Shaw, M. L. G. and Gaines, B. R. (1990). Integration issues in knowledge support environments. In Proceedings of the Seventh IASTED International Conference Hamza, M. H., ed, pp. 23-26, Los Angeles.

INTEGRATION ISSUES IN KNOWLEDGE SUPPORT ENVIRONMENTS

Mildred L. G. Shaw & Brian R. Gaines
Knowledge Science Institute
Department of Computer Science
University of Calgary
Calgary, Alberta
Canada T2N 1N4

ABSTRACT

Knowledge acquisition for knowledge-based systems has become a major research area in the last five years. The problem now is to make sense of the diverse and wide ranging material available and, in particular, to apply the results to improve effectiveness in the development of knowledge-based systems. There are a number of major impediments to such understanding and application including a diversity of techniques and tools that overlap in their applications where it is not clear whether they are competitive alternatives or complementary partners; a lack of variety, detail and evaluation in the case histories of applications of the techniques and tools; and a lack of access to the techniques and tools outside the narrow research communities originating them. Tools and techniques for knowledge acquisition for knowledge based systems need to be integrated with the overall system and not treated as separate components. The variety of sources of knowledge, representations and applications within the system, and user roles, makes such integration complex. The complexity raises many issues not generally thought of as part of knowledge acquisition yet fundamentally significant in extending and integrating acquisition tools. This paper gives an analysis of the structure of knowledge-based systems as it relates to the integration of acquisition. It is intended that this structure be used as a basis for system design and it will be extended to form a 'requirements specification' for knowledge based systems integrating acquisition with other aspects of their operation.

INTRODUCTION

Previous IASTED papers [5,6] have detailed the knowledge engineer's tasks in overcoming the famous bottleneck in the development of knowledge-based systems [2]. They have also given examples of Knowledge Support System Zero, which provides a highly interactive and graphic knowledge acquisition environment on the Apple Macintosh family of computers, and the use of this system in identifying differences between experts. The emphasis is now on the support environment, including the expert system shell in addition to the knowledge acquisition system, with a user community of experts, knowledge engineers and clients. The notion of a user community is important in drawing attention to another trend in knowledge-based systems, that the distinction between 'experts', 'knowledge engineers' and 'clients' is often one of variable roles rather than hard and fast job definitions. The expert may be a major client for the system, or may play a major part in knowledge engineering, or both. Some clients may customize or update the system in the role of knowledge engineers or extend its capabilities in the role of experts.

A major problem is not so much to access research and experience in knowledge acquisition but to make sense of the diverse and wide ranging material available and, in particular, to apply the results to improve effectiveness in the development of knowledge-based systems. There are a number of major impediments to such understanding and application:

- A diversity of techniques and tools that overlap in their applications and where it is not clear whether they are competitive alternatives or complementary partners
- Lack of variety, detail and evaluation in the case histories of applications of the techniques and tools
- Lack of access to the techniques and tools outside the narrow research communities originating them
- Lack of standardization in the knowledge representation resulting from the techniques and tools making it difficult to integrate them and to interface them to existing knowledge based systems
- Lack of standardization in the forms of data required by the techniques and tools making it difficult to apply them in the same situation and compare them
- Lack of standardization in the user interfaces to the interactive tools making it difficult to integrate them in an effective environment for human-computer interaction
- Lack of portability in the run-time environments required by the tools making it difficult to integrate them with other systems

These are not unreasonable problems at this stage of development of knowledge based systems. We are in the midst not only of the evolution of a rapidly developing technology but also in a revolution of our understanding of the nature of knowledge and knowledge processes in society and technology. Premature standardization resulting in over-rigid systems would be inappropriate and a major impediment to progress. However, we have to balance the need for integration, dissemination and application of knowledge acquisition techniques and tools against the dangers of rigidity. This paper addresses the issues involved, the evolutionary trends, the key requirements for integration and the practical approaches that may be taken.

