4th AAAI-Sponsored Knowledge Acquisition for Knowledge-Based Systems Workshop
Banff, Canada, October, 1989.

# DESIGN REQUIREMENTS FOR KNOWLEDGE SUPPORT SYSTEMS

Brian R Gaines

Knowledge Science Institute
University of Calgary
Calgary, Alberta, Canada T2N 1N4

## ABSTRACT

The social role of information technology is analyzed in order to provide a framework for reasonable requirements for knowledge support systems. The many different scientific and engineering communities targeted on the development of specific knowledge support technologies are noted, and the problems of integrating developments from different communities are highlighted. The architectures of current integrated knowledge support systems are analyzed within the social framework and the strengths and weaknesses of current modules are evaluated. Major system developments in recent years are used to exemplify the analysis, and expected directions for future development are outlined. The framework, analyses and examples are used to define a structured requirements specification for future research and development related to knowledge support systems.

## INTRODUCTION

This paper is the fourth in a series presented at previous AAAI Workshops on Knowledge Acquisition for Knowledge Based Systems which develop models of individuals and their role in the knowledge processes of society, and of the wide range of technologies supporting these knowledge processes (Gaines, 1987a, c, 1989a). Human society acts as a distributed anticipatory system modeling and controlling the world to enhance its probability of continued existence. Knowledge transfer processes within society are manifestations of coordinating communications between the components of the distributed system. Computer-based systems, such as expert systems, knowledge acquisition systems and intelligent tutoring systems, play a supportive role in extending and enhancing these processes. This paper analyzes these knowledge support roles in detail, defining the functions of the technologies involved, describing the current state of the art, and developing a requirements specification for integrated knowledge support systems.

Figure 1 shows the way in which the specific negotiation process between an expert and a client is part of the general knowledge processes within the society of which they are part (Gaines, 1989a). The expert role is supported by, and accountable to, an appropriate professional community responsible for the management of the knowledge processes within the relevant profession. The client role is supported by, and accountable to, an appropriate client community responsible for the identification of problems and the provision of resources within the relevant applications domains. The concept of a 'role' is emphasized because individuals play both expert and client roles within many different communities. Neither 'roles' nor 'communities' are absolute distinctions—they each represent useful simplifications of the, otherwise unmanageable, complexity of social processes.

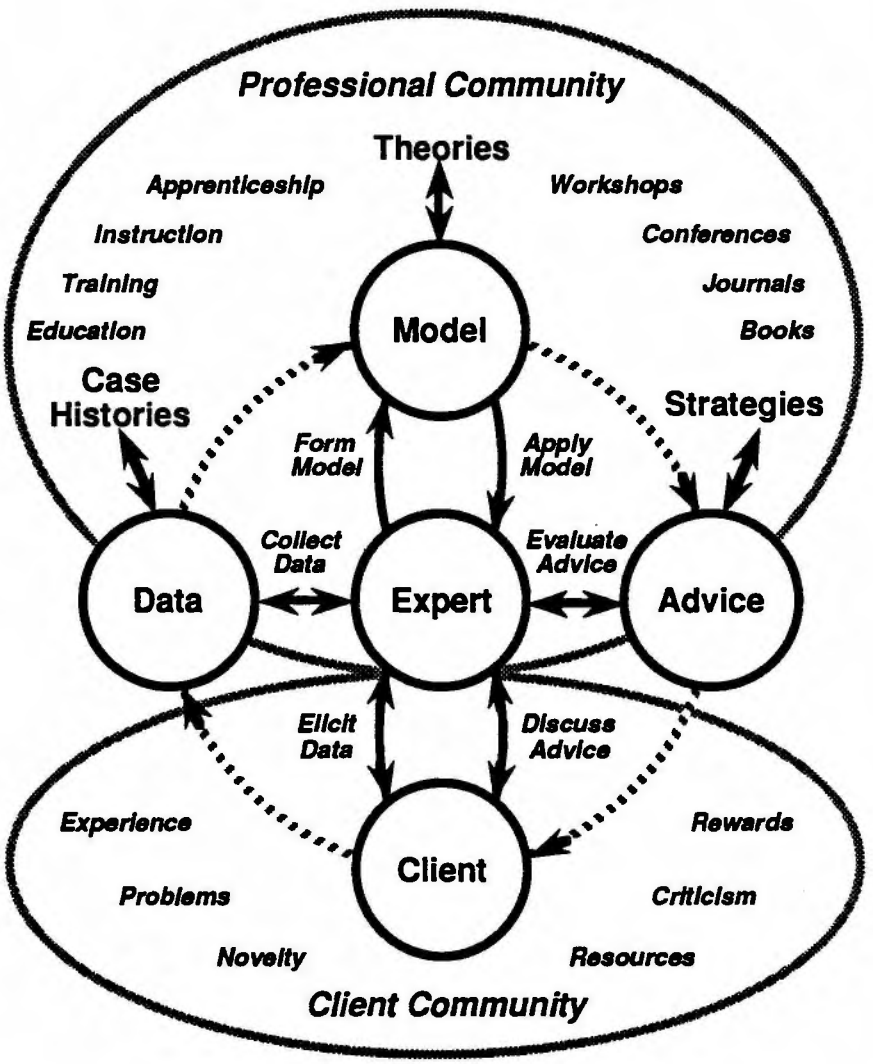


**Fig.1 The social context of knowledge support processes**

The negotiation cycle at the center of Figure 1 captures Hawkin's (1983) model of client-expert interaction. The expert elicits data about the problem from the client, forms a model based on this data, generates advice based on this model and feeds this back to the client. The client may query the advice and, possibly, the model, and the reasons for such a query lead to further data elicitation, and repeat of the elicitation/modeling/advice/query cycle. In this process the expert may collect new cases, form new models, and generate new advice-giving strategies, thus increasing his or her expertise.

The social mechanisms in the client community constitute positive feedback loops encouraging the concentration of individual expertise by giving established experts greater access to novel problems (Gaines, 1988c). The social mechanisms in the professional community constitute negative feedback loops encouraging the dissemination of individual expertise throughout that community.

This paper extends the model of Figure 1 to incorporate existing knowledge-based system technologies with a view to analyzing their role in social knowledge processes, and their strengths and weaknesses in fulfilling it. This analysis is used to develop a requirements specification for knowledge support system development during the next decade.

## KNOWLEDGE SUPPORT SYSTEMS

In analyzing the current state of the art in knowledge-based system technologies, it is interesting to look back at a previous generation of information technologies supporting human decision processes and see how our requirements have progressed. Keen and Morton (1978) define a *decision support system* as one using a computer to:

- Assist managers in their decision processes in semistructured tasks
- Support, rather than replace, managerial judgements
- Improve the effectiveness of decision making rather than its efficiency.

These look remarkably like the expectations for 'expert systems' since their inception. The only significant extension is that we expect an expert system to have some kind of 'explanation' capability, to:

- Account for the reasoning underlying advice given.

If we adopt a general definition of a 'manager' as a person responsible for activities in areas where we do not have effective, overt models, then the requirements above capture the general stereotype of an 'expert system.'

