Eliciting Expertise

Nigel Shadbolt,
Dept. of Electronics and
Computer Science,
University of
Southampton,
U.K.

Introduction

Since the last edition of this book there have been rapid developments in the use and exploitation of formally elicited knowledge. Previously, (Shadbolt and Burton, 1995) the emphasis was on eliciting knowledge for the purpose of building expert or knowledge-based systems. These systems are computer programs intended to solve real-world problems, achieving the same level of accuracy as human experts. Knowledge engineering is the discipline that has evolved to support the whole process of specifying, developing and deploying knowledge-based systems (Schreiber et al., 2000)

Now there is a much wider interest in capturing and modelling knowledge and expertise. This has arisen because the importance of Knowledge Management (KM) is universally recognised by organisations large and small. There are many different characterizations of KM but the central assumption is that knowledge is a valuable asset that must be managed (Nonaka and Takeuchi, 1995; Stewart, 1997). What we are looking for in KM is a means to get the right knowledge to the right people at the right time and in the right form. These are difficult challenges many of them identical to those encountered when building early knowledge-based systems (Hayes-Roth, Waterman and Lenat, 1983). Acquiring, documenting, distributing, reusing and maintaining knowledge are all difficult and time-consuming tasks. We have argued elsewhere that the tools and techniques, methods and approaches of knowledge engineering are well suited to the KM enterprise (Milton et al., 1999)

This chapter will discuss the problem of knowledge elicitation for knowledge intensive systems in general. These systems range from classical knowledge-based systems through to structured intranets, from workflow support tools through to best practice guidelines. The content elicited from experts does not have to exist in an electronic format at all. However, increasingly the results of hard won knowledge elicitation and expertise modelling find their way into some form of digital system.

Knowledge elicitation comprises a set of techniques and methods that attempt to elicit an expert's knowledge through some form of direct interaction with that expert. The first section will review the nature and characteristics of this 'bottleneck' in system construction. We will then look at a range of methods and techniques for elicitation. Where appropriate we will describe their implementation in software. Methodologies for expertise modelling will be described and we will illustrate the kinds of knowledge that will be present in expert behaviour. We will consider the different types of expert that may be encountered and the attendant consequences for elicitation. Finally, we will consider the extent to which the burgeoning amount of content on the web is changing the way might think about aspects of the knowledge acquisition problem.

There is still no comprehensive theory of knowledge acquisition available. It remains an art as much as a science. It is not the purpose of this chapter to investigate the theoretical shortcomings of knowledge acquisition but to deliver practical advice and guidance on performing the process.

Knowledge Intensive Systems

In the early days of Artificial Intelligence much effort went into attempts to discover general principles of intelligent behaviour. Newell and Simon's (1963) General Problem Solver exemplified this approach. They were interested in uncovering a general problem solving strategy that could be used for any human task.

In the early 1970s this position was challenged. A new slogan came to prominence - 'in the knowledge lies the power'. A leading exponent of this view was Edward Feigenbaum of SRI. He observed that experts are experts by virtue of domain specific problem solving strategies together with a great deal of domain specific knowledge. Programs that attempted to implement detailed knowledge about tasks and the subjects to which they applied came resulted in the class of programs called Expert or Knowledge-Based Systems. These are now widely used and often quite invisible to the end user. The spelling and grammar checker that is being used to write this chapter owes its origins to knowledge-based systems technology. There are systems that look for patterns to detect credit card fraud, classify radar tracks, interpret patient vital signs and support in the design of aero engines.

There are a variety of ways in which expertise is encoding in run time systems. Many systems will incorporate some of rule based or object orientated representation. For a review of the major types of knowledge based system architecture and of different knowledge representation formalisms see Stefik, 1993.

The Problem of Elicitation

The people who build knowledge intensive systems are typically not people with a deep knowledge of the application domain. However, it is they who must gather the domain knowledge and then implement it in a form that the machine can use.

In the simplest case, one may be able to gather information from a variety of non-human resources: textbooks, technical manuals, case studies and so on. However, in most cases one needs actually to consult a practising expert. This may be because there isn't the documentation available, or because real expertise derives from practical experience in the domain, rather than from a reading of standard texts. The task of gathering information generally, from whatever source is called *knowledge acquisition*.(KA). The sub-task of gathering information from the expert is called *knowledge elicitation* (KE). In this chapter we will be concentrating on KE. Few knowledge intensive systems are ever built without recourse to experts at some stage. Those systems not informed by actual expert understanding and practice are often the poorer for it.

Many problems arise before elicitation of the detailed domain knowledge. We need to understand the purpose and requirements for any knowledge intensive system. Sometimes the failure is in formulating the role of the system, on other occasions it is a failure to appreciate what it is realistic to build. Systems can fail because no one has thought of the social and organisational problems that must be resolved in deploying a system. Very often the effort and resources required to build systems are underestimated: this occurs in both the development and maintenance of systems. A particularly nasty situation arises when one is expected to conjure up knowledge for areas in which no evidence of systematic practice exists at all. Here one is expected to provide theories for domains where there is no theory. Providing we can resolve these issues then we get down to KE in the expectation that it will be time well spent.

Two questions dominate in KE. How do we get experts to tell us, or else show us, what they do? How do we determine what constitutes their problem solving competence? This is a hard enough problem in itself but there are a variety of circumstances that contrive to make the problem even harder. Much of the power of human expertise lies in laid-down experience, gathered over a number of years, and represented as heuristics[1]. Often the expertise has become so routinised that experts no longer know what they do or why. In many cases the knowledge required to build a system is distributed across an organisation and in the heads of a number of experts. Experts do not always agree so there is the problem of reconciling conflicting or differing views.

There are obviously clear commercial reasons to try to make KE an effective process. We would like to be able to use techniques that will minimise the effort spent in gathering, transcribing and analysing an expert's knowledge. We would like to minimise the time spent with expensive and scarce experts. And, of course, we would like to maximise the yield of usable knowledge.

There are also sound engineering reasons why we would like to make KE a systematic process. We would like the procedures of KE to become common practice and conform to clear standards. This will help ensure that the results are robust. Robust methods are ones that can be used on various experts in a wide range of contexts by any competent knowledge engineer or KE practitioner. We also hope to make our techniques reliable. This will mean that different practitioners can apply them with the same expected utility. Placing elicitation on such a systematic footing will also be important in the development of methodologies that direct the process of specifying, constructing and maintaining systems.

We will begin by describing, in sufficient detail for the reader to apply them, examples of major KE methods. We will mention other techniques and where the reader can find out more about them. We will then review aspects of expertise and human information processing that are likely to directly affect the KE process. We will also indicate various software tools that implement or support some of these KE methods. We will describe the methodologies for acquisition and modelling expertise

---

[1]An heuristic is defined as a rule of thumb or generally proven method to obtain a result given particular information.

that are beginning to emerge. Finally, we will discuss the effect that the presence of evermore content on the web is having on the knowledge acquisition problem

Elicitation Techniques

The techniques we will describe are methods that we have found in our previous work to be both useful and complementary to one another. We can subdivide them into *natural* and *contrived* methods. The distinction is a simple one. A method is described as natural if it is one an expert might informally adopt when expressing or displaying expertise. Such techniques include interviews or observing actual problem solving. There are other methods we will describe in which the expert undertakes a contrived task. The task elicits expertise in ways that are not usually familiar to an expert. The first two categories of elicitation method are both natural under this definition and are varieties of interview and protocol analysis.

The Structured Interview

Almost everyone starts in KE by determining to use an interview. The interview is the most commonly used knowledge elicitation technique and takes many forms, from the completely *unstructured* interview to the formally-planned, *structured* interview.

The structured interview is a formal version of the interview in which the person eliciting the knowledge plans and directs the session. A significant benefit of the structured interview is that it provides structured transcripts that are easier to analyse than unstructured conversation. In reality the structured interview is a class of techniques (Hoffman et al., 1995).

The formal interview specified here constrains the expert-elicitor dialogue to the general principles of the domain. Experts do not work through a particular scenario extracted from the domain by the elicitor; rather experts generate their own scenarios as the interview progresses.

A template for such an interview is as follows.

1. Ask the expert to give a brief (10 minute) outline of the target task, including the following information:

   An outline of the task, including a description of the possible solutions or outcomes of the task;

   A description of the variables that affect the choice of solutions or outcomes;

   A list of major rules or procedures that connect the variables elicited to the solutions or outcomes.

2. Take each rule or procedure elicited in Stage 1, ask when it is appropriate and when it is not and if it is a procedure how to is preformed. The aim is to reveal the scope (generality and specificity) of each existing rule, and hopefully generate some new rules.

3. Repeat Stage 2 until it is clear that the expert will not produce any additional information.

A useful way of obtaining a domain overview (stage 1 of the structured interview) is to ask probe questions that relate to an individuals specific experience. It is also important in this technique to be specific about how to perform stage 2. We have found that it is helpful to constrain the elicitor's interventions to a specific set of *probes*, each with a specific function. Here is a list of probes (P) and functions (F) that can help in stages 1 & 2.

P1.1 Could you tell me about a typical case?
F1.1 Provides an overview of the domain tasks and concepts

P1.2 Can you tell me about the last case you encountered?
F1.2 Provides an instance based overview of the domain tasks and concepts

P2.1 Why would you do that?
F2.1 Converts an assertion into a rule

P2.2 How would you do that?
F2.2 Generates *lower order* rules

P2.3 When would you do that?
Is <the rule> always the case?
F2.3 Reveals the generality of the rule and may generate other rules

P2.4 What alternatives to <the prescribed action/decision> are there?
F2.4 Generates more rules

P2.5 What if it were not the case that <currently true condition>?
F2.5 Generates rules for when current condition does not apply

P2.6 Can you tell me more about <any subject already mentioned>
F2.6 Used to generate further dialogue if expert dries up

P2.7 Can you tell me about an unusual case you encountered/heard about from some other expert?
F2.7 Refines the knowledge to include rare cases and special procedures

The idea here is that the elicitor engages in a type of slot/filler dialogue. The provision of template questions about concepts, relations, attributes and values makes the elicitor's job very much easier. It also provides sharply focused transcripts that facilitate the process of extracting usable knowledge. Of course, there will be instances when none of the above probes are appropriate (such as the case when the elicitor wants the expert to clarify something). However, you should try to keep these interjections to a minimum. The point of specifying such a fixed set of linguistic probes is to constrain the expert to giving you all, and only, the information you want.

The sample of dialogue below is taken from a real interview of this kind. It is the transcript of an interview by a knowledge engineer (KE) with an expert (EX) in the domain of geological analysis[2].

KE What would you do at this stage?
EX I would look at the grain size of the hand specimen and see how fine it was
KE Why would you look at the grain size?
EX That will tell me if the rock has been formed near to the surface or deep inside the earth. The finer the grain size the faster it cooled. Coarse crystals indicate that the rock was cooling slowly + forming deeper down + we say its emplacement is plutonic + if it cooled near the surface its emplacement is volcanic.
KE Are there any alternatives to coarse and fine grain size?
EX There are glasses + you can't see any structure here because the rock cooled so fast
KE What would you look at next?
EX Colour is important + the lighter the rock the more acidic it is.
KE Why is a lighter rock more acidic?
EX Acidic rocks are higher in quartz and colour is a good indicator of quartz content – leucocratic or light things have a lot of quartz - melanocratic that is darker rocks have olivines and pyroxines.

This is quite a rich piece of dialogue. From this section of the interview alone we can extract numerous rules such as

IF grain size is large
THEN rock is plutonic

IF rock is leucocratic
THEN rock has high quartz content

Of course these rules may need refining in later elicitation sessions, but the text of the dialogue shows how the use of the specific probes has revealed a well-structured response from the expert[3].

