a grant from Summer Faculty Research Program (Contract No F49620-85-C-0013) of the U.S. Air Force Office of Scientific Research. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes not withstanding any copyright notation herein.

References

Bierre, P 1985 The Professor's Challenge *AI Magazine* 5(4): 60-70

Boose, J. H. 1984 Personal Construct Theory and the Transfer of Human Expertise. In Proceedings of the Third National Conference on Artificial Intelligence, 27-33. Menlo Park, Calif : American Association of Artificial Intelligence.

Bramer, M A 1982 A Survey and Critical Review of Expert Systems Research. In *Introductory Readings in Expert Systems,* ed D. Michie, 3-29. London: Gordon and Breach

Buchanan, B. G 1982 New Research on Expert Systems In *Machine Intelligence,* vol 10, eds J. Hayes, D Michie, and Y-H Pao New York: Wiley

Chandrasekaran, B., and Mittal, S 1983 Deep versus Compiled Knowledge Approaches to Diagnostic Problem Solving. *International Journal of Man-Machine Studies* 19:425-436

Cooke, N. M 1985 Modeling Human Expertise in Expert Systems, Technical Report, MCCS-85-12, Computing Research Laboratory, New Mexico State Univ.

Davis, J 1986 Summary of the Workshop on Knowledge Acquisition Paper presented at Artificial Intelligence Research in Environmental Science Conference. Boulder, Colo : NOAA

Davis, R ; Buchanan, B G ; and Shortliffe, E H 1977 Production Systems as a Representation for a Knowledge-Based Consultation Program. *Artificial Intelligence* 8:15-45.

Davis, R and Lenat, B. 1982 *Knowledge-Based Systems in AI.* New York: McGraw-Hill

Denning, P J 1986. Towards a Science of Expert Systems *IEEE Expert* 1(2): 80-83.

Doyle, J 1984 Expert Systems without Computers or Theory and Trust in Artificial Intelligence *AI Magazine* 5(2): 59-63.

Dreyfus, H , and Dreyfus, S. 1986 Why Expert Systems Do Not Exhibit Expertise. *IEEE Expert* 1(2): 86-90.

Duda, R. O , and Gashnig, J. G. 1981 Knowledge-Based Expert Systems Come of Age. *Byte* September:238-281.

Duda, R O., and Shortliffe, E. H 1983. Expert Systems Research. *Science* 220:261-276.

Einhorn, H J 1974. Expert Judgment *Journal of Applied Psychology* 59:562-571.

Eliot, L. B. 1986. Analogical Problem Solving and Expert Systems. *IEEE Expert* 1(2): 17-28

Ericcson, K. A , and Simon, H A. 1984 *Protocol Analysis: Verbal Reports as Data.* Cambridge, Mass : MIT Press

Feigenbaum, E A 1977 The Art of AI. In Proceedings of the Fifth International Joint Conference on Artificial Intelligence International Joint Conference on Artificial Intelligence

Feltovich, P. J. 1981 Knowledge-Based Components of Expertise in Medical Diagnosis, Technical Report, PDS-2, Learning Research and Development Center, Univ. of Pittsburgh.

Freiling, M.; Alexander, J.; Messick, S.; Rehfuss, S.; and Shulman, S 1985. Starting a Knowledge Engineering Project: A Step-by-Step Approach *AI Magazine* 6(3): 150-163.

Friedland, P. 1981. Acquisition of Procedural Knowledge from Domain Experts In Proceedings of the Seventh International Joint Conference on Artificial Intelligence, 856-861 International Joint Conferences on Artificial Intelligence

Gentner, D., and Stevens, A , eds 1983 *Mental Models* Hillsdale, N.J : Lawrence Erlbaum

Gevarter, W B. 1984 *Artificial Intelligence Computer Vision and Natural Language Processing* Park Ridge, N.J : Noyes

Gibson, J G. 1979. *The Ecological Approach to Visual Perception* Boston: Houghton-Mifflin

Gorfein, D., and Hoffman, R , eds. 1987 *Memory and Learning. The Ebbinghaus Centennial Conference.* Hillsdale, N J.: Lawrence Erlbaum. Forthcoming.

Hartley, R 1981 How Expert Should Expert Systems Be? In Proceedings of the Seventh International Joint Conference on Artificial Intelligence International Joint Conferences on Artificial Intelligence

Hayes-Roth, F ; Waterman, P. A ; and Lenat, D., eds. 1983 *Building Expert Systems.* Reading, Mass : Addison-Wesley.

Hoffman, R. 1986 Procedures for Efficiently Extracting the Knowledge of Experts, Technical Report, U.S. Air Force Summer Faculty Research Program, Office of Scientific Research, Bolling Air Force Base.