However, our expectations of knowledge-based systems in general have progressed well beyond this minimal list in recent years:

- The *knowledge acquisition* community has generated requirements for systems to incorporate knowledge elicitation and analysis modules automating expertise transfer (Boose & Gaines, 1988; Gaines & Boose 1988).
- The *machine learning* community has generated requirements for systems to incorporate inductive modeling modules automating empirical induction (Michalski, Carbonell & Mitchell, 1986).
- The *case-based reasoning* community has generated requirements for systems to incorporate comparison modules automating analogical reasoning (Kolodner, 1988).
- The *natural language* and *machine translation* communities have generated requirements for systems to incorporate linguistic analysis and generation modules automating language understanding (Cullingford, 1986; Lehrberger & Bourbeau, 1988).
- The *knowledge representation, explanation-based learning* and *symbolic reasoning* communities have generated requirements for systems to incorporate deep reasoning modules automating reductionist derivation (Cercone & McCalla, 1987; Hobbs & Moore, 1985).
- The *intelligent tutoring* community has generated requirements for systems to incorporate user modeling modules automating individualized knowledge transfer (Wenger, 1987).
- The *electronic mail* and *teleconferencing* communities have generated requirements for systems to incorporate communication modules automating inter-person communication (Vervest, 1987; Hiltz, 1984).
- The *electronic publishing* community has generated requirements for systems to incorporate typographic and graphic modules automating knowledge presentation (Vliet 1988).
- The *information retrieval* community has generated requirements for systems to incorporate knowledge indexing modules automating contextual retrieval (Oddy, Robertson, van Rijsbergen & Williams, 1981).
- The *visual programming* community has generated requirements for systems to incorporate interactive graphic modules automating visual interaction (Shu, 1988).
- The *hypermedia* community has generated requirements for systems to incorporate knowledge linkage modules automating associative thinking (Barrett, 1989).
- The *computer support of cooperative work* community has generated requirements for systems to incorporate knowledge-sharing modules automating group collaboration (Greif, 1988).

**Fig.2 Computer-based knowledge support processes**

Figure 2 shows some of the relations between a knowledge structure and other knowledge structures in the world, the mind, media and computers. The central knowledge structure might represent any of those shown peripherally so that the diagram captures a wide variety of knowledge transfers and transformations. The lower part of the diagram is largely concerned with knowledge application and the upper part with knowledge acquisition. Additional activities have been added such as the computer support of creativity which was the objective of Smith's (1977) *Pygmalion* system and of the Japanese *Sixth Generation* research proposal (STA, 1985; Gaines, 1986a), but does not seem to be a major current research focus. Moving around the lower part of the diagram from left to right there are a variety of user roles that correspond to various processes in the professional and client communities of Figure 1. To a large extent the ones to the right build on, and subsume, those to the left—for example, all roles shown are part of the essential serendipity (Roberts, 1989) of the creative process. Even if the system is not designed for the activities to the right it is unavoidable that it be used to support those roles since users will be involved in them regardless of the system designers' intentions. However, the effective support of any of the roles shown requires careful system design based on an understanding of the knowledge processes involved.

The figure and the list above are by no means exhaustive. There are communities concerned with numeric data processing, data bases, simulation, instrumentation and control, person-computer interaction and so on, which are also components of knowledge-based systems. The scientific *communities* involved in specifying these requirements are emphasized because this draws attention to the current fragmentation of research on knowledge-based systems. There have been a few major projects that cross these communities and combine requirements, notably the MYCIN/TEIRESIAS/ GUIDON developments (Buchanan & Shortliffe, 1984; Clancey & Shortliffe, 1984), but the majority of developments fall within particular communities and focus on narrowly defined objectives. This division of labor has been a major strength in developing specific knowledge technologies, but a counter-balancing integration of the requirements and techniques for achieving them is becoming increasingly important. Each area has far to go in its specific developments, but each of them is also facing limitations through problems where potential solutions are in other areas.

The term *module* is used deliberately because it is symptomatic of problems in developing integrated systems that there are now modules without 'modularity.' Current modules are generally designed as stand-alone systems where open-system considerations (Slonin, Schonbach, Bauer, MacRae & Thomas, 1987) of inter-operability, portability and scalability are not significant requirements. These considerations are fundamental meta-requirements if advances in knowledge-based system technologies in each of the specialist communities are to migrate across boundaries, supporting developments within each community, and also supporting the development of integrated, large-scale, knowledge-based systems. *Integratability* is itself a significant technology with its own methodologies and techniques. A new world of system integration possibilities would be opened if every system developed was split into functional modules which could be called as libraries by other modules.

The generic term *knowledge support system* is used to encompass all systems whose primary function is to support knowledge processes in society. The primary focus in this paper is on computer-based knowledge support systems, but the term is deliberately left unqualified to encompass non-computer-based systems supporting knowledge processes. Effective relationships with existing media and social phenomena supporting knowledge processes form another major meta-requirement for computer-based knowledge support systems. Information technology is a by-product of what Beninger (1986) terms the 'control revolution' taking place from the mid nineteenth century to today. He views this as a necessary social response to the complexities of global economic processes whose problems of instability were identified by Durkheim (1893). This revolution and its supporting technologies are themselves intermediate products of the longer-term evolution of human civilization and its supporting knowledge processes (Gaines & Shaw, 1986; Wojciechowski, 1986).

To view information-technology as a by-product of social processes is not to deny its role as an enabling technology generating change in those processes. The notions of cause and effect are only convenient perspectives from which to view the complexities of the total social system. Innis (1972) has analyzed the profound role of media in generating social change throughout recorded civilization, and Meyrowitz (1985) has given a deep analysis of the impact of electronic media on social behavior. Heim (1987) has given a thought-provoking study of the impact of the word processor on academic scholarship and the underlying conceptual structures. Knowledge support technologies are both a response to social needs and an agent of social change. As Popper (1968) has noted knowledge is produced by people for particular purposes, but once it exists it has unintended and unexpected consequences making it quasi-autonomous.

It is interesting to read the recent literatures of the various communities listed above and discern the commonality of their underlying objectives. For all the achievements of modern information technology, it has yet to deliver the level of support of human knowledge processes that have excited those involved in its development. This is a healthy source of frustration and continuing challenge, but the frustration will not be overcome and the challenge met by developments in one community alone. It is the combination of the current and coming developments from each community in integrated knowledge support systems that is necessary to satisfy the aspirations in each community. Sooner or later an effective 'divide and conquer' strategy has to be transformed into an 'integrate and evolve' strategy. Perhaps we and the technologies are not yet ready, but unless we continue to test that readiness the frustration will continue and the larger challenges will not be met.

The next section puts some critical technical developments within the social context of Figure 1 in order to analyze the current state of the art in knowledge support systems and identify reasonable requirements for the next stage of research.

## KNOWLEDGE SUPPORT SYSTEMS

Figure 3 shows the architecture of a knowledge support system standing between the professional and client communities of Figure 1 as an intermediary between experts and clients. The detailed architecture captures the current structure of knowledge acquisition tools, knowledge bases and expert system shells, extended to allow for the integration of hypermedia and intelligent tutoring concepts. The shaded modules indicate areas where our understanding, methodologies and tools are currently relatively weak. The role of the 'knowledge engineer' shown on the right links back to earlier arguments that this role is no longer one of an *intermediary* between expert and knowledge-base, but rather one of *managing* the knowledge acquisition and overall system development processes (Gaines, 1987b). The knowledge engineer and the technologies for knowledge acquisition, advice giving, explanation, and tutoring, are part of the *knowledge technologies community*.

The top oval encompassing the *knowledge acquisition system* in Figure 3 contains tools for eliciting knowledge from experts and for analyzing the knowledge to create a knowledge base. The intermediate knowledge bases elicited include the experts' conceptual structures and problem-solving strategies, and sample cases in the domain. The analysis tools are also used to feed back this knowledge to the experts in various forms for checks of coherence and consistency, and to prompt the experts to explore and structure their knowledge. Shaw and Woodward (1989) have analyzed and tabulated existing knowledge acquisition methodologies and techniques in terms of their elicitation and analysis components, and their relative emphasis on the elicitation of problem structure versus problem-solving strategy. Boose (1989) has noted the weaknesses of current knowledge acquisition tools in eliciting problem-solving strategies which limits their application in synthesis tasks such as design and planning.