INTEGRATION ISSUES IN THE USER COMMUNITY

In most current systems the knowledge acquisition tools, support tools and applications interface were designed separately and operate in very different ways on the assumption that they will be used by experts, knowledge engineers and clients, respectively. However, this diversity can be very confusing if users are flexible in their roles and switch between them. In addition, the complexity of the knowledge engineering role suggests that a variety of specializations will develop in building complex systems that will result in further roles and role switching. This leads to the design requirement:

Integration of User Roles: The user interface to a knowledge based system should be uniform and consistent across all user roles including expert, knowledge engineer and client.

However, it is important to note some features of the separation of roles:

Separation of Expert and Client Roles: The user interface to the clients should take into account the need to ensure that they use the terms in the same way as the expert. This may involve training procedures or the use of a different vocabulary.

The switch from manual to interactive elicitation has focused attention on the significance of the encoding process previously carried out tacitly by the knowledge engineer. Indeed, one of the problems in manual knowledge engineering may have been confusion between the knowledge representation meaningful to the expert and that usable by the expert system shell. A strong case has been made in manual elicitation for the use of mediating knowledge representations that are meaningful to the expert with encoding for the shell being treated as a separate phase [4]. Such a separation is analogous to that between high level languages and machine code in conventional data processing. It may be seen as a weakness of current shell support environments that a sufficiently high level of knowledge representation is not available. The transition from Lisp code through OPS5 production rules through frames to generalized objects may be seen as the evolution of higher level knowledge representations.

The sheer variety of possible representations of knowledge that are meaningful to people makes it unlikely that the gap between a natural representation in elicitation and a computable one after encoding will ever be completely closed. Hence, as before, design requirements capturing both integration and separation are required:

Integration of Elicitation and Encoding: The knowledge representation processible by the shell in a knowledge based system should be as directly meaningful as possible to an expert.

Separation of Elicitation and Encoding: Intermediate knowledge representations should be used in elicitation for a knowledge based system if the shell representation is not directly meaningful to an expert.

The knowledge engineer's interviewing role is significant even when interactive elicitation is being used. The knowledge engineer gains information in discussion that may be used to advise the expert in the use of the tools, used to set up particular tools for elicitation, or loaded directly into the system. The system should support these various processes where appropriate:

Integration of Manual and Interactive Elicitation: Make provision for the knowledge engineer to use knowledge elicited from the expert in a variety of ways to enhance the interactive elicitation process.

And it should not be assumed that completely automatic elicitation is always possible:

Separation of Manual and Interactive Elicitation: The reduction in load on knowledge engineers through interactive elicitation should be used to enable them to be more effective in ensuring that knowledge not captured interactively is still transferred to the system.

Elicitation from multiple experts raises many problems of dealing with different vocabularies, contradictions, and so on. The system must make provision, as does AQUINAS [1] both for integrating diverse sources of knowledge and for making contradictory sources of advice available if required. The design requirements are:

Integration of Multiple Experts: Provide tools for analyzing and expressing similarities and differences between different experts' knowledge to enable it to be combined when similar and a consensus to be negotiated when different.

Separation of Multiple Experts: When a consensus is not reached the system should be able to use contradictory sources of knowledge in generating advice by showing dissenting opinions.

These distinctions can be extended further. For example, there may be major differences between clients making a separation between interfaces appropriate. However, the major issues relevant to knowledge acquisition are captured in the design considerations discussed.

INTEGRATION ISSUES IN THE EXPERT SYSTEM SUPPORT ENVIRONMENT

The significance of the expert system support environment in providing tools for the knowledge engineer is apparent in the discussion above. The quality of this environment is very important in the development of expert systems, particularly in supporting the interaction between knowledge engineers and experts. The basic tools required are those for editing, displaying and validating the knowledge base. It is important that these tools present a uniform and consistent view of the knowledge base and that they aid consistency maintenance in the knowledge base during editing. The design requirements are:

Integration of Support Environment: Provide tools for editing, displaying and validating the knowledge base which are mutually self-consistent and, if possible, automatically propagate edits to maintain consistency or, otherwise, show clearly where parts of the knowledge base have become inconsistent.