Hoffman, R. R. 1985. Some Implications of Metaphor for the Philosophy and Psychology of Science. In *The Ubiquity of Metaphor,* eds W. Paprotte and R Dirven, 327-380. Amsterdam, Netherlands: John Benjamin

Hoffman, R. R. 1984. Methodological Preliminaries to the Development of an Expert System for Aerial-Photo Interpretation, Technical Report, ETL-0342, The Engineer

Topographic Laboratories

Hoffman, R R 1980 Metaphor in Science In *Cognition and Figurative Language,* eds R Honeck and R Hoffman, 393-423 Hillsdale, NJ: Lawrence Erlbaum Associates

Hoffman, R R , and Kemper, S 1987 What Could Reaction-Time Studies Be Telling Us about Metaphor Comprehension? In *Metaphor and Symbolic Activity* vol 2 Forthcoming

Hoffman, R R., and Nead, J M 1983 General Contextualism, Ecological Science, and Cognitive Research *The Journal of Mind and Behavior* 4 507-560

Jenkins, J J 1985 Acoustic Information for Objects, Places, and Events In *Persistence and Change Proceedings of the First International Conference on Event Perception,* eds W Warren and R Shaw, 115-138 Hillsdale, N.J Lawrence Erlbaum

Johnson, P E.; Hassebrock, F ; Duran, A S.; and Moller, J H. 1982 Multimethod Study of Clinical Judgment. *Organizational Behavior and Human Performance* 30 201-230

Kelly, G 1955 *The Psychology of Personal Constructs* New York: Norton

Kintsch, W 1972 *The Representation of Meaning in Memory* Hillsdale, N J : Lawrence Erlbaum

Klein, G. A 1987 Applications of Analogic Reasoning *Metaphor and Symbolic Activity* 2(3): 201-218

Kunz, J 1983 Analysis of Physiological Behavior Using a Causal Model Based on First Principles In Proceedings of the Second National Conference on Artificial Intelligence. Menlo Park, Calif.: American Association for Artificial Intelligence

McCarthy, J 1983. President's Quarterly Message: AI Needs More Emphasis on Basic Research *AI Magazine* 4(4): 5

McCarthy, J , and Hayes, P 1969 Some Philosophical Problems from the Standpoint of AI In *Machine Intelligence,* vol 4, eds B Meltzer and D Michie. Edinburgh, Scotland: Edinburgh University Press

McDermott, J 1980 R1: A Rule-Based Configurer of Computer Systems, Technical Report, Computer Science Dept., Carnegie-Mellon Univ

McIntosh, P. S 1986 Knowledge Acquisition for THEO: An Expert System for Solar-Flare Forecasting Paper presented at Artificial Intelligence Research in Environmental Science Conference Boulder, Colo : NOAA

McNeill, D , and Levy, E 1982 Conceptual Representations in Language Activity and Gesture In *Speech, Place, and Action,* eds R Jarvella and W. Klein, 271-295 New York: Wiley

Mintzer, O , and Messmore, J 1984 Terrain Analysis Procedural Guide for Surface Configuration, Technical Report, ETL-0352, Engineer Topographic Laboratories

Mittal, S , and Dym, C L 1985. Knowledge Acquisition from Multiple Experts *AI Magazine* 6(2) 32-36

Moninger, W , and Stewart, T 1987 A Proposed Study of Human Information Processing in Weather Forecasting. In *Bulletin of the American Meteorological Association* Forthcoming

Nii, H P , and Aiello, N 1979 AGE (Attempt to Generalize): A Knowledge-Based Program for Building Knowledge-Based Programs. In Proceedings of the Sixth International Joint Conference on Artificial Intelligence, 645-655 International Joint Conferences on Artificial Intelligence

Pachella, R G 1974 The Interpretation of Reaction Time in Information-Processing Research In *Human Information Processing: Tutorials in Performance and Cognition,* ed B Kantowitz Hillsdale, N.J : Lawrence Erlbaum.

Patton, C 1985 Knowledge Engineering: Tapping the Experts *Electronic Design,* May:93-100

Politakis, P , and Weiss, S. 1984 Using Empirical Analysis to Refine Expert Systems Knowledge Bases *Artificial Intelligence* 22:23-48

Pople, H E 1982. Heuristic Methods for Imposing Structure on Ill-Structured Problems In *Artificial Intelligence in Medicine,* ed P Szolovits Boulder, Colo : Westview.

Posner, M I 1978 *Chronometric Explorations of Mind* Hillsdale, N.J Lawrence Erlbaum.

Prerau, D S. 1985. Selection of an Appropriate Domain for an Expert System. *AI Magazine* 6(2): 26-30

Purves, W K 1985 A Biologist Looks at Cognitive AI *AI Magazine* 6(2) 38-43.

Quinlan, J. R. 1984 Fundamentals of the Knowledge Engineering Problem In *Introductory Readings in Expert Systems,* ed. D Michie, 33-46. London: Gordon and Breach

Raulefs, P 1985 Knowledge Processing Expert Systems In *AI· Towards Practical Applications,* eds T. Bernold and G Albers Amsterdam, Netherlands: North-Holland

Reder, L M 1987 Beyond Associationism: Strategic Components in Memory Retrieval In *Memory and Learning: The Ebbinghaus Centennial Conference,* eds D Gorfein and R Hoffman. Hillsdale, N J Lawrence Erlbaum Forthcoming

Shanteau, J 1984. Some Unasked Questions about the Psychology of Expert Decision Makers In Proceedings of the 1984 IEEE Conference on Systems, Man, and Cybernetics, ed M El Hawaray. New York: Institute of Electrical and Electronics Engineers, Inc.