Semi-Structured Interviews

Techniques exist to impose a lesser amount of structure on an interview. We mention two examples here. One of these is the Knowledge Acquisition Grid (LaFrance, 1987). This is a matrix of knowledge types and forms: examples of knowledge forms are *layouts* and *stories*; examples of question types are *grand tour* and *cross-checking*. A grand tour involves such things as distinguishing domain boundaries and the overall organization of goals; cross-checking involves the engineer attempting to validate the acquired knowledge by, for example, playing devil's advocate.

---

[2] In the transcripts we use the symbol + to represent a pause in the dialogue.

[3] In fact, a possible second-phase elicitation technique would be to present these rules back to the expert and ask about their truthfulness, scope and so forth.

Secondly, there is the teachback technique of (Johnson and Johnson, 1987). In this technique when formulates a representation of the knowledge that has been acquired in an interview. This is then 'taught back' to the expert, who can then check or, when necessary, amend the information.

Unstructured Interviews

Unstructured interviews have no agenda (or, at least, no *detailed* agenda) set either by the knowledge elicitor or by the expert. Of course, this does not mean that the elicitor has no goals for the interview, but it does mean that she has considerable scope for proceeding; there are few constraints and herein lie its advantages. Firstly, the approach can be used whenever one of the goals of the interview is to establish a rapport between the expert and the knowledge elicitor. There are no formal barriers to the discussion covering whatever material either participant sees fit. Secondly, one can get a broad view of the topic easily; the knowledge elicitor can 'fill in the gaps' in her own perceived knowledge of the domain. Thirdly, the expert can describe the domain in a way with which he is familiar, discussing topics that he considers important and ignoring those he considers uninteresting.

The disadvantages are clear enough. The lack of structure can lead to inefficiency. The expert may be unnecessarily verbose. He may concentrate on topics whose importance he exaggerates. The coverage of the domain may be patchy. The data acquired may be difficult to integrate, either because it does not form a coherent body of content, or because there are inconsistencies. This last will be an even more likely occurrence if the information provided by *several* experts is to be collated.

In all of the interview techniques mentioned so far (and in some of the other generic techniques as well) there exist a number of dangers that have become familiar to practitioners of knowledge elicitation.

One problem is that in an interview experts will only produce what they can verbalise. If there are non-verbalisable aspects to the domain, the interview will not recover them. It may be that the knowledge was never explicitly represented or articulated in terms of language (consider, for example, pattern recognition expertise). Then there is the situation where the knowledge was originally learnt explicitly in a propositional or language-like form. However, in the course of experience such knowledge can become routinised or automatised[4]. This can happen to such an extent that experts may regard the complex decisions they make as based only on hunches or intuitions. In actual fact, these decisions are based upon large amounts of remembered data and experience, and the continual application of that knowledge. In this situation they tend to give *black box* replies 'I don't know how I do that....', 'It is obviously the right thing to do....'.

Another problem arises from the observation that people (and experts in particular) often seek to justify their decisions in any way they can. It is a common experience of

---

[4]We often use a computing analogy to refer to this situation and speak of the expert as having *compiled* the knowledge.

the knowledge elicitor to get a perfectly valid decision from an expert, and then to be given a spurious justification as to why it was made and how it originated.

For these and other reasons one should always supplement interviews with additional elicitation methods. Elicitation ought always to consist of a programme of techniques and methods. This brings us on to consider another family of techniques much favoured by knowledge engineers.

Protocol Analysis

Protocol Analysis (PA) is a generic term for a number of different ways of performing some form of analysis of the expert(s) actually solving problems in the domain. In all cases the elicitor takes a record of what the expert does - preferably by video or audio tape - or at least by written notes. Transcripts or protocols are then made from these records and the elicitor tries to extract meaningful structure, rules and processes from the protocols.

We can distinguish two general types of PA -*on-line* and *off-line*. In on-line PA the expert is being recorded solving a problem, and concurrently a commentary is made. The nature of this commentary specifies two sub-types of the on-line method. The expert performing the task may be describing what they are doing as problem solving proceeds. This is called *self-report*. A variant on this is to have another expert provide a running commentary on what the expert performing the task is doing. This is called *shadowing*.

Off-line PA allows the expert(s) to comment retrospectively on the problem solving session - usually by being shown an audio-visual record of it. This may take the form of retrospective self-report by the expert who actually solved the problem. It could also be a critical retrospective report by other experts, or there could be group discussion of the protocol by a number of experts including its originator. In the case in which only a behavioural protocol is obtained then obviously some form of retrospective verbalisation of the problem-solving episode is required.

Before PA sessions can be held, a number of pre-conditions should be satisfied. The first of these is that the elicitor is sufficiently acquainted with the domain to understand the expert's tasks. Without this the elicitor may completely fail to record or take note of important parts of the expert's behaviour.

A second requirement is the careful selection of problems for PA. This sampling of problems is crucial. PA sessions may take a relatively long time, only a few problems can be addressed in any programme of acquisition (Shadbolt and Burton, 1989). Therefore, the selection of problems should be guided by how representative they are. Asking experts to sort problems into some form of order (Chi et al., 1981, 1982) may give an insight into the classification of types of problems and help in the selection of suitable problems for PA (see also the next two sections on concept sorts and laddering for methods that can be used to help structure a classification of types of problem).

A further condition for effective PA is that the expert(s) should not feel embarrassed about describing their expertise in detail. It is preferable for them to have experience in thinking aloud. Uninhibited thinking aloud has to be learned in the same way as talking to an audience. One or two short training sessions may be useful. In these training sessions a simple task can be used as an example. This puts the expert at ease and familiarises them with the task of talking about their problem solving.

Where a verbal or behavioural transcript has been obtained we next have to undertake its analysis. Analysis might include the encoding of the transcript into 'chunks' of knowledge (actions, assertions, propositions, key words, etc.), and should result in a rich domain representation with many elicited domain features together with a number of specified links between those features. The example below is from a self-report of an expert geologist. It is immediately apparent that protocols can be extremely dense sources of information. A very significant amount of work is required to analyse and structure the content in this very small fragment of a self report on one specimen.

> To start off with it's obviously a fairly coarse-grained rock ... and you've got some nice big orthoclase crystals in here - this is actually SHAP GRANITE - I know it just because everybody's seen SHAP GRANITE - or it's a very strong possibility that it's SHAP GRANITE ... it's a typical teaching specimen - as I say the obvious things are these very big orthoclase crystals pink colouration and you can certainly see some cleavage in some of them - you can certainly make out there are feldspar cleavages in there - it's a coarse-grained rock anyway, you can see the crystals nice and coarsely - these large porphyritic crystals - you can see, in the ground mass, you can see quartz - get some light on it (HOLDS SPECIMAN UP TO WINDOW) quartz, which is this fairly clear mineral you can actually look into it and see through it as opposed to calcite or feldspars where it's more cloudy - you can't actually see any good crystal faces on these cut sections - small flakes of biotite, black micacious looking - small plates, you can certainly see some on this specimen even without a hand lens.

There are a number of principles that can guide the protocol analysis. For example, analysis of the verbalization resulting in the protocol can distinguish between information that is attended to during problem-solving, and that which is used implicitly. A distinction can be made between information brought out of memory (such as a recollection of a similar problem solved in the past), and information that is produced 'on the spot' by inference. The knowledge chunks referred to above can be analysed by examining the expert's syntax, or the pauses he takes, or other linguistic cues. Syntactical categories (e.g. use of nouns, verbs, etc.) can help distinguish between domain features and problem-solving actions, etc.

In trying to decide when it is appropriate to use PA bear in mind that it is alleged that different KE techniques differentially elicit certain kinds of information (Hoffman et al, 1995). With PA it is claimed that the sorts of knowledge elicited include; the "when" and "how" of using specific knowledge. It can reveal the problem solving and reasoning strategies, evaluation procedures and evaluation criteria used by the expert, and procedural knowledge about how tasks and sub-tasks are decomposed. A PA gives you a complete episode of problem solving. It can be useful as a verification method to check that what people say is what they do. It can take you deep into a particular problem. However, it is intrinsically a narrow method since usually one can only run a relatively small number of problems from the domain.

When actually conducting a PA the following are a useful set of tips to help enhance its effectiveness. Present the problems and data in a realistic way. The way problems and data are presented should be as close as possible to a real situation. Transcribe the protocols as soon as possible, the meaning of many expressions is soon lost, particularly if the protocols are not recorded. In almost all cases an audio recording is sufficient, but video recordings have the advantage of containing additional and disambiguating information. Avoid long self-report sessions. Because of the need to perform a double task the process of thinking aloud is significantly more tiring for the expert, than being interviewed. This is one reason why shadowing is sometimes preferred. In general, the presence of the elicitor is required in a PA session. Although the elicitor adopts a background role, her very presence suggests a listener to the interviewee, and lends meaning to the talking aloud process. Therefore, comments on audibility, or even silence by the elicitor, are quite acceptable.

Protocol analyses share with the unstructured interview the problem that they may deliver unstructured transcripts that are hard to analyse. Moreover, they focus on particular problem cases and so the scope of the knowledge produced may be very restricted. It is difficult to derive general domain principles from a limited number of protocols. These are some of the practical disadvantages of protocol analysis. However, there are more subtle problems.

Two actions, which look exactly the same to the knowledge elicitor, may be very different in their extent and intent. For example, our geologist who performs a particular test to a specimen may apply that same test to another but with a quite different purpose. The knowledge elicitor simply does not know enough to discriminate the actions. The obverse to this problem can arise in shadowing and the retrospective analyses of protocols by experts. Here the expert(s) may simply wrongly attribute a set of considerations to an action after the event. This is analogous to the problems of misattribution in interviewing.

A particular problem with self-report, apart from being tiring, is the possibility that verbalisation may interfere with performance. The classic demonstration of this is for a driver to attend to all the actions involved in driving a car. If one consciously monitors such parameters as engine revs, current gear, speed, visibility, steering wheel position and so forth, the driving invariably gets worse. Such skill is shown to its best effect when performed automatically. This is also the case with certain types of decision making expertise. By asking the expert to verbalise, one is in some sense destroying the point of doing protocol analysis - to access procedural, real-world knowledge.

Having pointed to these disadvantages, it is also worth remembering that context is oftentimes important for memory - and hence for problem solving. For most non-verbalisable knowledge, and even for some verbalisable knowledge, it may be essential to observe the expert performing the task. For it may be that this is the only situation in which the expert is actually able to perform it.

Finally, when performing PA it is useful to have a set of conventions for the actual interpretation and analysis of the resultant data. Ericsson and Simon (1993) provide the classic exposition of protocol analysis although it is oriented towards cognitive psychology. Useful additional references are Kuipers and Kassirer (1983), Belkin, Brooks and Daniels (1987), McGraw and Harbison-Briggs (1989), Scott et al. (1991) and Firlej and Hellens (1991).

Critical Decision Method

This method contains elements of both interviewing and protocol analysis but in a context that stresses the examination of problem solving in natural decision making contexts (Zsambok and Klein, 1997). Klein and his colleagues developed a set of opening queries to stimulate recall of salient cases – cases that involved critical decisions (K1ein et al 1986). A set of probe questions that were designed to elicit specific, detailed information about the important cues, choice points, options, actions plans and the role of experience in decision making. A distinctive feature of this approach was that it seemed well suited to eliciting knowledge relating to highly dynamic situations where the requirement was to rapidly a situation and identify an effective and feasible course of action (Klein, 1993a, 1993b). Domains examined using the approach included acute clinical care, military planning, fire fighting and industri.al process control.

A CDM session is organised around an account of a specific incident from the expert's own experience. The expert is guided in the recall and recounting of the incident and its context. There then follow three information-gathering passes back through the incident. First a time line is built that verifies the points at which decisions are made. Second there is a phase of deepening that produces a more comprehensive and contextually rich account of the incident – focusing for example, on the cues used to recognise salient features of the incident. A final information sweep uses a "what if" approach to identify potential errors, alternative decision points, and expert/novice differences.