The bottom oval encompassing the *expert system shell* in Figure 3 contains tools for making inferences from general knowledge about specific cases from clients and for interacting effectively with the clients. The intermediate interactions generated include queries to define the specific case, advice about appropriate action and explanations about the basis for such advice. The analysis tools are also used to feed back this knowledge to the experts in various forms for checks of coherence and consistency, and to prompt the experts to explore and structure their knowledge. Clancey (1987a) has noted the weaknesses of explanation facilities in current expert system shells which limits their application in knowledge transmission tasks such as tutoring.

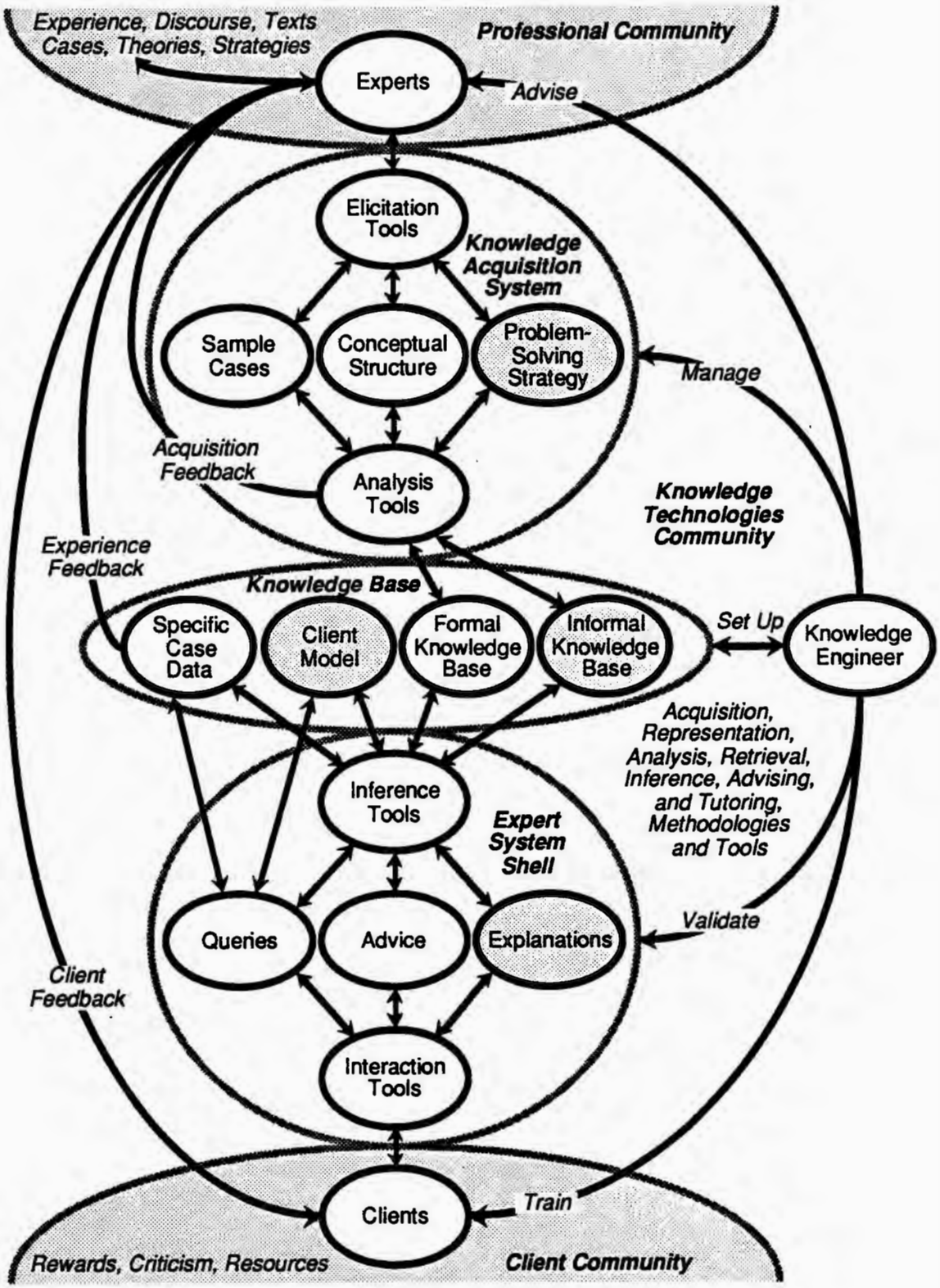


**Fig.3 Knowledge technologies within the social context of client-expert relations**

The central oval encompassing the *knowledge base* in Figure 3 contains four components: a *formal knowledge base* capturing knowledge structures that may be used computationally as part of a formal inference process; an *informal knowledge base* capturing knowledge that may be used for annotation and explanation but not for computation; *specific case data* that may be used to model the client's situation in a form suitable for advice generation; and a *client model* that may be used to target advice and explanations to the needs and capabilities of particular clients.

The formal knowledge and specific case data components are part of all expert system shells. The informal knowledge base is a common extension of current shells through links with hypermedia tools in order to support knowledge processes such as association and visualization that do not depend on formal inference but support human understanding of such inference. The client model is a research extension of current shells to provide user modeling supporting intelligent tutoring systems and intelligent user interfaces. It is reasonable to expect that most interactive systems supporting knowledge-based discourse with clients will need to have some form of client model not only to cope with diversity in the community of clients but also to cope with the varying knowledge states of individual clients.

It is noteworthy that the flows of information in the situation portrayed in Figure 3 are such that the specific case data and client model both necessarily depend on client interaction with the system, either through queries originating from the inference tools in the shell or from dialog initiated by the client. The system only knows about the case as described by the client. The accuracy of its inference procedures and resultant advice are critically dependent on the client understanding the conceptual structures of the expert sufficiently to answer queries correctly (Shaw & Gaines 1988). Similarly, the system only knows about the client as modeled through discourse with the system and hence the richness and accuracy of the client model are critically dependent on what may be a very limited communication.

The relations between the professional and client communities after the introduction of knowledge-based system technologies can also be analyzed in terms of the feedback paths on the left of Figure 3. At the upper left, *acquisition feedback* from the knowledge acquisition system to the expert during the development of the formal and informal components of the knowledge base is well-established practice. At the center left, *experience feedback* from the application is shown as an equally important but less well-supported feature. If the expert, after the acquisition phase, is completely detached from the clients' experiences in using the knowledge-based system it can seriously undermine his or her role as a manager of the inductive process, and the related development of the professional community. A significant requirement for any knowledge support system is that it capture case data, particularly anomalous cases, and feed this back to the expert (Gaines, 1988c) as shown in the experience feedback path.

At the lower left, even if the experience feedback path is totally effective, direct *client feedback* through expert-client interaction is still necessary if the case falls outside the conceptual structure that has been elicited from the expert. It may be necessary to transcend the discourse between client and expert supported by the system and initiate another level of communication that may result in major changes to the overall system. In any event, in a professional community the need to support and interact with clients outside the constraints of formal relationships imposed by any means, including technology, is well-recognized as crucial to professional development (Schön, 1983).

The role of the expert within the professional community as a recipient of new knowledge disseminated within that community, as shown at the top of Figure 3, emphasizes the need for knowledge acquisition to be an ongoing process, not a one-off development.