The limited converse is that different perspectives should continue to be available.

Separation of Support Environment: Support whatever variety of perspectives on the knowledge base is necessary to ensure users in different roles have access that they find meaningful.

Validation is a complex topic in its own right and it is important that a variety of tools are provided to support different forms of validation. In particular, the objective validation of the knowledge base in terms of its performance and its subjective validation in comparison with experts should be supported. The design requirements are:

Integration of Sources of Validation: Provide tools for validating the knowledge base both in terms of its absolute performance and in comparison to experts.

Objective validation may require significant case histories that can be collected only from an operational system in actual applications.

Separation of Sources of Validation: The tools for subjective and objective validation may need to be substantially different and operate over different timescales.

There is a natural separation in subjective validation between experts' direct agreement with the knowledge base and correspondence between their behavior and that of the system based on the knowledge base. Both types of validation should be supported. The design requirements are:

Integration of Knowledge and Performance Evaluation: Provide tools for enabling experts to evaluate the knowledge base directly and for comparing system and expert behavior in advising clients.

It is possible for there to be contradictions between these two forms of evaluation.

Separation of Knowledge and Performance Evaluation: Take into account that there may be contradictions between experts' evaluations of knowledge and the resulting correspondence in performance.

INTEGRATION ISSUES IN KNOWLEDGE ACQUISITION

The significant features of the knowledge acquisition component best analyzed by considering the natures of the classical and expert system design paradigms. The integration of the classical systems analysis with expertise transfer techniques is a major task. The primary problems currently are in the widely different forms of knowledge representation usually involved, and these will be noted in a later section in terms of integration of conventional and knowledge-based system components. However, at a knowledge acquisition level the problem can be seen as one of the imbalance in the types of knowledge structures studied in depth to date. There has been an imbalance favoring symbolic knowledge to redress the past emphasis on quantitative methods. This needs redressing with the knowledge structures underlying arithmetic being integrated with those underlying non-arithmetic symbol manipulation. The design requirements are:

Integration of Systems Analysis and Expertise Transfer: Provide knowledge representation that is adequate to encompass quantitative and qualitative data and reasoning.

The success of the expert systems approach should not detract from the long term significance of building deep models of the system itself. The objective performance evaluation of expert systems already discussed can provide data on deeper issues. It corresponds to learning from experience in human experts.

Separation of Systems Analysis and Expertise Transfer: Provide for system analysis of performance data from expert systems in applications to improve their performance and the depth of their models.

The integration of various techniques for expertise transfer is already a theme at the knowledge acquisition workshops. For example, case history and text analysis may be used to provide terms and concepts that aid interviewing and behavior description. The design requirements are:

Integration of Expertise Transfer Techniques: Provide intermediate knowledge representations that allow for transfer between different sources of knowledge.

The limited converse is that the different sources of knowledge, and the possibilities of contradictions between them, should be recognized.

Separation of Expertise Transfer Techniques: Keep track of the different sources of knowledge within the system and allow for contradictions in inferences from it.

INTEGRATION ISSUES OVERALL

These issues can be organized into an overall analysis of the distinctions underlying integration issues in knowledge acquisition for knowledge based systems. It raises some additional integration issues. For example,a distinction can be made between the application and explanation components of the expert system shell. This is important in acquisition because the knowledge required for explanation may be more extensive and more demanding than that required for application. Explanation facilities in most shells are very weak currently but as they develop to take into account the more advanced analyses of the nature of explanation [3] the requirements for explanation-oriented knowledge elicitation will increase. The design requirements are:

Integration of Application and Explanation: Explanation should be treated as a major component of application and knowledge acquisition should be targeted accordingly.

In many areas such as validation there are major differences between application and explanation that need to be taken into account.

Separation of Application and Explanation: The need for separate validation of performance in application and explanation should be taken into account.