The table below contains a range of probe questions types with exemplars that we have found to be particularly useful in various phases of the CDM. There is no reason to use these questions exclusively for an individual phase although it is clear that the *options* and *choice* probe types are likely to feature substantially in the "what if" phase of information gathering

Probe Type	Probe Examples
Cues	What were you seeing, hearing, smelling?
Knowledge	What information did you use in making this decision? How was it obtained?
Analogues	Were you reminded of any previous incidents?
Scenarios	Does this case fir a standard or typical scenario? Does it fit a scenario you were trained to deal with?
Goals	What were your specific goals and objectives at the time?
Options	What other courses of action were considered or available?
Choice	How was this option selected/other options rejected? What rule

	was being followed?
Anticipation	Did you imagine the possible consequences of this action? Did you imagine the events that would unfold?
Experience	What specific training or experience was necessary or helpful in this decision? What more would have helped?
Decision making	How much time pressure was involved in making the decision? How long did it take to make the decision?
Aiding	What training, knowledge or information could have helped?
Situation assessment	If you were asked to describe the situation to a colleague at this point, have would you summarise the situation?
Errors	What mistakes are likely at this point? How might a novice have behaved differently?
Hypotheticals	If a key feature of the situation had been different, what differences would it have made in you decision?

Table 1 Sample CDM Probe Questions

A typical CDM session can last around 2 hours and depending on the domain more or less time might be spent on recollecting a rich complex incident whilst in another setting the majority of the effort is devoted to examining counterfactual situations. The CDM does have limitations. In distributed problem solving no one individual may handle more than one element of a task. They would never know whether their judgements or assessments were correct. In high workload environments we have observed that incidents and events can become merged. When responding to an opening query one sometimes sees an expert recount an incident but then become confused when asked for a time line or other details. Despite these shortcomings the style of interview and problem solving reflection provides a rich output from which the elicitor can extract important task relevant knowledge – a more detailed account of the method can be found in Hoffman et al 1998.

The techniques discussed so far are *natural* and intuitively easy to understand. Experts are used to expressing their knowledge in these sorts of ways. The techniques that follow are what we what we have termed *contrived* and permit the expression of knowledge in ways that are likely to be unfamiliar to the expert.

Concept Sorting

Concept sorting is a technique that is useful when we wish to uncover the different ways an expert sees relationships between a fixed set of concepts.

In the version we will present an expert is presented with a number of cards on each of which a concept word is printed. The cards are shuffled and the expert is asked to sort the cards into either a fixed number of piles of else to sort them into any number of piles the expert finds appropriate. This process is repeated many times.

Using this task one attempts to get multiple views of the structural organisation of knowledge by asking the expert to do the same task over and over again. Each time the expert sorts the cards he should create at least one pile that differs in some way from previous sorts. The expert should also provide a name or category label for each pile on each different sort.

Performing a card sort requires the elicitor to have some basic conception of the domain. Cards have to be made with the appropriate labels before the session. However, no great familiarity is required as the expert provides all the substantial knowledge in the process of the sort. We now provide an example from our geology domain to show the detailed mechanics of a sort.

The concepts printed on the cards were the names of igneous rocks drawn from a structured interview with the expert. He had described 18 rock types

1	adamellite	10	granite
2	andesite	11	lherzolite
3	basalt	12	microgranite
4	dacite	13	peridotite
5	diorite	14	picrite basalt
6	dolerite	15	rhyodacite
7	dunite	16	rhyolite
8	gabbro	17	syenite
9	granodiorite	18	trachyte

The expert was shown possible ways of sorting cards in a *toy* domain, as part of the briefing session, and then asked to sort the real elements in the same way.

The dimensions/piles which the expert used for the various sorts were as follows:

Sort 1: grain size	Piles 1=coarse, 2=medium, 3=fine
Sort 2: colour	Piles 1=melanocratic, 2=mesocratic, 3=leucocratic
Sort 3: emplacement	Piles 1=intrusive, 2=extrusive
Sort 4: presence of olivine	Piles 1=always, 2=possibly, 3=never
Sort 5: presence of quartz	Piles 1=always, 2=possibly, 3=never
Sort 6: % of silica	Piles 1= >68%, 2= <68%, 3= about 68%
Sort 7: density	Piles 1=v.light, 2=light, 3=medium, 4=dense,5=v.dense

Here is a table showing the pile of each sort for each element. You will see that many of the elements are distinguishable from one another - even with these few sorts.

	SORT						
ROCK	1	2	3	4	5	6	7
1	1	3	1	1	1	2	1
2	3	2	2	3	2	2	3
3	3	2	2	2	2	2	4
4	3	2	2	3	2	3	2
5	1	2	1	3	2	2	3

6	2	1	1	2	2	2	4
7	1	1	1	1	3	2	5
8	1	2	1	2	2	2	4
9	1	3	1	3	1	3	1
10	1	3	1	3	1	1	1
11	1	1	1	1	3	2	5
12	2	3	1	3	1	1	1
13	1	1	1	1	3	2	5
14	3	1	2	1	3	2	4
15	3	3	2	3	1	1	2
16	1	3	2	3	1	1	1
17	1	3	1	3	1	2	3
18	3	3	2	3	2	2	2

Table 1: Tabulated results from the card sort

Using this information we can attempt to extract decision rules directly. An example of a rule extracted from the sorting is :

IF	the grain size is fine	(sort 1/pile 3)
AND	the colour is mesocratic	(sort 2/pile 2)
AND	its emplacement is extrusive	(sort 3/pile 2)
AND	it does NOT contain olivine	(sort 4/pile 3)
AND	may possibly contain quartz	(sort 5/ pile 2)
AND	it contains less than 68% silica	(sort 6/ pile 2)
AND	its density is medium	(sort 7/ pile 3)
THEN	the rock is andesite	(outcome 2)

As you can see from the example such sorts produce long and cumbersome rules. In fact many of the clauses may be redundant - once you have established that the grain size is small, then it is going to be an extrusive rock.

However, the utility of this technique does not reside solely in the production of decision rules. We can use it, as we have said, to explore the general inter-relationships between concepts in the domain. We are trying to make explicit the implicit structure that experts impose on their expertise.

When using any of these KE methods knowledge elicitors should beware a type of semantic mindset whereby the expert or elicitor focuses on only one type of knowledge element. To derive the full benefit of a KE method one should play many variations on the theme. For example, in concept sorting the cards can name knowledge elements of any type not just objects in a domain. The cards might name tasks, goals, actions, resources etc. The restriction is that in any sorting session the cards should be of the same knowledge type.

Variants of the simple sort are different forms of *hierarchical* sort. One such version is to ask the expert to proceed by producing first two piles, on the second sort three, then four and so on. Finally we ask if any two piles have anything in common. If so you have isolated a higher order concept that can be used as a basis for future elicitation.

The advantages of concept sorting can be characterised as follows. It is fast to apply and easy to analyse. It forces an explicit format on the constructs that are underlie an experts understanding. In fact it is often instructive to the expert. A sort can lead the expert to see structure that he himself has not consciously articulated before. Finally, in domains where the concepts are perceptual in nature (i.e. x-rays, layouts and pictures of various kinds) then the cards can be used as a means of presenting these images and attempting to elicit names for the categories and relationships that might link them.

There are, of course, features to be wary of with this sort of technique. Experts can often confound dimensions by not consistently applying the same semantic distinctions throughout an elicitation session. Alternatively, they may over simplify the categorisation of elements, missing out important caveats.

An important tip with all of the contrived techniques we are reviewing is to always audiotape these sessions. An expert makes many asides, comments and qualifications in the course of sorting ranking and so on. In fact one may choose to use the contrived methods as a means to carry out auxiliary structured interviews. The structure this time is centred on the activity of the technique.

It is worth noting that we have found (Schweikert et al., 1987) an expert's own opinion of the worth of a technique no guide to its real value. In methods such as sorting we have a situation in which the expert is trying to demonstrate expertise in a non-natural or contrived manner. He might be quite used to chatting about his field of expertise, but sorting is different and experts may be suspicious of it. Experts may in fact feel they are performing badly with such methods. However, on analysis one finds that the yield of knowledge is as good and sometimes better than for non-contrived techniques (Shadbolt and Burton, 1990).

Laddered Grids

Another somewhat contrived technique that you will need to explain carefully to the expert before starting. The expert and elicitor construct a graphical representation of the domain in terms of the relations between domain or problem solving elements. The result is a qualitative, two-dimensional graph where nodes are connected by labelled arcs. No extra elicitation method is used here, expert and elicitor construct the graph together by negotiation.

In using the technique the elicitor enters the conceptual map at some point and then attempts to move around it with the expert. A formal specification of how we use the technique is shown below together with an example of its use.

Start the expert off with a seed item

Move around the domain map using the following prompts

To move DOWN the expert's domain knowledge:

Can you give examples of <ITEM>?

To move ACROSS the expert's domain knowledge:

What alternative examples of <CLASS> are there to <ITEM>?

To move UP the expert's domain knowledge:

What have <SAME LEVEL ITEMS> got in common?

What are <SAME LEVEL ITEMS> examples of?

To elicit essential properties of an item:

How can you tell it is <ITEM> ?

To discriminate items:

What is the key difference between <ITEM 1> and <ITEM 2>?

The elicitor may move around the knowledge map in any order which seems appropriate or convenient. As the session progresses, the elicitor keeps track of the elicited knowledge by drawing up a network on a large piece of paper or if computer supported via some other graphical characterisation. This representation allows the elicitor to make decisions (or ask questions) about what constitutes higher or lower order elements in the domain, what differences exist between elements in the network. In order to give the reader the flavour of the technique, there follows an extract from a laddered grid elicitation session. Once again, the knowledge domain is geology.

KE	So how could you tell something was dacite?
EX	Well + examine the fresh surface and the weathered surfaces first + looking at grainsize, the relationship between the grains
KE	Can I just stop you there. What type of grain size is it?
EX	Coarse, medium, fine grain, oh, you want me to actually say what dacite is?
KE	The grain, in dacite what would it be?
EX	Er + medium grained.
KE	Medium grained, right. So can you give me other examples of medium grained rocks?
EX	Medium grained rocks + dolerite... Granodiorite as well... And we'll stay with that.
KE	Right, erm, what alternative is there to a medium grained rock?
EX	Well, you can have a coarse grained one or a fine grained one, those are sort of the three major ones.
KE	Right, can you give me examples of coarse grained rocks?

EX Er, gabbro, granite... hmm, yeah, those two.
KE And any examples of fine-grained rocks?
EX Er, basalt... er andesite, trachyte...microgranite as well.
KE Right, erm so. What about others
EX Some of these are sort of a metamorphic ones where you're going to get large grains in a fine-grained matrix. There are phenocrysts in them, that's what we call the large grains
KE Is, is there a word for that kind of texture or?
EX Porphyritic mixture
KE Can you give me the examples of the porphyritics...
EX Nepheline-syenite, oh and Kentallenite
KE How would you go about telling the difference between dolerite and granodiorite? What is the key difference?
EX Whether it's got quartz or hasn't got quartz or the percentage of quartz present will define whether it's an acidic rock or a basic rock, basic not having any quartz in it at all, and then er if there's a low amount, that's going to be an intermediate rock
KE Which, which are the intermediate?
EX Dacite + you've got high quartz are granite, microgranite, and andesite, and no quartz gabbro, basalt, dolerite and trachyte, intermediate dacite.