As emphasized previously, the distinctions between experts, knowledge engineers and clients in Figure 3 is one of roles, not individuals (Gaines, 1987c). In particular, the expert and client roles may be played by the same person at different times. It is clearly good practice for the expert to act as a client for the system incorporating his or her expertise. If the knowledge acquisition system is embedded in the advisory system, it is feasible for the knowledgeable client to assume the role of expert and post an anomalous case edited to have the correct advice to the database of sample cases and re-run the analysis tools to generate a new knowledge base (Gaines, Rappaport & Shaw, 1989). The distinctions made in Figure 3 do not preclude such activities but they are significant none the less because the *responsibilities* and *accountabilities* are different for the same individual in his or her different roles.

## THE STATE OF THE ART

There are now many systems that implement part of Figure 3 but few that attempt to encompass the complete structure shown. It is interesting to analyze the more complete systems in terms of their structure, achievements, strengths and weaknesses, with a view to determining the current state of the art.

The outstanding example of a relatively complete implementation of a knowledge support system with the architecture of Figure 3 is the MYCIN/TEIRESIAS/GUIDON system already referenced. This has also been extremely well-documented by its developers (Shortliffe, E.H., 1976; Davis, 1982; Buchanan & Shortliffe, 1984; Clancey & Shortliffe, 1984; Clancey, 1987a, b, c) and others (Wenger, 1987; Evans & Patel, 1989) so that the analysis can be extended as far as required.

In terms of the lower oval in Figure 3, the EMYCIN variant of MYCIN is the original expert system shell on which later developments have been modeled. Its explanation facilities are based on the presentation of the rules used in deriving its conclusions, and this has again been a model for later systems. The main developments since have been in more structured knowledge representations for expressing the experts' conceptual structures such as frame and object representations which allow more concise and expressive inference rules to be used. Research developments include different forms of inference, particularly case-based reasoning based on direct comparisons with the sample cases rather than than dependence solely on inference rules. None of these changes affects the potential system advisory performance since structured representations can be flattened and case comparisons can be expressed as rules, but they do significantly affect the discourse possible with the system by both expert and client.

In terms of the upper oval in Figure 3, the TEIRESIAS component of MYCIN provided the primary knowledge acquisition system. It has some remarkable features which do not appear to have been replicated in later systems, notably an ability to guide debugging of the knowledge base on anomalous cases through meta-rules, and primitive capabilities to translate restricted natural language to rules for acquisition and rules to natural language for acquisition feedback (Davis 1982). The meta-rules also provide a vehicle for the capture of problem-solving strategies, but this was applied primarily to debugging strategies rather than diagnostic strategies.

MYCIN and TEIRESIAS supported the advisory and developmental activities of the system viewed as a knowledge support system, and it was natural to suppose that this support role could be extended to tutoring. However, in developing GUIDON as an intelligent tutoring system based on MYCIN, Clancey (1987a, b, c) found it necessary to radically restructure both the knowledge base and inference tools resulting in NEOMYCIN, and to add a modeling capability to keep track of the students' knowledge states as they played out their own strategies in attempting the diagnostic task.

The knowledge base restructuring was to some extent a decompilation from decision rules back to the knowledge underlying them, and to some extent the addition of knowledge irrelevant to performance but significant in explanation. The student modeling capability involved a separation of problem-solving strategies, particularly hypothesis management from the domain knowledge. The direction of development of expert system shells towards object-oriented databases has made it easier to represent knowledge through declarative structures leading to decisions through the application of strategic meta-rules. The choice of cognitively natural structures and strategies is still a major problem. Archetypal research focusing on the ontology of knowledge is the KADS project in Europe (Breuker & Wielinga, 1987), and archetypal research focusing on problem-solving strategies is the 'generic task' approach (Chandrasekaran, 1988) capturing the methodologies underlying many of the problem-solving methods used in significant expert system developments (McDermott, 1988).

It is easy to assume that advances in technology supporting an improved system that are the end point of one project are the natural starting point for further projects. However, Clancey (1987a), in summing up the GUIDON development experience, sounds a note of caution whose rationale is very significant to future requirements:

> "Despite the now apparent shortcomings of MYCIN's rule formalism, we must remember that the program was influential because it worked well. The uniformity of representation, so much the cause of the 'missing knowledge' and 'disguised reasoning' described here, was an important asset. With knowledge so easy to encode, it was perhaps the simple parametrization of the problem that made MYCIN a success. The problem could be built and tested quickly at a time when little was known about the structure and methods of human reasoning. ... We can treat the knowledge base as a reservoir of expertise, something never before captured in quite this way, and use it to suggest better representations."

These remarks can be read not only as a commentary on the state of the art in knowledge-based systems but also, contrapuntally, as an insight into cognitive processes whereby knowledge is expressed and iteratively refined. The elegance and explicability of more structured knowledge representations do not imply that they are necessarily more appropriate for knowledge acquisition than other representations that may be closer to the 'compiled' knowledge structures underlying skills. In terms of requirements, it is not a question of right and wrong representations, but rather a reasonable plurality of representations with usable tools to move between them.

The significance of the MYCIN-NEOMYCIN transition for knowledge support system architectures is not so much that intelligent tutoring should be supported, but that effective explanation may involve knowledge structures over and above those required for effective performance. There is a range of behaviors we may expect from a knowledge support system that involve different operational criteria and performance evaluation. Information retrieval requires effective indexing but no computation with the stored material. Advice giving requires effective inferencing but no communication of the basis of the advice. Explanation, as an auditing activity, involves effective communication of a trace but no justification of the steps in the trace. Explanation, as a tutoring activity, involves justification of reasoning within a framework of a student's existing knowledge but no overt attempt to promote creative thinking.

These, and other, operational modes are reasonable requirements for knowledge support systems, but the concept of knowledge 'acquisition' begins to break down if we consider it as primarily as a transfer process leading to a single underlying knowledge base. A system adequate to operate in all these modes will involve multiple levels and representations of knowledge and will be in a state of continuous refinement.

Other system developments targeted on a knowledge support role have shown similar patterns of development to that analyzed above. The transition from ETS (Boose, 1986) to AQUINAS (Boose & Bradshaw, 1987) has been one of increasing the richness of knowledge structures, including extensions to manage and utilize multiple sources of expertise, to enhance the case-based reasoning tools, and to add an intelligent dialog manager automating some activities of the knowledge engineer (Kitto & Boose, 1987). The heterogeneous integration of KSS0 (Gaines & Shaw, 1987) as a knowledge acquisition tool, with NEXPERT (Rappaport 1987) as an expert system shell supporting the formal knowledge base, and HyperCard (Goodman, 1987) as a hypermedia system supporting the informal knowledge base, has created an integrated knowledge support system in which acquisition and performance are complementary continuing activities (Gaines, Rappaport and Shaw, 1989). Musen's (1989) systems for knowledge acquisition for clinical trials exemplify the power of a multi-level, meta-systems approach.

The major problem in all these extensions and enhancements has been to maintain an essential simplicity in application—to increase functionality without loss of usability. This requires a system architecture which is initially simple but unfolds to offer increasingly complex capabilities in response to the growing needs and skills of the user. It might be this is best achieved through a series of distinct tools, for psychological as well as technical reasons, but it is also a challenge to achieve this responsiveness within a single, coherent architecture.

## THE ROLE OF HYPERMEDIA

The one module in Figure 3 that the GUIDON developments did not address is the 'informal knowledge base' on the right. Presumably, the reasons for the this were that the focus of attention was on formal knowledge structures and also that hypermedia technology was not in a sufficiently advanced state. It is an interesting thought experiment to imagine a MYCIN development within a 'library and information science' ethos in which medical text books and journal articles are seen as the paramount vehicles for knowledge. Effective information presentation, including decision trees, and hypertext linkage to relevant background information, clinical trial data, counter-indications, and so on could be used to provide a system supporting medical scholarship for those at various levels of expertise. The IRIS Intermedia applications experience suggests that such an approach does effectively support the knowledge processes involved, including knowledge transfer (Yankelovich, Landow & Heywood, 1987).