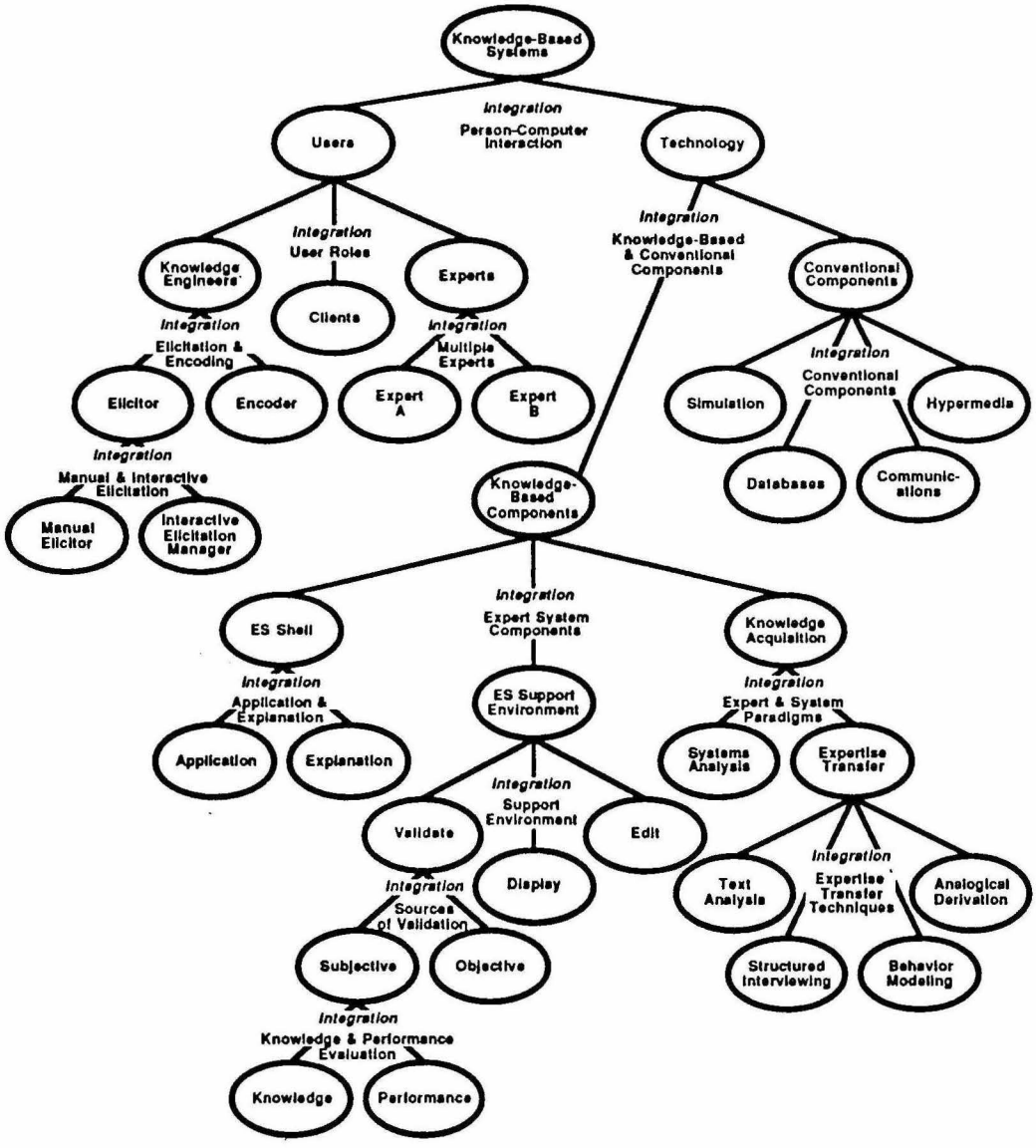

Fig.1 Overall structure of distinctions relating to knowledge acquisition integration in knowledge based systems