In the course of this laddered grid interview the elicitor drew up a hierarchical representation of the domain as shown in Figure 1. This is only one of a number of representations that could have been made. In this case the concepts of fine, medium and coarse grained rocks have been understood to be classes of rock type. Similarly the concept of an acidic, intermediate or basic rock has been treated as a class of rock type. However, the grain size and acidity (amount of quartz) could have been represented as properties of the particular rock types. These sorts of representational decisions abound in any knowledge elicitation exercise. Decisions as to which are the most appropriate representations come down to

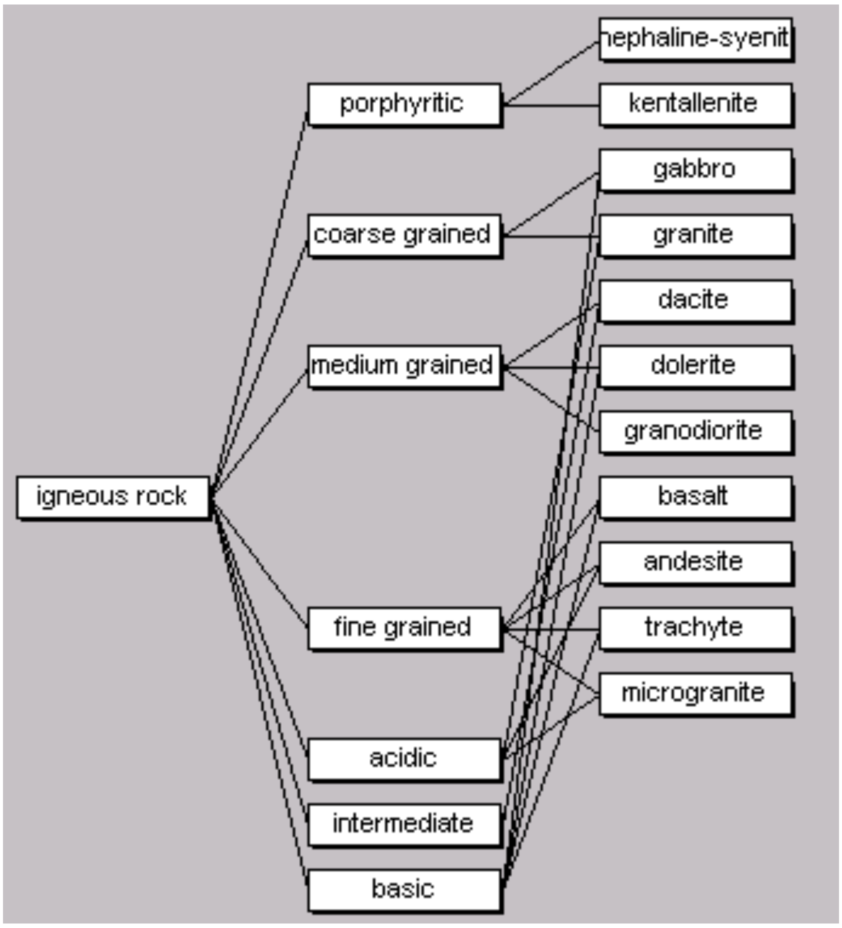

Figure 1 Laddered Grid in the geology domain

This hierarchy gives rise to the following set of rules that could be included in the knowledge base of a knowledge intensive system for geological rock classification.

IF	the rock is of medium grain size
AND	the rock is intermediate
THEN	the rock may be dacite

IF	the rock is of coarse grain size
AND	the rock is acidic
THEN	the rock may be granite

IF	the rock is of coarse grain size
AND	the rock is basic
THEN	the rock may be gabbro

As with the previous contrived method it is important to keep an audio record of the session for future review or transcription. Laddering is an excellent way of carrying out a structured interview. Also it is a techniques that can be used on a variety of knowledge types; objects, actions, tasks, goals, etc..

We have found that this form of knowledge elicitation is very powerful for structured domains. As with other contrived techniques we have found that whilst an expert may think this technique is revealing little of interest, subsequent analysis provides good quality content.

The Limited Information Task

A technique which can prove an excellent complement to the methods already outlined does not provide a spatial representation of the domain, but rather a set of hints or suggestions which may prove useful in the construction of knowledge intensive systems is a technique called the *limited information task* (Hoffman et al, 1995) or *20 questions* (Grover, 1983). The expert is provided with little or no information about a particular problem to be solved. The expert must then ask the elicitor for specific information that will be required to solve the problem. The information that is requested, along with the order in which it is requested, provides the elicitor with an insight into the expert's problem solving strategy. One difficulty with this method is that the elicitor needs a good understanding of the domain in order to make sense of the experts' questions, and to provide meaningful responses. The elicitor should have forearmed themselves with a problem from the domain together with a *crib sheet* of appropriate responses to the questions.

In one of the versions of the limited information task that we use we tell the expert that the elicitor has a scenario in mind and the expert must determine what it is. The scenario might represent a problem, a solution or a problem context. The expert is told that they may ask the elicitor for more information, though what the elicitor gives back is terse and does not go much beyond what was asked for in the question. The expert may be asked to explain why each of the questions was asked.

An example of the kind of interaction produced by this technique is shown below. Here the problem domain is in the construction of lighting systems for the inspection of industrial products and processes.

EX: Is this in the manufacturing industry?
KE: Yes
EX: So we've ruled out things like fruit, vegetables, cows?
KE: Yes
EX: Is it the metal industry?
KE: The material is wood
EX: So we could be dealing with a large object here like a chair or table
KE: The object is large
EX: It's likely to be a 3-D object, you've got to pick it up and turn it over
KE: That's right
EX: So what I need now are the dimensions of this object in terms of the cube that will enclose it
KE: It would have similar dimensions to the table top
EX: Do I inspect one surface or all the surfaces?
KE: All of them
EX: Is the inspector looking for one or many faults?
KE: One particular fault

EX: Can you describe it for me?
KE: It's pencil marks about half an inch long
EX: What colour is the wood?
KE: Dark unfinished wood
EX: We've got a contrast problem here. At this point I'd go and look at the job + to see if the graphite pencil marks reflect light + sometimes it does, but it depends on the wood + if it does you can select the light to increase the contrast between the fault and the background

:

:

:

EX: I'd be doing this in three phases: first a general lighting, then specific for surface lighting, and then some directional light [expert then gives technical specifications for these types of light]

This interview gives us an interesting insight into the natural line of enquiry of an expert in this domain. Often traditional knowledge-based systems gather the right data but the order in which it is gathered and used can be very different from how an expert works. This can decrease the acceptability of any implemented system if other experts are to use it, and it also has consequences for the intelligibility of any explanations the system offers in terms of a retrace of its steps to a solution.

It will be seen that we can once again extract decision rules directly from the dialogue:

IF	fault colour is black
AND	object colour is dark
THEN	contrast is a problem

The drawbacks to this technique are that the elicitor needs to have constructed plausible scenarios and the elicitor has to be able to cope with questions asked of him. The experts themselves are sometimes uncomfortable with this technique; this may well have to do with the fact that, as with other contrived techniques, it is not a natural means of manifesting expertise. Whilst a few scenarios may reveal some of the general rules in a domain the elicitation is very case specific. In order to get a broad range of knowledge for a sweep of situations many scenarios would need to be constructed and used.

An interesting variation on this method is a form of telephone consultancy. Here we take two domain experts and place them at opposite ends of a table and ask them to imagine that one is a "client" who is ringing up the other, a "consultant", to ask for advice concerning a particular problem. They then engage in a conversation in which the "consultant" tries to elicit the nature and context of the problem, and finally attempts to offer appropriate advice. In this variation of the limited information task you can rely on one of the experts to generate interesting cases. In addition, the expert "role playing" as the client can provide appropriate responses to the "consultant's" enquiries. The only drawback is that sometimes expert's construct extremely difficult cases for each other in order to test each other’s mettle!

A Taxonomy of KE Techniques

We have sampled some of the major approaches to elicitation and where appropriate given a detailed description of techniques that are likely to be of use. There are many variants on the methods we have described. Below we have provided a taxonomy of methods with which we are familiar together with a primary reference for each one.

- Non-contrived
  - Interviews
    - Structured
      - Fixed Probe (Shadbolt and Burton, this volume)
      - Focused Interviews (Hart, 1986, Clayton et al 1991)
      - Forward Scenario Simulation (Grover, 1983)
      - Critical Decision Method (Klein et al. 1998)
    - Semi-Structured
      - Knowledge Acquisition Grid (LaFrance, 1987)
      - Teach Back (Johnson & Johnson, 1987)
    - Unstructured (Weis & Kulikowski, 1984)
  - Protocol Analysis
    - Verbal
      - On line (Johnson, Zualkerman and Garber, 1987)
      - Off line (Elstein, Shulman and Sprafka, 1978)
      - Shadowing (Clarke, 1987)
    - Behavioural (Ericsson and Simon, 1984)
- Contrived
  - Conceptual Mapping
    - Sorting and Rating (Gammack, 1987)
    - Repertory Grid (Shaw and Gaines 1987)
    - Pathfinder (Schvaneveldt et al. 1985)
  - Goal Decomposition
    - Laddered Grid (Hinkle, 1965)
    - Limited-Information Task (Grover, 1983, Hoffman, 1987)

Table 2 A taxonomy of elicitation methods

Having discussed the principle methods of elicitation we should spend a little time reflecting on the nature of two other major components of the KE enterprise, namely - the experts and the expertise they possess.

On Experts

Experts come in all shapes and sizes. Ignoring the nature of your expert is another potential pitfall in KE. A coarse guide to a typology of experts might make the issues clearer. Let us take three categories we shall refer to as *academics*, *practitioners*, *samurai* (in practice experts may embody elements of all three types). Each of these types of expert differs along a number of dimensions. These include; the outcome of their expert deliberations, the problem solving environment they work in, the state of the knowledge they possess (both its internal structure and its external manifestation), their status and responsibilities, their source of information, the nature of their training.

How are we to tell these different types of expert apart when we encounter them? The academic type regards their domain as having a logically organised structure. Generalisations over the laws and behaviour of the domain are important to them. Theoretical understanding is prized. Part of the function of such experts may be to explicate, clarify and teach others. Thus they talk a lot about their domains. They may feel an obligation to present a consistent story both for pedagogic and professional reasons. Their knowledge is likely to be well structured and accessible. These experts may suppose that the outcome of their deliberations should be the correct solution of a problem. They believe that the problem can be solved by the appropriate application of theory. They may, however, be remote from every day problem solving.

The practitioner class on the other hand are engaged in constant day-to-day problem solving in their domain. For them specific problems and events are the reality. Their practice may often be implicit and what they desire as an outcome is a decision that works within the constraints and resource limitations in which they are working. It may be that the generalised theory of the academic is poorly represented in and articulated by the practitioner. For the practitioner heuristics may dominate and theory is sometimes thin on the ground.

The samurai is a pure performance expert - their only reality is the performance of action to secure an optimal performance. Practice is often the only training and responses are often automatic.

One can see this sort of division in any complex domain. Consider for example medical domains where we have professors of the subject, busy doctors working the wards, and medical ancillary staff performing many important but repetitive clinical activities.

The knowledge elicitor must be alert to these differences because the various types of expert will perform very differently in KE situations. The academic will be concerned to demonstrate mastery of the theory. They will devote much effort to characterising the scope and limitations of the domain theory. Practitioners, on the other hand, are driven by the cases they are solving from day to day. They have often *compiled* or *routinised* any declarative descriptions of the theory that supposedly underlies their problem solving. The performance samurai will more often than not turn any KE interaction into a concrete performance of the task - simply exhibiting their skill.

But there is more to say about the nature of experts and this is rooted in general principles of human information processing[5]. Psychology has demonstrated the limitations, biases and prejudices that pervade all human decision-making - expert or novice. To illustrate consider the following facts, all potentially crucial to the enterprise of KE.

It has been shown repeatedly that the context in which one encodes information is the best one for recall. It is possible then, that experts may not have access to the same information when in a KE interview, as they do when actually performing the task. So

---

[5]An excellent review of the psychology of expertise is Chi et al (1988) and a fascinating glimpse into the constituents of some aspects of expertise can be found in Ericsson (1996)

there are good psychological reasons to use techniques that involve observing the expert actually solving problems in the context in which they normally work. In short, protocol analysis techniques may be necessary, but will not be sufficient for effective knowledge elicitation.