What would be missing in such a system would be the capability of the system to act as a concise advisor, asking appropriate questions and making diagnoses. However, adding OPS5 or Prolog rules to the informal knowledge base in such a way that they 'shadow' corresponding textual statements, and are mutually accessible through hypertext links, creates a mechanism for conventional 'expert system' consultations. Figure 4 shows the shadowing of the informal by the formal as a system architecture. In fact, systems with such facilities already exist but they have been created through the reverse process, of annotating the formal structures of a knowledge-based systems with hypertext links.

The significance of this thought experiment is not in the technology, but in where it places the definitive knowledge structures. Books, journals, diagrams and pictures are still our prime repositories of knowledge. Formal knowledge structures supplement them by allowing them, in part, to become operational which is a significant extension. However, operationalism is a supplement to communication, not a replacement for it. A person who does something for you when you ask them to show *you* how to do it has a very limited role repertoire. Human society is robust in its functioning because it has developed a very wide range of forms of interaction and levels of discourse.

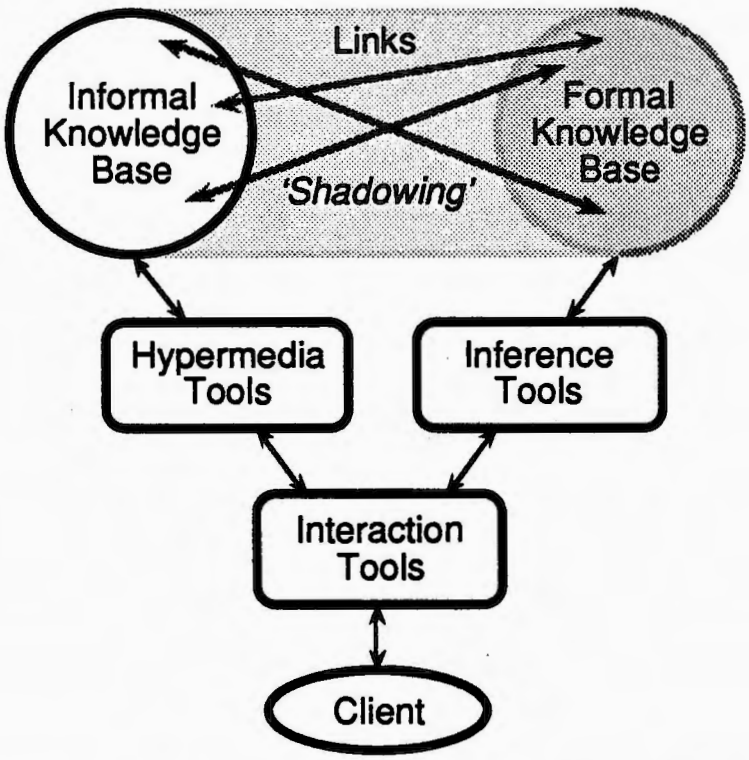


**Fig.4 Shadowing informal knowledge with formal knowledge**

Sergot (1988) has addressed the relationship between informal and formal knowledge structures when assessing the various legal expert systems that have been developed. He points out that the apparent clarity of law when expressed in logical rules is misleading. It is a key rule in the interpretation of legal documents that the exact wording is critical so that paraphrasing is forbidden. Yet the computational structures of the expert system are inherently a paraphrase of the legal text in a formal language. Sergot also notes some paraphrases involve more problems than others. Much legal expertise relates to the interpretation of written law, and it is good practice within the formal knowledge base to keep the more volatile interpretative rules separate from the more stable formalization of the written legislation.

These considerations are ubiquitous, not a peculiarity of the legal system. The facts of scientific experiments, such as clinical trials, and our interpretations of them are distinct knowledge structures. Both are theory-laden, in that our conceptual frameworks determine what we can express. The written descriptions, instrumentation records, photographs and diagrams with which we describe both are the paramount repositories of knowledge. Formal, operational translations of these may also attain a similar status but the formal and informal knowledge structures do not determine one another, they act only as mutual constraints. In particular, interpretations are volatile and subject to reconceptualization.

This is the essence of scholarship, explanation and tutoring, and also of the role of the expert as a manager of the reflective equilibria between cases and explanations, facts and theories, decisions and rationalizations. The formal and informal knowledge bases in Figures 3 and 4 are not alternatives or separated through technological distinctions. They are both essential and are linked through subtle relationships which are derived from the essential nature of knowledge. The types of link that will exist in a knowledge base are shown in Figure 5.

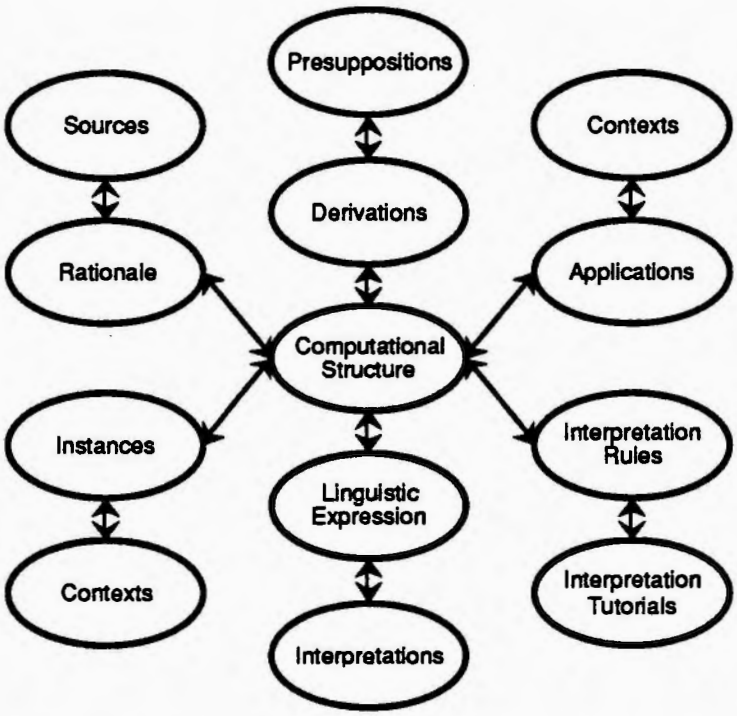


**Fig.5 Links between informal and formal knowledge**

In the center of Figure 5 is some computational structure which may be as small as a single rule or as large as a complete expert system—annotation is necessary at all levels and varies in nature at different levels. Immediately below is the linguistic expression of the structure, possibly quoted from some document which is itself the definitive statement of the knowledge. This in turn is linked below to interpretive commentaries. Above the computational are various alternative derivations of it which themselves depend on presuppositions to which they are linked. At the top left are various rationales for the structure linked back to their sources—these rationales may involve references to the other structures such as derivations. At the bottom left are instances of the knowledge structure linked into contexts in which they can occur. At the top right are applications of the knowledge structure again linked to their contexts. At the bottom right are the explicit interpretation rules for the terms within the computational structure linked to tutorial material to support the client in using these terms consistently with the system.

Figure 5 presents only an example of significant possibilities and omits the many cross-reference links that will occur. The use of labeled links to carry knowledge structures was an important part of Trigg's (1983) original Textnet design, and Van Lehn (1985) has described a case history in which the development of such links led to changes in knowledge structures in a discipline. Hypermedia developers have emphasized the support of knowledge processes through the user chasing the informal links, whereas expert system developers have emphasized the computer chasing the formal links. The two approaches are complementary and can be coupled within their existing technical frameworks and existing user interfaces (Gaines & Vickers 1988).