At the top of Figure 1 some of the conventional components of a data processing system have been introduced and a distinction has been made between them and the knowledge-based components. The term 'knowledge' has tended to be used in a specialized way in the knowledge based system context as 'knowledge which is symbolic and computationally processible.' Much significant knowledge does not come within this narrower definition yet is still important in knowledge based systems. This leads to a need to integrate several conventional data processing technologies, for example:

- *Simulation* is important in showing the time sequences of the behavior of dynamic systems
- *Databases* provide a convenient means for storing large quantities of data with efficient retrieval, and can be used to build a variety of object hierarchies in knowledge bases from different perspectives
- *Communications* is important in allowing interaction between experts at remote sites during group elicitation processes
- *Hypermedia* provide a means for storing and retrieving knowledge even if it is not computationally processible and, in particular, provide for knowledge transfer through video material allowing mimicking at a non-verbal level

The design requirements are:

Integration of Knowledge-Based and Conventional Components: Extend the concept of a knowledge-based system to include the knowledge structures of conventional data processing.

The only danger is that some essential features of knowledge based systems are lacking in existing forms of data processing.

Separation of Knowledge-Based and Conventional Components: Extend conventional data processing to include features of knowledge based systems such as explanation.

At the top of Figure 1 the final integration is between the users and technology, that is the achievement of effective person-computer interaction. This is a major problem in all systems of the complexity of that shown in Figure 1, and is already apparent in knowledge acquisition. The gap between functionality and usability can be very difficult to bridge, and it is easy to create multi-source, multi-format, multi-window operating environments that present great power but utter confusion to the user. The design requirements are:

Integration of Users and Technology: Provide a uniform and consistent user interface to the knowledge based system that maximizes the usability of the available functionality.

However, the interface must reflect reality and not be cosmetic.

Separation of Users and Technology: Do not use syntactic sugar and other interface tricks to give the user the impression of greater capabilities than exist in the system.

CONCLUSIONS

Tools and techniques for knowledge acquisition for knowledge based systems need to be integrated with the overall system and not treated as separate components. The variety of sources of knowledge, representations and applications within the system, and user roles, makes such integration complex. The complexity raises many issues not generally thought of as part of knowledge acquisition yet fundamentally significant in extending and integrating acquisition tools. This paper has given a preliminary analysis of the structure of knowledge-based systems as it relates to the integration of acquisition. It is intended that this structure be used as a basis for system design and it will be extended to form a 'requirements specification' for knowledge based systems integrating acquisition with other aspects of their operation.

ACKNOWLEDGEMENTS

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada. We are grateful to many colleagues at the knowledge acquisition workshops for discussions which have highlighted many of the issues raised in this paper.

REFERENCES

[1] Boose, J.H. & Bradshaw, J.M. Expertise transfer and complex problems: using AQUINAS as a knowledge acquisition workbench for expert systems. **International Journal of Man-Machine Studies 26(1)** 1987, pp. 3-28.

[2] Hayes-Roth, F., Waterman, D.A. & Lenat, D.B., Eds. **Building Expert Systems**. Reading, Massachusetts:Addison-Wesley, 1983.

[3] Hughes, S. Question classification in rule-based systems. Bramer, M.A., Ed. **Research and Development in Expert Systems III,** Cambridge: University Press, 1986, pp.123-131.

[4] Johnson, N. Mediating representations in knowledge elicitation. **Proceedings of the First European Workshop on Knowledge Acquisition for Knowledge Based Systems,** Department of Computer Science, Reading University, 1987, pp. A2-1-10.

[5] Shaw, M.L.G. A framework for knowledge acquisition from more than one expert. Hamza, M.H. (ed.) **Proceedings of the Third IASTED International Symposium on Expert Systems: Theory and Applications**, Calgary:Acta Press, 1988, pp. 76-79.

[6] Shaw, M.L.G. & Gaines, B.R. Extensions to knowledge acquisition methodologies and tools. Hamza, M.H. (ed.) **Proceedings of the Sixth IASTED International Symposium on Expert Systems: Theory and Applications,** Calgary:Acta Press, 1989, pp. 76-79.