Consider now the issue of biases in human cognition. One well-known problem is that humans are poor at manipulating uncertain or probabilistic evidence. This may be important in KE for those domains that require a representation of uncertainty. Consider the rule:

IF	the engine will not turn over
AND	the lights do not come on
THEN	the battery is flat with probability X

This seems like a reasonable rule, but what is the value of X, should it be 0.9, 0.95, 0.79? The value that is finally decided upon could have important consequences for the working of any knowledge intensive system, but it is very difficult to decide upon it in the first place. Medical diagnosis is a domain full of such probabilistic rules. However, even expert physicians cannot accurately assess probability values in their own domains of expertise.

In fact there are a number of documented biases in human cognition that lie at the heart of this problem (see for example the classic work of Kahneman, Slovic and Tversky, 1982). People are known to undervalue prior probabilities, to use the ends and middle of the probability scale rather than the full range, and to *anchor* their responses around an initial guess. Cleaves (1987) lists a number of cognitive biases likely to be found in knowledge elicitation, and makes suggestions about how to avoid them. Faced with these difficulties many knowledge elicitors prefer to avoid the use of uncertainty wherever possible.

Cognitive bias is not limited to the manipulation of probability. A series of experiments has shown that systematic patterns of error occur across a number of apparently simple logical operations. For example, *Modus Tollens* states that if 'A implies B' is true, and 'not B' is true, then 'not A' must be true. However people, whether expert in a domain or not, make errors on this rule. This is in part due to an inability to reason with contrapositive statements. Also in part it depends on what A and B actually represent. In other words, they are affected by the content. This means that one cannot rely on the veracity of experts' (or indeed anyone's) reasoning.

All this evidence suggests that human reasoning, memory and the representation of knowledge is rather more subtle than might be thought at first sight. The knowledge engineer should be alert to some of the basic findings emanating from cognitive psychology. Whilst no text is perfect as a review of bias in problem solving the book by Meyer and Booker (1991) is reasonably comprehensive.

On Expertise

Clearly the expertise embodied by experts is not of a homogeneous type (Feltovich et al., 1997). In constructing any knowledge intensive system it is likely that very different types of knowledge will be uncovered which will have very different roles in the system.

There are a number of analyses available of the epistemology of expertise. Our analysis is based to a large extent on that of CommonKADS (Schreiber et al, 2000).

Firstly, we can distinguish what is called domain level knowledge. This term is being used in the narrow sense of knowledge that describes the concepts and elements in the domain and relations between them. This sort of knowledge is sometimes called declarative, it describes what is known about things in the domain. The propositions below can all be seen as domain level knowledge in this sense.

Granite is a coarse grained rock
Andesite has a high quartz content

Extract 1 Domain Knowledge from an analysis of a laddered grid obtained from an expert geologist

There is also knowledge and expertise that has to do with what we might call the inference level. This is knowledge about how the components of expertise are to be organised and used in the overall system. It tells us the type of inferences that will be made and what role knowledge will play in those inferences. This is quite a high level description of expert behaviour and may often be implicit in expert practice. The following is a description of knowledge about part of an inference level structure called systematic diagnosis.

> To perform systematic diagnosis we will have knowledge about a complaint, and knowledge about observables from the patient or object. We select some aspect of the complaint and using a model of how the system should be performing normally we look to see if a particular parameter of the system is within normal bounds.

Extract 2 Analysis of verbal and behavioural protocols obtained from an expert in abdominal pain

Another type of expert knowledge is the task level. This is sometimes called procedural knowledge. This is knowledge to do with how goals and sub-goals, tasks and sub-tasks should be performed. Thus in a classification task there may exist a number of tasks to perform in a particular order so as to utilise the domain level knowledge appropriately. This type of knowledge is present in the following extract.

> First of all perform a general inspection of the object. Next examine the sample with a hand lens. Next use a prepared thin-section and examine that under a cross-polarising microscope.

Extract 3 Analysis of a verbal protocol obtained from an expert geologist

Finally, there is a level of expert knowledge referred to as strategic knowledge. This is information that monitors and controls the overall problem solving. This can have to do with the way resources are used. What to do if the proposed solution fails or is found to be inappropriate in some way. What to do when faced with incomplete or insufficient data. Such information is contained in the following extract from an interview.

> If I had time I would always check the disc head alignments. If its a BRAND X machine I'd always check that because they are notorious for going wrong.

Extract 4 Part of a structured interview transcript obtained from an expert computer technician

Any field of expertise is likely to contain these various sorts of knowledge to greater or lesser extents. At any particular knowledge level the information may be explicit or implicit in an experts' behaviour. Thus in some domains the experts may have no real notion of the strategic knowledge they are following whilst in others this knowledge is very much in the forefront of their deliberations. Also, of course, the requirements on any knowledge intensive system about how far it needs to implement these various levels will vary. It is almost universally acknowledged that significant reasoning about problem domains requires more than just modelling simple relationships between concepts in the domains. It may require causal models of how objects influence and affect one another, models of the processes in which objects participate. This is a hard problem. And often the limitations of implementation technologies first means that sophisticated domain models cannot be supported.

This brings us to a final important feature of KE. Since knowledge elicitation is such a time consuming and expensive business, not all of whose results can be immediately used, there is an increasing interest in developing ways of storing, archiving and retrieving knowledge that makes the best use of the elicitation investment (Neches et al., 1991). The key to this lies in a change in our way of thinking about the content of knowledge-based systems. This has already been outlined earlier in this section. It is called the knowledge level view and was originally conceived by Alan Newell (1982).

Managing knowledge in this way requires using an expressive and unambiguous intermediate representation of the knowledge to be stored. A number of candidates exist for this including graphical and language oriented representations – many of these draw inspiration from knowledge representation languages developed in AI (Young and Gammack, 1987; Rich and Knight, 1991; Sowa, 1999; Norvig and Russell, 2003). One of the major deliverables in many projects we have worked on has not been any implemented system but a set of knowledge documents that describe in a structured way the knowledge in a particular domain. In these cases we use CommonKADS as a modelling and representational standard (Schrieber et al, 2000).

CommanKADs embodies the knowledge level thesis put forward by Bill Clancey in his classic 1985 paper on heuristic classification. The structure, shown in Figure 2, was the result of a rational reconstruction of a number of existing knowledge intensive systems. His claim was that the knowledge bases of many systems were for the most part undifferentiated. The knowledge bases of these systems had been built with little regard as to how the knowledge was used. His analysis uncovered what he saw as an important type of problem solving system. Not all systems would contain and use knowledge in this way. Not all systems would be examples of heuristic classification, but some important ones were.

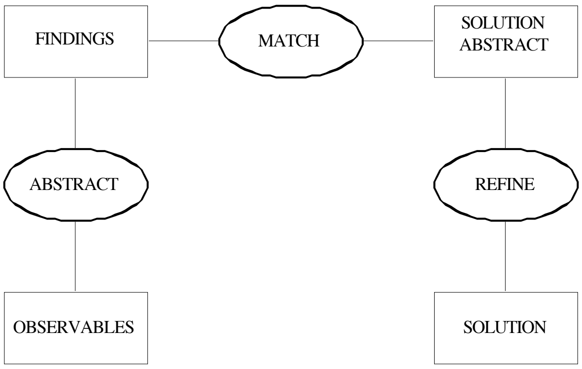

Figure 2 Heuristic Classification

One system which Clancey characterised as heuristic classification was the classic MYCIN medical knowledge-based system (Shortliffe, 1979). We shall illustrate his approach using MYCIN-like knowledge. Figure 2 is to be understood as a structure at what we have termed earlier as the inference level. It tells us what kinds of inferences are performed in this domain and the type of knowledge used by these inferences. The rectangles should be seen as types of data and the ellipses as types of inferences.

Let us take the left hand side of this structure that contains a process called abstraction. This is the process by which observations or data are transformed into abstract observations or findings. The process of abstraction can be realised using a number of methods. One of these is qualitative abstraction. Examples are shown below – in this example we move from quantitative observations to qualitative findings.

if	patient has white blood cell count < 2500
then	patient has low white blood cell count

if	patient has temperature > 101
then	patient has fever

What we have provided above is the actual domain level knowledge that plays the inference layer role of abstraction. Much of the knowledge in MYCIN's knowledge base as to do with this type of knowledge level processing - the process of abstraction, moving from a quantitative description of data to a qualitative one.

The top part of Figure 2 involves inferences from findings to abstract solutions. This type of inference is called match - and is understood as a type of association knowledge. An example of such knowledge might be a rule such as:

if	patient has fever
and	patient has low white blood cell count
then	patient has gram negative infection

The concept of gram negative infection might be viewed as a diagnosis or solution but it is not a very specific one. What sort of gram negative infection is it? The right hand side of the heuristic classification structure deals with inference types that refine general to specific solutions. Such knowledge might consist of hierarchical typologies containing knowledge such as

gram negative infection
has sub types
e. coli infection
:
:

To establish a particular infection the system is likely to use knowledge that discriminates the sub-types.

Notice that in this account although we have talked about the inference structure from left to right no explicit control knowledge has been given. We might have stipulated that the system start with patient data and reason forward to a possible solution. We might have stated that the system should hypothesise a solution and see if there was evidence from the patient's condition to support the hypothesis. Or else a mixture of these ways of moving around the structure of Figure 2 might have been adopted. This additional knowledge is, of course, the task layer we mentioned earlier. Sometimes the standard method of moving around the inference structure is modified. This might arise if when a particular disease is suspected then one immediately looks for a particular piece of supporting observational data. Such knowledge comprises the strategic layer.

What we have described was exemplified via a MYCIN-like example. But heuristic classification as a type of problem solving might apply to many domains; financial assessment of an individual's credit worthiness, the likelihood of finding a mineral resource at a particular location, classifying a particular work place setting as conforming to a particular health and safety standard etc.. It is this generality that is the power of these knowledge level approaches. If we know what kind of application we are building we can use the models to indicate the type of knowledge we need to acquire, how we might structure the knowledge base, how we can archive and index knowledge for future use.

These knowledge level models have formed the basis for a number of important methodologies that aim to support the knowledge engineering process (Shadbolt and O'Hara, 1997). A similar attempt to exploit structured templates can be found in modern approaches to designing software – for example the work on Design Patterns (Gamma et al., 1995) and Object Oriented Design (Booch, 1993). The principles from OOD are ones that can be usefully adopted in any knowledge modelling exercise. In particular the concentration on acquiring hierarchical descriptions of a domain in the form of class hierarchies of the sort we see in Figure 1. OOD stresses the importance of associating with each class the necessary properties to distinguish it from other objects. It also holds that the classes should represent the most general levels of abstraction consistent with discriminating between objects.

There is a growing need to standardize, share, and exchange knowledge descriptions in all application areas and across a wide range of individuals and organizations. For example, efforts are underway to build “knowledge-rich” thesauri to define the relevant terms in diverse fields such as medicine (Humphreys et al 1998), genetics (Gene Ontology Consortium, 2000) and art Petersen, 1994) – but there are also attempts to provide such structured resources for general terms in language, for example WordNet (Fellbaum, 1998). An organizing principal in these thesauri is the *subsumption* or *is-a* relation – but there are others such as *part-of*. Recently, computer scientists seeking to promote the exchange of knowledge between machines and humans have promoted the use of ontologies. These contain an explicit description of the semantics (“meaning”) of the types introduced. Tools and methods are now becoming available to support the modeling of ontologies (Noy et al., 2001). The construction of ontologies will be an important new application context for knowledge elicitation techniques.

Methodologies and Programmes of KE

We turn next to the question as to how KE techniques should be assembled to form a programme of acquisition and when we should use the various techniques. The choice may depend on the characteristics of the domain, of the expert, and of the required system. Furthermore, it is clear that some techniques are going to be more costly in terms of time with the expert, or else the effort required for subsequent analysis of transcripts.