The short-term problems are technical ones of having the appropriate 'hooks' available in expert system and hypermedia packages to support communication between them so that they appear as a seamless single application to the user. Often this can be achieved when it is not part of the original design, but more powerful and elegant implementations can be achieved if an open systems approach is adopted from the start.

## REQUIREMENTS FOR KNOWLEDGE SUPPORT SYSTEMS

Having highlighted developments stemming from the MYCIN project as illuminating key aspects of knowledge support system requirements, it is appropriate to commence drawing up a more explicit specification by listing the requirements specified by MYCIN users, that the system should (Clancey & Shortliffe, 1984 p.464):

- Be able to explain their decisions to users
- Be portable and flexible so that users can access them at any time and place
- Display an understanding of its own knowledge
- Improve cost-efficiency
- Automatically learn new information when interacting with experts
- Display common sense

and by the system's developers, that research should focus on (Clancey & Shortliffe, 1984 p.467):

- Psychological studies providing new insights into simulating expert decision-making
- Techniques for representing and using causal and mechanistic relationships allowing reasoning from first principles
- Methods for acquiring expert knowledge, encoding it and checking it for consistency and completeness
- Explanation facilities guided by understanding of how people explain things to one another and adapt to the knowledge and experience of the person requesting advice
- New machine architectures supporting high-performance decision-making programs
- Melding of symbolic techniques drawn from artificial intelligence and analytic techniques of statistics, pattern recognition and decision theory
- Novel ways in which personal computing and graphics might improve the acceptability and cost-effectiveness of systems aiding decision-making tasks.

These requirements include the 'second-generation' (Gaines 1988a) knowledge-based systems concepts that systems should be able to learn through deep reasoning (Steels 1986) and empirical induction, an emphasis on knowledge adequate to support explanation and understanding, a need to understand processes of human expertise and knowledge acquisition, and considerations of new technologies enabling system improvement in a variety of ways. They omit considerations of the embedding of expert systems in society and the embedding of expert systems in larger-scale information systems.

Is it possible to develop a 'requirements specification' for knowledge support systems that captures the diversity of issues addressed in this paper and elsewhere, and yet does not become so all-embracing as to be meaningless? I believe that it is both reasonable and necessary to begin to attempt to do so. Reasonable, because there are now enough development and application examples to ground the specification in significant experience. Necessary, because large-scale system developments require clear objectives and careful management if they are result in effective sustainable applications.

One of the problems we have to face is that any requirements specification for knowledge support systems is bound to apply to all information technology, that is, we can not differentiate something that is specifically a 'knowledge-based' system. Knowledge support systems carry with them all the problems and requirements of conventional information technologies plus additional requirements that stem from greater expectations about their capabilities and roles. However, success at the conventional level, particularly in areas of open-architecture, user interfaces and interactivity, is essential to success at the other levels peculiar to knowledge support systems. Some general technical considerations are too important not to be mentioned, although they are non-specific.

## THE TOP-LEVEL GENERATIVE META-REQUIREMENTS

The overall consideration from which all requirements for knowledge support systems can be generated is that they be specified in terms of their roles in the social knowledge processes shown in Figure 1. This is in contrast to specifying them in technical terms or individual cognitive terms as is common currently.

**Overall Requirement**: A knowledge-based system is a component in a structure of social knowledge processes. Specify the top-level requirements in terms of the relevant structure, existing knowledge processes, intended changes and unexpected side-effects.

Some meta-requirements may be derived immediately from this in terms of the generic social mechanisms shown in Figure 1.

**Meta-Requirement 1**: The social structure usually involves a professional community responsible for managing the processes of knowledge acquisition and dissemination and a client community dependent on the knowledge for its activities. Each community has existing requirements for knowledge processes that need to be satisfied by any knowledge support system. Specify these as part of the system requirements.

**Meta-Requirement 1.1**: The professional community has procedures for managing knowledge acquisition and dissemination. Specify that these should be supported, not by-passed, by the knowledge support system. For example, the 'experience feedback' and client 'feedback' paths in Figure 3 should be explicitly recognized and supported.

**Meta-Requirement 1.2**: The professional community has procedures for assessing the risk of using new knowledge processes, controlling resulting hazards, and ensuring proper accountability. Specify that these should be supported, not by-passed, by the knowledge support system.

Other meta-requirements may be derived by considering the role of knowledge in terms of the generic social mechanisms shown in Figure 1. Knowledge is valuable, preserved in many forms, fallible, constantly changing, and created through a wide variety of processes, some intrinsically irrational.

**Meta-Requirement 2**: A knowledge support system will not contain all the knowledge relating to the processes in which it is involved, or provide all the facilities required. Specify how the system will relate to existing knowledge support systems including non-computer media and information technology.

**Meta-Requirement 2.1**: Specify the system to have an open architecture such that other systems are readily embedded in it, and it is readily embeddable in other systems.

**Meta-Requirement 2.2**: Specify field enhancement, upgradability, scalability and portability of functionality as essential requirements. Specifying the automation of these leads to various forms of embedded learning.

Another class of meta-requirements that are peculiar to knowledge-based systems is to note the intrinsic reflexivity of knowledge processes, that they apply to themselves, and require that the system support this.

**Meta-Requirement 3**: Knowledge processes are intrinsically reflexive, applying to themselves. Require that this reflexivity be explicitly supported by models within the system of itself, its users, its context, its processes and its tasks.

**Meta-Requirement 3.1**: Specify that the system should be able to explain its own operation.

**Meta-Requirement 3.2**: Specify that the system should be able to adapt to the differing knowledge states of its users.

Further refinement and instantiation of these meta-requirements is in itself adequate to generate specifications for the various sub-processes shown in Figure 2.

Figure 6 shows a system architecture for a large-scale knowledge-support system in an industrial context that has the capability to satisfy these requirements through integration of existing heterogeneous systems (Gaines 1988b). The lowest level is a network layer supporting open system integration. The middle level is a combination of user interface and information systems with virtual circuits between them, that is, current information technology. The upper level comprises the knowledge layers relating to the organization and the world within which it operates. The organizational layer captures the knowledge processes within the organization, and the external world layer captures those outside it. A companion paper at this meeting analyzes some of the knowledge acquisition requirements for such a system (Sharp 1989).