Shortliffe, E H 1980 Consultation Systems for Physicians In Proceedings of the Canadian Society for Computational Studies of Intelligence. Victoria, B C : Canadian Society for Computational Studies of Intelligence

Smith, E E.; Adams, N ; and Schon, D. 1978 Fact Retrieval and the Paradox of Interference *Cognitive Psychology* 10:438-464

Sridharan, N S. 1985 Evolving Systems of Knowledge *AI Magazine* 6(3): 108-120

Stefik, M ; Aikins, A.; Balzer, R.; Benoit, J ; Birnbaum, L ; Hayes-Roth, F.; and Sacerdoti, E 1982 The Organization of Expert Systems: A Tutorial *Artificial Intelligence* 18:135-173.

Sternberg, R J 1977 *Intelligence, Information Processing, and Analogical Reasoning* Hillsdale, N J : Lawrence Erlbaum

Taylor, E C 1985 Developing a Knowledge Engineering Capability in the TRW Defense Systems Group *AI Magazine* 6(2): 58-63

Townsend, J T , and Ashby, F G 1983 *Stochastic Modeling of Elementary Psychological Processes.* Cambridge, Mass Cambridge University Press

Way, D. S 1978 *Terrain Analysis* Stroudsberg, Pa.: Dowden, Hutchinson, and Ross

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

Woods, D. D. 1985. Cognitive Technologies: The Design of Joint Human-Machine Cognitive Systems. *AI Magazine* 6(4): 86-92

Notes

1 For discussions of the psychology of expertise, see Cooke (1985), Einhorn (1974), Feltovich (1981), and Shanteau (1984)

2 One exception is the work of Boose (1984), who has adopted ideas about interviewing from psychologist George Kelly (1955) Boose's concern has been with building a computer system for interviewing experts

3. A proposition is generally regarded as an "atomic fact"--an assertion of the existence of some entity and the predication of some property or relation of this entity Treatments that are relevant to protocol analysis can be found in Kintsch (1972, chapter 2) and Ericcson and Simon (1984, chapters and 6).

4. Tape-recording tips: (1) conduct the interview in a small, quiet room; (2) use batteries liberally so you always receive a clear signal; and (3) preface your recording by identifying each of the participants.

5. Constrained-processing tasks can also involve the measurement of reaction time or *decision latency* Such measures are important for the purposes of cognitive simulation and cognitive psychology, because the results can be informative about specific sequences of mental operations (such as memory encoding and retrieval) For a discussion of the logic of reaction-time experiments, see Hoffman and Kemper (1987), Pachella (1974), Posner (1978), or Townsend and Ashby (1983).

6. What is salient here is that the analogic reasoning is in terms of a comparison to previously encountered scenarios Because reasoning by analogy pervades all problem solving, the analogy component of the method of scenarios does not seem as salient for present purposes as the reliance on scenarios to form the analogies For reviews of the abundant literature on the role of analogy and metaphor in problem solving, see Eliot (1986), Gentner and Stevens (1983), Hoffman (1980, 1985), and Sternberg (1977)

7. Some detailed comments are in order about the transcription process for the sake of those who might choose to go this route. As I have already implied, the transcription process takes time Without any doubt, the most time-consuming aspect of the transcription process is the time spent pausing and backing up the tape to interpret and write down the segments where more than one person speaks at a time. The moral: The examiner should consciously try to withhold comments or questions while the expert is talking. The examiner should use a notepad to jot down questions and return to them when the expert's monologue is over The goal of any interview, whether structured or unstructured, is not to edify the knowledge engineer, but to let the expert talk and get the knowledge out so that it can be formalized (hence my earlier distinction between interviews and discussions) It also takes time to code the transcript for propositional content, anywhere from one to five minutes for each page of transcript. The coding should involve at least two independent coders (or judges) until evidence indicates that their respective codings show agreement which is high enough to warrant dropping the extra coders.

8 To be fair, some AI authors are sensitive to the difference, for example, Bierre (1985), McCarthy (1983), Purves (1985,) and Woods (1985). Furthermore, the major AI conventions include sessions on cognitive research, and the AI technical journals include occasional reviews of recent cognitive research

9 I am hesitant to attribute this particular claim about schemas to any one person I've read so many claims about the representation of knowledge (templates, prototypes, networks, semantic features, and so on), and I've yanked out so much hair, that I'd rather ruffle everyone's feathers I also find it worrisome when claims are made about computer or mental representations on the basis of the argument that there are "storage" limitations. Indeed, "storage" might be the wrong metaphor altogether. For further discussion, see Gorfein and Hoffman (1987)