There are a number of articles and books available on 'how to do knowledge elicitation'. These often contain advice of the most general kind, and emphasise the pragmatic considerations of knowledge intensive system development. General reviews can be found in Welbank (1983), Hoffman (1987), Kidd (1977), Hart (1986) McGraw and Harbison-Briggs (1989), Firlej and Hellens (1991) and Clayton et al (1991). While these reviews are based on experience of the general kind, there have also been a number of attempts to make formal recommendations.

Knowledge engineers have developed a number of principles that form the basis for the techniques and tools used for knowledge acquisition and modeling. Moreover, there are a number of assumptions in much of this area that are worth making explicit.

Broad repertoire of techniques: There is much evidence to suggest that different techniques can be more or less efficient in the types of knowledge they can elicit (Burton et al., 1987; 1988), the so-called differential access hypothesis (Hoffman, Shadbolt, Burton & Klein, 1995). Hence, to efficiently acquire the knowledge in a domain often requires a range of techniques.

Acquisition as Modelling: Traditionally, knowledge engineering was viewed as a process of “extracting” or “mining from the expert’s head” and transporting it in computational form to a machine. This has turned out to be a crude and rather naive view. Today, knowledge engineering is approached as a modelling activity. *A model is a purposeful abstraction of some part of reality*.

The knowledge-level principle: In knowledge modelling, first concentrate on the conceptual structure of knowledge, and leave the programming details for later. Many software developers have an understandable tendency to take the computer system as the dominant reference point in their analysis and design activities. But there are two important reference points: the computational artefact to be built, but most importantly, there is the human side.

Knowledge structures: Knowledge has a stable internal structure that is analyzable by distinguishing specific knowledge types and roles. It goes without saying that knowledge, reasoning, and problem-solving are extremely rich phenomena. Knowledge may be complex, but it is not chaotic: knowledge appears to have a rather stable internal structure.

Evolutionary Development: A knowledge project must be managed by learning from your experiences in a controlled "spiral" way. The development of simple or very well-known types of information systems usually proceeds along a fixed management route. This is especially clear in the so-called waterfall model of systems development.

As we have already noted the most thorough attempt to integrate KE procedure is provided by CommonKADS - Knowledge Acquisition and Domain Structuring (Schrieber et al 20000 for an overview of CommonKADS; Wielinga et al., 1992 for its roots). KADS embodies a number of principles for the elicitation of knowledge and construction of a system. These principles are:

1. The knowledge and expertise should be analysed before the design and implementation starts.
2. The analysis should be model-driven as early as possible.
3. The content of the model should be expressed at the knowledge level.
4. The analysis should include the functionality of the prospective system.
5. The analysis should proceed in an incremental way
6. New data should be elicited only when previously collected data have been analysed.
7. Collected data and interpretations should be documented.

Principle 1 is quite straightforward and requires no further explanation. Principle 2 requires that one should being to bear a model of how the knowledge is structured early on in the process, and use it to interpret subsequent data. Principle 3 means that one should use an appropriate intermediate level knowledge representation device, and try to characterise knowledge in terms of its use and functioning. Principle 4 is a reminder that a complete analysis includes an understanding of how the system is to work - e.g. who will use it, and in what situation. One cannot gain a full understanding of the problem simply by trying to map out an expert's knowledge without regard to how it will be used. Principle 5 emphasises the fact that there is a wide variety of related topics within a domain. This means that construction of a model should be 'breadth-first', embodying all aspects at once, rather than attempting fully to represent one sub-part after another. Principles 6 and 7 are once again straightforward. Like many of the best recommendations the utility of these statements is most apparent when they are not adhered to.

The identification of an appropriate model for an application becomes an important knowledge elicitation exercise when adopting a CommonKADS approach. The models themselves can be organised into tree like structures, see Figure 3. A series of questions about the domain attempt to establish which node in the tree best corresponds to the features of the application (O'Hara et al., 1998).

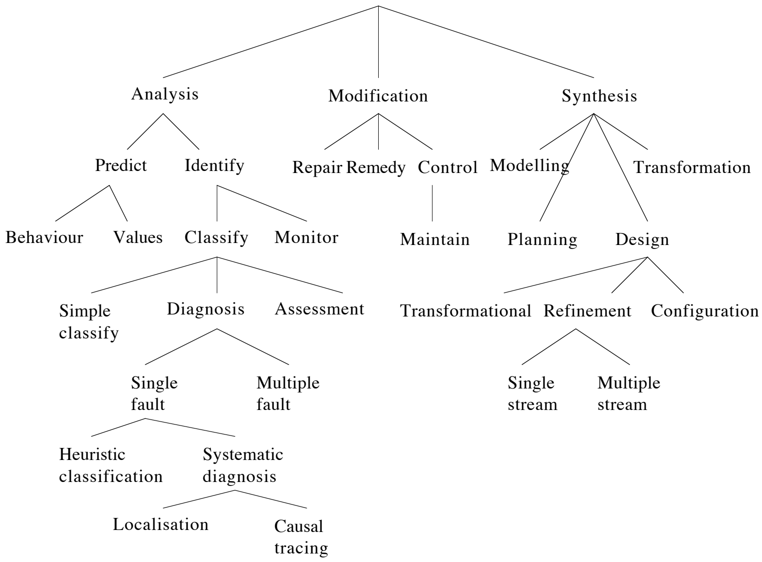

Figure 3 A Taxonomic Hierarchy of Problem Solving Models

Currently the most complete model set is that of CommonKADS (Schreiber, 2000). However, these models introduce a new level of complexity in the acquisition process and whilst they may by extremely helpful when trying to understand the implementation of expertise models whether the adoption of these various methodologies makes for more efficient and effective KE is a moot point as these claims have not been formally evaluated (Shadbolt et al 1999).

Automatic Knowledge Acquisition

As KE is acknowledged to be a time consuming and difficult process, the idea of automated elicitation is most attractive. A number of programs have been developed towards this goal, and we will briefly consider some of them in this penultimate section.

Software tools for KE can be split three categories: (1) *domain dependent* tools which have been developed for specific task domains; (2) *domain independent* tools which are computer implementations of one particular KE technique; and (3) *integrated systems* for acquisition and elicitation support.

Domain specific tools are tailored to elicit the types of knowledge known to be important in a particular application. An early example of this sort of system was SALT (Marcus, 1989). This had built in knowledge of the problem solving strategies used when configuring electro-mechanical systems. The expert interacts through an automated interview which allows the selection by menu of elements in the domain. The interview results are automatically converted into rules, and the expert has an opportunity to edit the resultant rules. In this task-oriented approach, the complexity of the knowledge acquisition process is reduced through the use of a model of the required knowledge as a template for customising otherwise general KA techniques and tools. Similar approaches can be found in medical domains such as therapy and treatment planning (Tu et al. 1995). These systems are by definition restricted to particular tasks and domains. Moreover, most of these domain oriented tools have remained research prototypes.

The second approach to automation of the KA process is to focus on one particular technique and support its use. Of those systems which implement individual standard techniques, many of the most successful are based on the *repertory grid*. This technique has its roots in the psychology of personality (Kelly, 1955; Jankowitz 2003) and is designed to reveal a conceptual map of a domain, in a similar fashion to the card sort as discussed above. The work of Shaw and Gaines was particularly influential in promoting its use (Shaw and Gaines, 1987). The technique as developed in the 50's was very time-consuming to administer and analyse by hand. This naturally suggested that an implemented version would be useful.

One of the earliest and best known programs was ETS (Boose, 1985) although this was developed primarily as a research tool. KSSO (Gaines, 1989; 1990) which we illustrate below formed the basis for a number of commercial products. More recently a web enabled freely accessible version of the software has become available (Gaines and Shaw 1997, http://tiger.cpsc.ucalgary.ca:1500/WebGrid/WebGrid.html) that provides an excellent means of experimenting with the approach and indeed undertaking machine supported elicitation sessions.

Briefly, subjects are presented with a range of domain elements and asked to choose three, such that two are similar, and different from the third. Suppose we were trying to uncover an astronomer's understanding of the planets. We might present him with a set of planets, and he might choose mercury and venus as the two similar elements, and jupiter as different from the other two. The subject is then asked for their reason for differentiating these elements, and this dimension is known as a construct. In our example 'size' would be a suitable construct. The remaining domain elements are then rated on this construct.

This process continues with different triads of elements until the expert can think of no further discriminating constructs. The result is a matrix of similarity ratings, relating elements and constructs. This is analysed using a statistical technique called *cluster analysis*. In KE, as in clinical psychology, the technique can reveal clusters of concepts and elements which the expert may not have articulated in an interview.

The automated versions are run in such a way that the repertory grid is built-up interactively, and the expert is shown the resultant knowledge. Experts have the opportunity to refine this knowledge during the elicitation process. In Figure 4 we can see that the expert has so far generated seven constructs along which the planets vary. In this case a seven point rating scale has been used and in the case of the construct size the smallest planet, mercury, has been given a rating 1 and the largest, jupiter, a rating of 7. The other planets have been rated in a comparative manner along the size construct[6]. The analysis has already revealed clusters of both constructs and elements. Thus jupiter and saturn are clustered together at around 82% similarity,

[6]In Figure 4 shading in the matrix is also used highlight ratings. Heavy shading designates a high value for an element on a construct.

neptune and uranus at around 88%, and these two pairs are clustered at around 80%. An astronomer might well observe that this group of four planets constitute the gas giants. A new concept has been uncovered. Similarly, constructs can be clustered. We see that the constructs relating to temperature and distance from the sun are clustered. Such associations can reveal causal or other law-like relations in the domain.

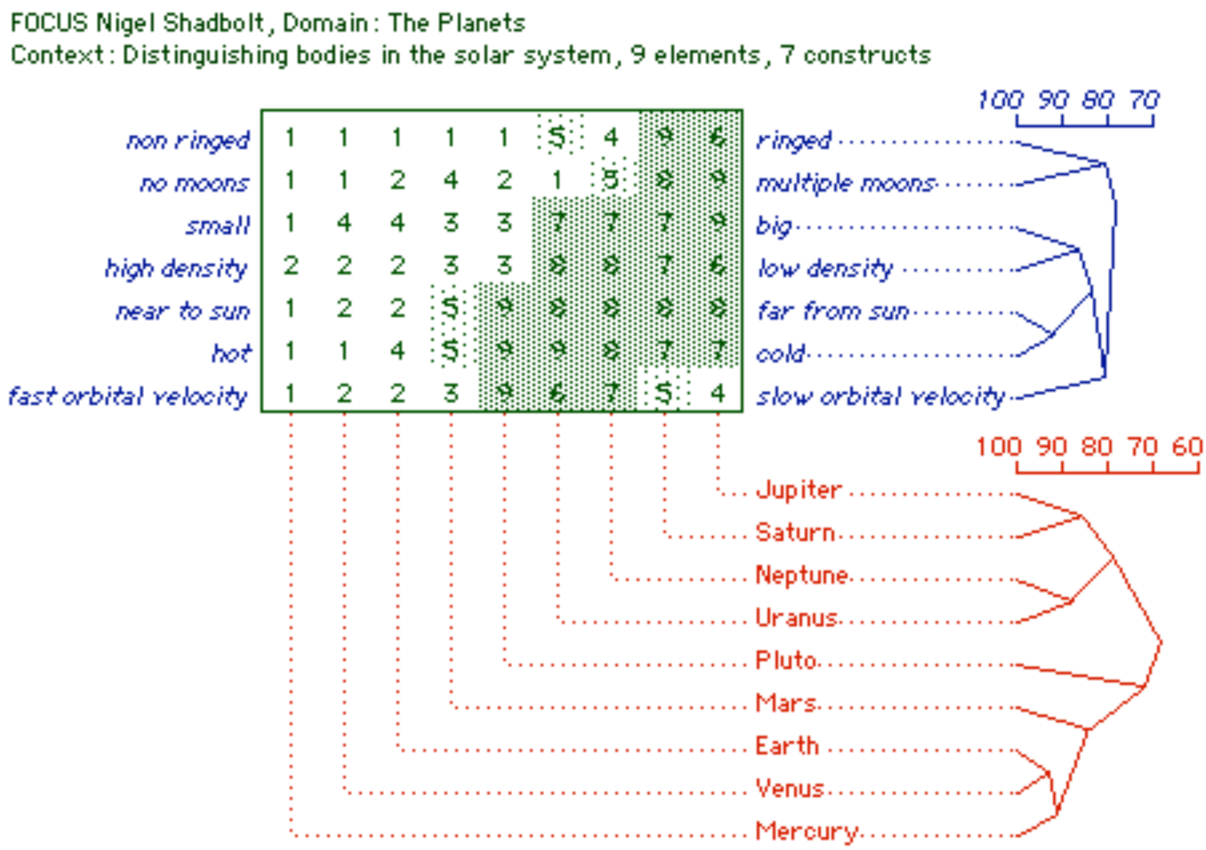

Figure 4 Knowledge elicited using WebGrid-III

We are also able to draw inferences from these structures in terms of implications between the constructs and elements. Examples of the sorts of implications we can draw are shown below

```
high density -> hot (3)
low density -> cold (5)

Overall Evaluation
  Correct 8/8  100.00%

fast orbital velocity -> high density (4)
slow orbital velocity -> low density (5)

Overall Evaluation
  Correct 9/9  100.00%

high density -> near to sun (4)
low density -> far from sun (5)
```

```
Overall Evaluation
  Correct 9/9  100.00%

big -> ringed (2 E1)
small -> non ringed (5)

Overall Evaluation
  Correct 7/8  87.50%   Errors  +:1  -:0  Total Errors 1/8  12.50%
```

Variants on this technique allow you to run sociograms so that one can compare one individual's view of a domain with another's – highlighting areas of consensus and difference. These systems can be found a place in any programme of elicitation.