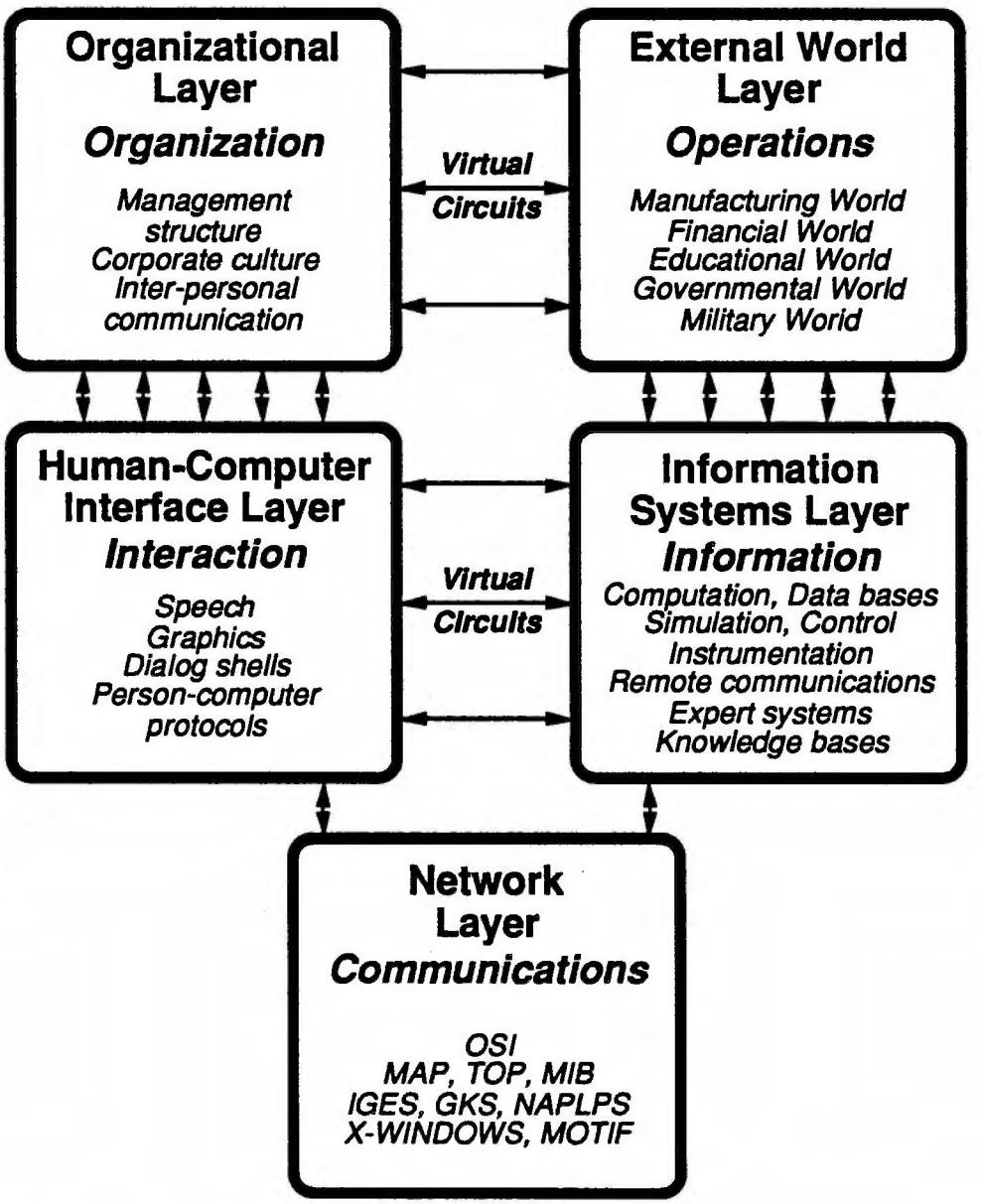


**Fig.6 Industrial knowledge support system integrating heterogeneous systems through a hierarchical protocol**

## CONCLUSIONS

It is difficult to know in writing a paper such as this whether one is helping to expedite system development by defining the overall problem, or whether the magnitude of what is defined is such as to undermine any confidence we may have had that the problems are tractable. It is sometimes better to go into a system development without analyzing requirements and potential problems in too great a detail—this is the way in which breakthroughs often occur. However, as a field matures the capability to produce quality systems consistently that meet requirements, including budgets, schedules and reliability, comes to depend on overt, clearly specified knowledge and well-defined prototypical architectures. The field of knowledge-based systems may become less exciting as this happens but its capability to fulfil its potential will greatly increase.

In terms of the slogans that arise out of the working groups at these meetings, this paper clearly has two, *social context* and *open systems*. Although the first relates to knowledge processes and the second to implementation technology, the two are also closely inter-related since it is the essential openness of an evolving society that generates the need for the openness in its supportive technology. Ultimately there is only one system that we are part of, that we are building, and that is building us.

## ACKNOWLEDGEMENTS

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada. I am grateful to many colleagues for discussions over the years that have influenced the research described in this paper, in particular John Boose, Jeff Bradshaw, Gordon Pask, Mildred Shaw, Alain Rappaport, Joan Vickers, Brian Woodward and other colleagues at the Knowledge Acquisition Workshops world-wide. I am particularly grateful to Mildred Shaw for her critical comments on this manuscript.

## REFERENCES

Barrett, E., Ed. (1989). **The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information.** Cambridge, Massachusetts: MIT Press.

Beninger, J.R. (1986). **The Control Revolution: Technological and Economic Origins of the Information Society**. Harvard: University Press.

Boose, J H. (1986) **Expertise Transfer for Expert System Design**. New York: Elsevier.

Boose, J H. (1989) A survey of knowledge acquisition techniques and tools. **Knowledge Acquisition 1** (1), 39-58 (March).

Boose, J.H. & Gaines, B.R., Eds. (1988). **Knowledge Acquisition Tools for Expert Systems**. London: Academic Press.

Boose, J.H. & Bradshaw, J.M. (1987) Expertise transfer and complex problems: using AQUINAS as a knowledge acquisition workbench for knowledge-based systems. **International Journal of Man-Machine Studies 26**, 3-28.

Breuker, J. & Wielinga, B. (1987) Use of models in the interpretation of verbal data. Kidd, A., Ed. **Knowledge Elicitation for Expert Systems: A Practical Handbook.** pp.17-44. Plenum.

Buchanan, B.G. & Shortliffe, E.H., Eds. (1984) **Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project**. Reading, Massachusetts: Addison-Wesley.

Cercone, N. & McCalla, G., Eds. (1987) **The Knowledge Frontier: Essays in the Representation of Knowledge**. New York: Springer.

Chandrasekaran, B (1988) Generic tasks as building blocks for knowledge-based systems: the diagnosis and routine design examples. **The Knowledge Engineering Review, 3**(3), 183-211.

Clancey, W.J. (1987a) **Knowledge-Based Tutoring: The GUIDON Program.** Cambridge, Massachusetts: MIT Press.

Clancey, W. J. (1987b). Intelligent tutoring systems: A tutorial survey. Van Lamsweerde, A. & Dufour, P. Eds. pp. 39-78. **Current Issues in Expert Systems.** London: Academic Press.

Clancey, W. J. (1987c). From GUIDON to NEOMYCIN and HERACLES in twenty short lessons. Van Lamsweerde, A. & Dufour, P. Eds. pp. 79-123. **Current Issues in Expert Systems.** London: Academic Press.

Clancey, W.J. & Shortliffe, E.H., Eds. (1984) **Readings in Medical Artificial Intelligence.** Reading, Massachusetts: Addison-Wesley.

Cullingford, R.E. (1986). **Natural Language Processing: A Knowledge Engineering Approach.** Totowa, New Jersey: Rowman & Littlefield.

Davis, R. (1982). TEIRESIAS: Applications of meta-level knowledge. Davis, R. & Lenat, D.B. **Knowledge-Based Systems in Artificial Intelligence.** pp.229-461. New York: McGraw-Hill.

Durkheim, E. (1893) **The Division of Labor in Society.** New York: Free Press.

Evans, D.A. & Patel, V.L., Eds. (1989). **Cognitive Science in Medicine: Biomedical Modeling.** Cambridge, Massachusetts: MIT Press.

Gaines, B.R. (1986a). Sixth generation computing: a conspectus of the Japanese proposals. **ACM SIGART Newsletter,** No.95, 39-44 (January).

Gaines, B.R. (1987a) An overview of knowledge acquisition and transfer. **International Journal of Man-Machine Studies 26** (4), 453-472 (April).

Gaines, B.R. (1987b). Rapid prototyping for expert systems. Oliff, M.D., Ed. **Intelligent Manufacturing: Proceedings from First International Conference on Expert Systems and the Leading Edge in Production Planning and Control.** pp.45-73. Menlo Park, California, Benjamin Cummins.

Gaines, B.R. (1987c) Advanced expert system support environments. Boose, J.H. & Gaines, B.R., Eds. **Proceedings of the Second AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop.** pp.8-0-8-14. Banff (October).