Another widely used machine supported method is concept mapping. The maps were developed by Joseph D. Novak in educational contexts to help students express and share their knowledge. In this technique the expert and knowledge elicitor construct a graphical network of nodes and relations representing knowledge about a domain. A concept map is a two-dimensional representation of a set of concepts and their relationships, shown as concept names connected by directed arcs encoding propositions in the form of simplified sentences.

A number of computer supported versions of this tool have been developed but one of the most accessible can be obtained from the Institute for Human Machine Cognition (Cañas et al, 1999) at the following URL http://cmap.ihmc.us . In a browser, concept-map links can lead to diagrams, digital video, text, and arbitrary remote resources. Using these tools, domain experts can easily construct, navigate, share, criticize, and collaboratively refine knowledge models.

A related technology allows one to elicit and construct graphical networks that represent the rationale behind actions and decisions (Selvin and Buckingham Shum, 2000). Research has also been carried out trying to understand how to build KA tools for domains in which graphical representations are the most important (Cheng et al., 2001).

Although programs continue to be written to support single KA techniques there is an increasing trend towards the third type of approach mentioned at the beginning of this section. This approach is to integrate several KA tools. The idea is that the whole is greater than the parts (Shadbolt et al, 1993; Motta et al., 1993). One such system, PCPACK (Schreiber et al, 2000 Chapter 8includes protocol editing (enabling text and documents to be annotated and analysed), concept and process laddering, card sorts and various other rapid knowledge formation methods. The results of elicitation are stored in a persistent object-oriented database. All the tools are able to access this database providing a means of transferring knowledge between the various tools. The entire package is interfaced through an intuitive direct manipulation interface. A flexible web publication hypertext system is able to annotate objects in the database together with concise documentation and tutorial material. A demonstration version of PC PACK can be downloaded from www.epistemics.co.uk and evaluated. More recent versions have begun to incorporate templates and problem solving models that guide the user through the elicitation of process so as to populate a knowledge repository.

I have not included individual or collections of tools that originate from the disciplines of data mining (Witten and Frank, 1999) and machine learning (Mitchell, 1997). Such a discussion is beyond the scope of this chapter and as such they are not techniques that are used in sustained face to face knowledge acquisition sessions with experts. However, good references to the state of the are contained in the references given above.

Expertise and the Web

The most significant difference between the world of knowledge we now inhabit as against that of a decade ago is the extraordinary rise of the importance of the internet. Any figures provided become rapidly out of date but as of 2002 estimates are that the indexed web – the web the search engines can get at – comprises some 10 billion indexed pages and this is dwarfed by the so-called deep web. This deep web consists of huge numbers of databases, innumerable excel spreadsheets, a deluge of other content that is potentially available as an information resource but is as yet either not included or fully indexed.

What is undeniable is that this provides access to a huge potential resource for the construction of any prospective knowledge intensive system. Recently systems have been built with significant recourse to knowledge on the web. The MIAKT project (Hu et al,. 2003) has used information from the web to help build an ontology for the domain of breast disease. It has also located on the web an extensive library of images that are indexed with a range of information about patients and symptoms. Similarly if one takes the domain used throughout this chapter – classification of rocks and minerals – there are substantial online resources. These range from dictionaries and definitions of terms, succinct summaries of the process of rock formation, compact representations of diagnostic heuristics, and extensive online databases.

The ability to search out content of this sort offers a new and powerful way to build initial knowledge structures. However, one should be aware of the very considerable problems that attach to this sort of content. These include issues of provenance, context and interpretation.

When we locate an apparently relevant piece of content on the web how are we to judge its provenance? Who asserted the content, how long ago, what sources did the author use, what qualifications are associated with the content? At the moment very little if any of this information is associated as meta-data with the content we are interested. In the absence of such information one tends to resort to the tried and trusted notion of looking at the brand associated with the content. One is likely to take the content hosted on an IEEE standards web site or a Governmental Medical agency on trust rather more than that offered in more informal networks. However, it is now apparent that individuals and organisations will go to real lengths to appear to be the trusted sites of reputable organisations when in fact they are seeking to misinform. This whole issue of trusting digital content is attracting considerable interest and attention.

A more general problem is that of context – we may be able to download significant amounts of content but how are we to recognise the context in which it is appropriate to apply that knowledge. Human experts are extremely sensitive to the conditions under which knowledge should be applied. They are also often very aware of the context in which a piece of knowledge should not be applied or else relied upon. A third problem is one that to solve would require a general solution to providing machines with a full understanding of the meaning of language. Take quite straightforward scientific knowledge such as the following taken from two web sites dealing with Kepler's Laws of planetary motion. Consider the differences and ask how it is that we see them as essentially the same. We as humans seldom notice because we are so adept at understand the equivalences between statement, the nuances of different phrases, and the conditions under which more or less precise statement are required.

1	The orbit of a planet/comet about the Sun is an ellipse with the Sun's centre of mass at one focus	Planets move in orbits that are ellipses
2	A line joining a planet/comet and the Sun sweeps out equal areas in equal intervals of time	The planets move such that the line between the Sun and the Planet sweeps out the same area in the same time no matter where in the orbit
3	The squares of the periods of the planets are proportional to the cubes of their semimajor axes	The square of the period of the orbit of a planet is proportional to the mean distance from the Sun cubed

There is no doubt that the web as an extended knowledge repository is set to figure even larger in years to come. This is precisely the aim of research underway in bringing about the so-called Semantic Web (Berners-Lee, 2001). The aim is to build content that is much more richly described and enriched with information about the content itself – its provenance, its context of acquisition and so on. This attempt to bring about a range of knowledge services that can populate the web with such content and then exploit it in a number of ways is the focus of current research for one of the authors (www.aktors.org).

With a new generation of knowledge technologies, with techniques from data mining and machine learning we can expect to see larger amounts of knowledge intensive processing driven by knowledge that has been acquired in an automatic or semi-automatic fashion (Crow and Shadbolt, 2001). These topics go well beyond the scope of this chapter but they will set the agenda for research in knowledge intensive systems for the next decade and beyond.

Conclusions

Notwithstanding the developments alluded to in the last section it is folly to imagine that face to face knowledge elicitation, one human with another, will cease to be important and necessary. In fact it is essential for all of the reasons outlined above – to know the provenance of the content, to understand the conditions under which it is to be applied, to know how to interpret a problem solving context. Human experts are able to demonstrate a mastery of when, where and how their knowledge applies and more particularly when, where and why it might not. Human experts have accumulated their expertise over thousands of hours. This experience enables them to recognise situations and contexts that fall inside and outside of their competence. It enables them to make subtle judgements about the quality of the information they are presented with and the decisions they make. The fact that expertise and knowledge is ultimately grounded in human practice means that we need to understand the methods and techniques, problems and opportunities afforded by knowledge elicitation.

The problem of knowledge elicitation remains a subtle and complex one. This chapter has described some of the methods and techniques that are used in this enterprise. We have also sought to provide an indication of the difficulties inherent in doing this kind of work. Knowledge elicitation is itself a form of complex expertise. Experienced knowledge engineers come to recognise the characteristics of expert thinking. They develop skills that allow them to capture an expert's knowledge despite the many obstacles they face. Recently, methodologies have begun to emerge that seek to structure and manage the acquisition process.

As knowledge intensive systems become more widely deployed more people will face the challenges of knowledge elicitation. Whether it is in the form of a corporate intranet or a clinical decision support system knowledge is still the power driving these applications. Knowledge elicitation remains an important area of research and practical application.

References

Belkin, N. J., Brooks, H. M. and Daniels, P.J. (1987) Knowledge Elicitation Using Discourse Analysis. International Journal of Man-Machine Studies, 27, pp. 127-144.

Berners-Lee, T., Hendler, J and Lassila, O. (2001) The Semantic Web, Scientific American, May 2001.

Booch, G., Rumbaugh, J., and Jacobson, I. (1998). The Unified Modelling Language User Guide. Reading, MA, Addison-Wesley.

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

Burton, A. M., Shadbolt, N. R., Hedgecock, A. P. and Rugg, G. (1987) A Formal Evaluation of Knowledge Elicitation Techniques for Expert Systems: Domain 1. In Moralee, D. S. (Ed.) Research and Development in Expert Systems IV. Cambridge: Cambridge University Press.

Burton, A. M., Shadbolt, N. R., Rugg, G. and Hedgecock, A. P. (1988) Knowledge Elicitation Techniques in Classification Domains. ECAI-88: Proceedings of the 8th European Conference on Artificial Intelligence. pp. 85-93.

Cañas, A. J., Leake, D. B. and Wilson, D. C. (1999) Managing, Mapping and Manipulating Conceptual Knowledge, AAAI Workshop Technical Report WS-99-10: Exploring the Synergies of Knowledge Management & Case-Based Reasoning, AAAI Press, Menlo Calif, (July 1999).

Cheng, P., Cupit, J. and Shadbolt, N. R. (2001) Supporting diagrammatic knowledge acquisition: an ontological analysis of Cartesian graphs. *International Journal of Human Computer Studies,* 54 p.457-94

Chi, M,. Feltovitch, P. and Glasser, R. (1981) Categorization and representation of physics problems by experts and novices. Cognitive Science, 5(2):121-152, April 1981. 55

Chi, M. T. H., Glaser, R. and Rees, E. (1982) Expertise in Problem Solving. In Sternberg, R. J. (Ed.) Advances in the Psychology of Human Intelligence: 1. Hillsdale NJ: Lawrence Erlbaum.

Chi, M. T. H., Glasser, R. and Farr, M. J. (1988) The Nature of Expertise. Lawrence Erlbaum Associates.

Clancey., W. J. (1985) 'Heuristic Classification' in *Artificial Intelligence* vol.xxvii pp.289-350

Clayton, J., Gibson, E., Scott. C. (1991). *A Practical Guide to Knowledge Acquisition*, **Addison Wesley**

Clarke, B. (1987) Knowledge Acquisition for Real Time Knowledge Based Systems. Proceedings of the first European Workshop on Knowledge Acquisition for Knowledge Based Systems. Reading University, UK.

Cleaves, D. A. (1987) Cognitive Biases and Corrective Techniques: Proposals for Improving Elicitation Procedures for Knowledge Based Systems. International Journal of Man-Machine Studies, 27, pp. 155-166.

Crow, L. and Shadbolt, N. R. (2001) Extracting Focused Knowledge from the Semantic Web, *International Journal of Human Computer Studies,* Vol. 54 (1) pp 155-184.

Elstein, A. S. Shulman, L. S. and Sprafka, S. A. (1978) Medical Problem Solving: An Analysis of Clinical Reasoning. Cambridge, Mass.: Harvard University Press. Cited in Kuipers and Kassirer, Ericsson, K. A. (1996). The road to excellence. Mahwah, NJ: Lawrence Erlbaum.

Ericsson, K. A. and Simon, H. A. (1984) Protocol Analysis: Verbal Reports as Data. Cambridge, Mass.: MIT Press.

Ericsson, K. A. and Simon, H. A. (1993). *Protocol Analysis: Verbal Reports as Data, revised editio*n. Cambridge, MA, MIT Press.

Fellbaum, C (1998) WordNet: An Electronic Lexical Database, Bradford Books.

Feltovich, P., Ford, K., and Hoffman, R. (1997). Expertise in Context – Human and Machine. Cambridge, MA, MIT Press.

Firlej, M. and Hellens, D. (1991) Knowledge Elicitation: A Practical Handbook. Prentice Hall International.

Gaines, B. R. (1988) Knowledge acquisition systems for rapid prototyping of expert systems, INFOR, 26(4), pp256-285.

Gaines, B. R. (1997). Using explicit ontologies in KBS development. International Journal of Human-Computer Studies, 45(2).

Gaines, B. R. and Linster, M. (1990) Development of second generation knowledge acquisition systems. In B. Wielinga, J. Boose, B. Gaines, G, Schreiber and M van Someren (eds) Current Trends in Knowledge Acquisition. IOS Press, Amsterdam.

Gaines, B. R. and Shaw, M. (1997) Knowledge acquisition, modeling and inference through the World Wide Web. *International Journal of Human-Computer Studies* 46(6) 729-759.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA, Addison-Wesley.

Gammack, J. G. (1987) Different Techniques, and Different Aspects of Declarative Knowledge. In Kidd, A. L. (Ed.) Knowledge Acquisition for Expert Systems: A Practical Handbook. New York: Plenum Press.

Gammack, J. G. and Young R. M. (1985) Psychological Techniques for Eliciting Expert Knowledge. In Bramer, M. (Ed.) Research and Development in Expert Systems. Cambridge: Cambridge University Press.

Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. *Nature Genet.* 2000; 25: 25-29

Grover, M. D. (1983). A Pragmatic Knowledge Acquisition Methodology. IJCAI-83: Proceedings 8th International Joint Conference on Artificial Intelligence.

Hart, A. (1986) Knowledge Acquisition for Expert Systems. London: Kogan Page.

Hayes-Roth, F. Waterman, D. A. and Lenat, D. B. (1983) Building Expert Systems. Reading, Mass.: Addison-Wesley.

Hinkle, D. N. (1965) The Change of Personal Constructs from the Viewpoint of a Theory of Implications. Unpublished Ph.D. thesis, University of Ohio.

Hoffman, R. R. (1987) The Problem of Extracting the Knowledge of Experts from the Perspective of Experimental Psychology. AI Magazine, 8, pp. 53-66.

Hoffman, R. R., Crandall, B. and Shadbolt, N.R. (1998) Use of the Critical Decision Method to Elicit Expert Knowledge: A Case Study in the Methodology of Cognitive Task Analysis, *Human Factors*, Vol 40, p254-276.

Hoffman, R., Shadbolt, N.R., Burton, A.M. and Klein,G. (1995) Eliciting Knowledge from Experts: A Methodological Analysis. *Organizational Behavior and Decision Processes* vol.62(2) (1995) pp.129-158. Academic Press 0749-5978/95.

Hu, B., Dasmahapatra, S. and Shadbolt, N. (2003) From Lexicon To Mammographic Ontology: Experiences and Lessons. In Calvanese, D., De Giacomo, G. and Franconi, E., Eds. Proceedings International Workshop of Description Logics-DL'03, pages 229-233, Rome.

Humphreys BL, Lindberg DA, Schoolman HM, Barnett. The Unified Medical Language System: an informatics research collaboration. *J Am Med Inform Assoc*. 1998 Jan-Feb; 5(1): 1-11.

Johnson, L. and Johnson, N. (1987) Knowledge Elicitation Involving Teachback Interviewing. In Kidd, A. (Ed.) Knowledge Elicitation for Expert Systems: A Practical Handbook. New York: Plenum Press.

Johnson, P. E., Zualkernan, I. and Garber, S. (1987) Specification of Expertise. International Journal of Man-Machine Studies, 26, pp. 161-181.

Kahneman, D., Slovic, P. and Tversky, A. (Eds.) (1982) Judgement under Uncertainty: Heuristics and Biases. New York: Cambridge University Press.

Kelly, G. A. (1955) The Psychology of Personal Constructs. New York: Norton.

Kidd, A. L. (Ed.) (1987) Knowledge Acquisition for Expert Systems: A Practical Handbook. New York: Plenum Press.

Kuipers B. and Kassirer, J.P. (1983) How to Discover a Knowledge Representation for Causal Reasoning by Studying an Expert Physician. IJCAI-83: Proceedings of the 8th International Conference on Artificial Intelligence.

LaFrance, M. (1987) The Knowledge Acquisition Grid: A Method for Training Knowledge Engineers' in *International Journal of Man-Machine Studies* vol.xxvii

Marcus, S. and McDermott, J. (1989) SALT: A Knowledge Acquisition Language for Propose-and-Revise Systems, Artificial Intelligence, 39, pp1-38

McGraw, K. L. and Harbison-Briggs, K. (1989) Knowledge Acquisition: Principles and Guidelines. Prentice-Hall International..

Meyer, M. and Booker, J. (1991) Eliciting and Analysing Expert Judgement: A Practical Guide. Academic Press.

Milton, N., Shadbolt, N., Cottam, H. and Hammersley, M. (1999). Towards a Knowledge Technology for Knowledge Management. *International Journal of Human-Computer Studies*, 51(3), 615-64.

Mitchell, T. (1997) Machine Learning, McGraw-Hill.

Motta, E.(1999). *Reusable Components for Knowledge Modelling: Principles and Case Studies in Parametric Design*. IOS Press, Amsterdam, 1999.

Motta, E., O'Hara, K., and Shadbolt, N. (1993) Grounding GDMs: A Structured Case Study, Journal of Knowledge Acquisition, 5(4).

Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T. and Swartout, W. (1991)*Enabling Technology for Knowledge Sharing*. AI Magazine, vol 12(3), 37-56.

Newell, A. (1982) The Knowledge Level, Artificial Intelligence, 18, pp87-127.

Newell, A. and Simon, H. A. (1963) GPS, A Program That Simulates Human Thought. In E. Feigenbaum and J. Feldman (Eds.) Computers and Thought McGraw-Hill, New York.

Nonaka, I. and Takeuchi, H. (1995). The Knowledge-Creating Company. Oxford, UK, Oxford University Press.

Norvig, P. and Russell, S. (2003) Artificial Intelligence: A Modern Approach. Prentice Hall.

Noy, N. F., Sintek, M., Decker, S. Crubezy, M., Fergerson, R.W. & Musen, M.A. Creating Semantic Web Contents with Protege-2000. *IEEE Intelligent Systems* 16(2):60-71, 2001.

O'Hara, K. and Shadbolt, N.R (1997) Interpreting Generic Structures: Expert Systems, Expertise and Context. In Paul Feltovich, Ken Ford and Robert Hoffman (eds.) Expertise in Context. AAAI and MIT Press. ISBN 0-262-56110-7.

O'Hara, K., Shadbolt, N.R. and Van Heijst (1998) Generalised directive models: integrating model development and knowledge acquisition, *International Journal of Human-Computer Studies*, 49(4), p497-522.

Petersen, T (1994) Art & Architecture Thesaurus, OUP

Rich, E. and Knight, K. (1991) Artificial Intelligence. 2nd Edition. McGraw-Hill.

Schreiber G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N.R, Van de Velde, W. and Wielinga, B. (2000) *Knowledge Engineering and Management.* MIT Press.

Schvaneveldt, R. W., Durso, F. T., Goldsmith, T. E., Breen, T. J., Cooke, N. M., Tucker, R. G. and De Maio, J. C. (1985) Measuring the Structure of Expertise. International Journal of Man-Machine Studies, 23, pp. 699-728.

Selvin, A.M. and Buckingham Shum, S.J. (2000). Rapid Knowledge Construction: A Case Study in Corporate Contingency Planning Using Collaborative Hypermedia Proc. KMaC 2000: Knowledge Management Beyond the Hype. Aston Business School, Aston University, Birmingham, UK, 16-19 July 2000

Shadbolt, N. R. (1989) Knowledge Representation in Man and Machine. In R. Forsyth (Ed.) Expert Systems: Principles and Case Studies 2nd Edition. Chapman and Hall, London.

Shadbolt, N., Motta, E. and Rouge, A. (1993) Constructing Knowledge-Based Systems, IEEE Software, November 1993, pp34-39.

Shadbolt, N.R and O'Hara, K. (1997) Model-Based Expert Systems and the Explanation of Expertise. In Paul Feltovich, Ken Ford and Robert Hoffman (eds.) Expertise in Context. AAAI and MIT Press. ISBN 0-262-56110-7.

Shadbolt, N.R. and Burton A.M (1989). Empirical Studies in Knowledge Elicitation. ACM-SIGART Special Issue on Knowledge Acquisition, 108.

Shadbolt, N.R. and Burton, M. (1995) Knowledge elicitation: a systematic approach, in Evaluation of human work: A practical ergonomics methodology edited by J. R. Wilson and E. N. Corlett, Taylor and Francis, London, England, 1995. pp.406-440. ISBN-07484-0084-2

Shadbolt, N.R., O'Hara, K., Crow, L. (1999) The Experimental Evaluation of Knowledge Acquisition Techniques and Methods: History, Problems and New Directions International Journal of Human-Computer Studies, 51(4), 729-755.

Shaw, M. L. G. and Gaines, B. R. (1987) An Interactive Knowledge Elicitation Technique using Personal Construct Technology. In Kidd, A. L. (Ed.) Knowledge Acquisition for Expert Systems: A Practical Handbook. New York: Plenum Press.

Shortliffe, E. H. (1979) Computer-Based Medical Consultations: Mycin, American-Elsevier, New York.

Sowa, J.F. (1999) *Knowledge Representation: Logical, Philosophical, and Computational* , Brooks Cole Publishing Co.

Stefik, M. (1993). Introduction to Knowledge Systems. Los Altos, CA. Morgan Kaufmann.

Stewart, T. (1997). Intellectual Capital – The New Wealth of Organizations. London, Nicholas Brealey.

Tu, S. W. Eriksson, H. Gennari, J. Shahar, Y. and Musen. M. A. (1995)Ontology-Based Configuration of Problem-Solving Methods and Generation of Knowledge-Acquisition Tools: Application of PROTEGE-II to Protocol-Based Decision Support. *Artificial Intelligence in Medicine,* 7(Issue):257-289, 1995.

Welbank, M. A. (1983) A Review of Knowledge Acquisition Techniques for Expert Systems. British Telecom Research, Martlesham Heath.

Wielinga, B. J. et al.(1992) "KADS: A Modelling Approach to Knowledge Engineering." Knowledge Acquisition Journal 4(1), 1992.

Witten and Frank (1999) Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Elsevier.

Young, R. M. and Gammack, J. (1987) Role of Psychological Techniques and Intemediate Representations in Knowledge Elicitation. *Proceedings of the first European Workshop on Knowledge Acquisition for Knowledge Based Systems.* Reading University, UK.

Zsambok, C. E. and Klein, G. (1997) *Naturalistic Decision Making*, Elrbaum.