Gaines, B.R. (1988a). Second generation knowledge acquisition systems. **Proceedings of the Second European Workshop on Knowledge Acquisition for Knowledge-Based Systems (EKAW'88).** GMD-Studien Nr.143. pp.17-1-17-14. Bonn, Germany: Gellsellschaft für Mathematik und Datenverarbeitung (June).

Gaines, B.R. (1988b). Organizational integration: modeling technology and management. Zunde, P., Ed. **Empirical Foundations of Information and Software Sciences II.** New York: Plenum Press.

Gaines, B.R. (1988c). Positive feedback processes underlying the formation of expertise. **IEEE Transactions on Systems, Man & Cybernetics,** SMC-18(6), 1016-1020 (November).

Gaines, B.R. (1989a) Social and cognitive processes in knowledge acquisition. **Knowledge Acquisition 1** (1), 39-58 (March).

Gaines, B.R. & Boose, J.H., Eds. (1988). **Knowledge Acquisition for Knowledge-Based Systems.** London: Academic Press.

Gaines, B.R., Rappaport, A. & Shaw, M.L.G. (1989). Integration of acquisition and performance. Boose, J.H. & Gaines, B.R., Eds. **Proceedings of the Fourth AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop.** pp.25-1-25-20. Banff (October).

Gaines, B.R. & Shaw, M.L.G. (1986). A learning model for forecasting the future of information technology. **Future Computing Systems, 1**(1), 31-69.

Gaines, B.R. & Shaw, M.L.G. (1987). Knowledge support systems. **ACM MCC-University Research Symposium.** Austin, Texas: MCC. pp.47-66.

Gaines, B.R. & Vickers, J.N. (1988). Design considerations for hypermedia systems. **Microcomputers for Information Management**, 5(1), 1-27 (March).
Goodman, D. (1987). **The Complete HyperCard Handbook.** New York: Bantam.
Greif, I., Ed. (1988). **Computer Support of Cooperative Work: A Book of Readings.** San Mateo, California: Morgan Kaufmann.
Hawkins, D. (1983). An analysis of expert thinking. **International Journal of Man-Machine Studies, 18**(1), 1-47 (January).
Heim, M. (1987). **Electric Language: A Philosophical Study of Word Processing.** New Haven, Connecticut: Yale University Press.
Hiltz, S.R., Ed. (1988). **Online Communities: A Case Study of the Office of the Future.** Norwood, New Jersey: Ablex.
Hobbs, J.R. & Moore, R.C. (Eds.) (1985) **Formal Theories of the Commonsense World.** Norwood, New Jersey: Ablex.
Innis, H.A. (1972) **Empire and Communications.** Toronto: University of Toronto Press.
Keen, P.G.W. & Morton, M.S.S. (1978) **Decision Support Systems: An Organizational Perspective.** Reading, Massachusetts: Addison-Wesley.
Kitto, C.M. & Boose, J.H. (1987) Selecting knowledge acquisition tools and strategies based on application characteristics. Boose, J.H. & Gaines, B.R. (Eds) **Proceedings of the Second AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop.** pp.12-0-12-11. Banff (October).
Kolodner, J., Ed. (1988). **Proceedings Case-Based Reasoning Workshop.** San Mateo, California: Morgan Kaufmann.
Lehrberger, J. & Bourbeau, L. (1988). **Machine Translation: Linguistic Characteristics of MT Systems and General Methodology of Evaluation.** Amsterdam: John Benjamins.
McDermott, J. (1988) Preliminary steps toward a taxonomy of problem solving methods. Marcus, S., Ed. **Automating Knowledge Acquisition for Expert Systems.** pp. 225-266. Kluwer: Boston.
Meyrowitz, J. (1985) **No Sense of Place: The Impact of Electronic Media on Social Behavior.** New York: Oxford University Press.
Michalski, R.S., Carbonell, J.G. & Mitchell, T.M., Eds. (1986). **Machine Learning: An Artificial Intelligence Approach Volume II.** Los Altos, California: Morgan Kaufmann.
Musen, M. (1989). **Automated Generation of Model-Based Knowledge Acquisition Tools.** London: Pitman.
Oddy, R.N., Robertson, S.E., van Rijsbergen, C.J. & Williams, P.W. (Eds.) (1981). **Information Retrieval Research.** London: Butterworths.
Popper, K.R. (1968). Epistemology without a knowing subject. Van Rootselaar, B., Ed. **Logic, Methodology and Philosophy of Science III.** pp. 333-373. Amsterdam: North-Holland
Rappaport, A. (1987) Multiple-problem subspaces in the knowledge design process. **International Journal of Man-Machine Studies 26**(4), 435-452 (April).
Roberts, R.M. (1989). **Serendipity: Accidental Discoveries in Science.** New York: John Wiley.
Schön, D.A. (1983). **The Reflective Practitioner.** New York: Basic Books.
Sergot, M. (1988). Representing legislation as logic programs. Hayes, J.E., Michie, D. & Richards, J., Eds.Macine Intelligence 11. pp.209-260. Oxford: Clarendon Press.
Sharp, M. (1989). Knowledge acquisition for advice systems: FUSION, a network cognitive support system. **Proceedings of the Fourth AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop.** Banff (October).
Shaw, M.L.G. & Gaines, B.R. (1988). A methodology for recognizing consensus, correspondence, conflict and contrast in a knowledge acquisition system. **Proceedings of the Third AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop.** pp.30-1-30-19, Banff (November).

Shaw, M.L.G. & Woodward, B. (1989). Mental models in the knowledge acquisition process. **Proceedings of the Fourth AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop**. Banff (October).

Shortliffe, E.H. (1976). **Computer-Based Medical Consultations: MYCIN**. New York: Elsevier.

Shu, N.C., Ed. (1988). **Visual Programming.** New York: Van Nostrand Reinhold.

Slonin, J., Schonbach, A., Bauer, M.A., MacRae, L.J. & Thomas, K.A. (1987). **Building an Open System.** New York: Van Nostrand Reinhold.

Smith, D.C. (1977). **Pygmalion.** Basel: Birkhauser.

STA (1985). **Promotion of R&D on Electronics and Information Systems That May Complement or Substitute for Human Intelligence.** Tokyo: Science and Technology Agency.

Steels, L. (1986). Second generation expert systems. Bramer, M.A., Ed. **Research and Development in Expert Systems III.** pp.175-183. Cambridge: University Press.

Swartout, W. R. (1986). Knowledge needed for expert system explanation. **Future Computing Systems, 1**(2), 99-114.

Trigg, R. (1983). A Network-based Approach to Text Handling for the Online Scientific Community. PhD Thesis, University of Maryland (University Microfilms 8429934).

Trigg, R., Suchman, L. & Halasz, F. (1986). Supporting collaboration in NoteCards. **Proceedings of Conference on Computer Supported Cooperative Work**. Austin, Texas: MCC (December).

Van Lehn, K. (1985). Theory reform caused by an argumentation tool. **Xerox PARC Technical Report ISL-11**, 1-15. Palo Alto, California: Xerox Palo Alto Research Center (July).

Vervest, P.H.M. (1988). **Innovations in Electronic Mail.** Amsterdam: North-Holland.

Vliet, J.C., Ed. (1988). **Document Manipulation and Typography**. Cambridge, UK: Cambridge University Press.

Wenger, E. (1987). **Artificial Intelligence and Tutoring Systems.** Los Altos, California: Morgan Kaufmann.

Wojciechowski, J.A. (1986). Computing systems in an ecology of knowledge perspective **Future Computing Systems, 1**(4), 333-352.

Yankelovich, N., Landow, G. & Heywood, P. (1987) Designing hypermedia "ideabases"—the Intermedia experience. **IRIS Technical Report 87-4**. Providence, Rhode Island: Brown University.