Rapid Prototyping Techniques for Expert Systems

Mildred L G Shaw — Dept. of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4.

Jeffrey M. Bradshaw — Knowledge Systems Lab., Boeing Computer Services, P.O. Box 24346, Seattle, WA 98124.

Brian R Gaines — Dept. of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4.

John H. Boose — Knowledge Systems Lab., Boeing Computer Services, P.O. Box 24346, Seattle, WA 98124.

ABSTRACT

Recent developments in rapid prototyping tools for expert system development are described. KITTEN and AQUINAS are integrated prototyping systems providing knowledge acquisition tools encompassing a diversity of forms of knowledge and relationships between knowledge. KITTEN can access a wide range of knowledge sources including text, interviews with experts, and observations of expert behavior. AQUINAS can present knowledge from multiple sources with clarity as to its derivation, consequences and structural relations. Both systems can encompass a diversity of perspectives including partial or contradictory input from different experts. Users of these knowledge acquisition tools are able to apply the knowledge in a variety of familiar domains and freely experiment with its implications. These systems offer the capability to expedite the prototyping stage of complex knowledge-based systems, motivating the experts to be closely involved in all aspects of the system development by giving them a supportive and comprehensible environment.

THE IMPORTANCE OF RAPID PROTOTYPING

Problems of knowledge engineering have been recognized since the early days of expert systems. It was possible that knowledge engineering might develop as a profession on a par with systems analysis and programming, and that an initial shortage of skilled knowledge engineers would cause problems which would be overcome eventually as the profession developed. However, this scenario now appears less and less likely. There is certainly a shortage of knowledge engineers and problems in developing applications, but doubts have been cast on the notion that human labor is the appropriate solution to the knowledge engineering problem.

The technology is now in a mass-market situation where many organizations see the need for expert systems. This has led to a growth in demand that is far more rapid than the growth in supply of trained and experienced knowledge engineers. In addition, the role of the knowledge engineer as an intermediary between the expert and the technology is being questioned not only on cost grounds but also in relation to its effectiveness. Knowledge may be lost through the intermediary and the expert's lack of knowledge of the technology may be less of a detriment than the knowledge engineer's lack of domain knowledge. Full exploitation of the potential of expert systems depends on the development of rapid prototyping systems directly usable by experts with the knowledge engineer acting only as manager not intermediary.

We have developed programs, PLANET and ETS that have been widely used for some years for knowledge elicitation. This paper briefly reviews the techniques involved and describes the most recent developments of comprehensive integrated rapid prototyping systems, KITTEN and AQUINAS.

ELICITING EXPERT DISTINCTIONS: PLANET AND ETS

A growing number of knowledge acquisition tools incorporate methods derived from Personal Construct Theory [22, 26]. Such tools include the Expertise Transfer System (ETS), [1-4], PLANET [26, 28, 29], AQUINAS [5, 6, 23], FMS Aid [20], KITTEN [30], Kriton [10, 11], KSSO [15,16] and others. Because of their reliance on personal construct methodology, these tools share a number of powerful strategies for knowledge acquisition, refinement, testing, and debugging. Because of the way the knowledge is gathered and structured, very little training is required to use these tools effectively. Persons can operate in an exploratory mode, freely experimenting with different ways of organizing their problem-solving knowledge and immediately seeing the implications of these changes graphically.

The key knowledge that experts use is their conceptual framework. This enables them to categorize the world of their expertise, and classify their experiences in such a way as to anticipate future events and act upon them. The distinctions experts make among their observations and experiences are crucial to the formation and validation of this conceptual framework, and enable them to build conceptual models of the whole situation. Distinctions and terminology, the names they give to these distinctions, may be elicited by presenting experts with sample items (or elements) to distinguish.

Consider a small appliance repairman who is building a system for diagnosing component faults. In distinguishing three faults (**gear-bad**, **power-supply-bad**, and **wiring-bad**) we might ask in what way are two alike and thereby different from the other one? We might first of all say that **power-supply-bad**, and **wiring-bad** are alike since they are symptomized by **no power**, whereas if **gear-bad** is the fault the **power is normal**. The distinction that underlies the dimension **no power/normal power** is termed a construct.

A scale allowing more distinctions may be used as required. By allowing an expert to assign each fault a rating value between, say, 1 (no power) and 5 (normal power) we can represent situations where power may be **weak** but not absent. Increased precision is possible by allowing the numeric scale on which each fault is rated to represent actual voltages, rather than a simple ordinal value.

The mapping of the faults onto the dimensions produces a two-dimensional grid of relationships which can be represented as a numeric data structure as used in PLANET and ETS (Figure 3). This structure may be viewed as a component of a database in entity-attribute form: a repertory grid has elements as entities, constructs as attributes and allocations of elements to locations of construct dimensions as values.

PLANET [28, 29] is a set of programs for eliciting personal knowledge systems from a variety of perspectives and then comparing and contrasting the systems elicited. It runs as a menu-driven suite of interactive programs on a variety of computers. It contains programs for: distinction elicitation with on-line feedback; hierarchical cluster analysis; principal component analysis; logical analysis in terms of inference rules; comparison of perspectives from different experts; the production of socionets linking similar construing within the group; and a mode grid showing the distinctions which are readily understood by the majority of the group. PLANET has been widely distributed internationally for over six years and used for educational, clinical and management studies as well as prototyping expert systems. Figure 1 shows the relationship of the programs in PLANET.

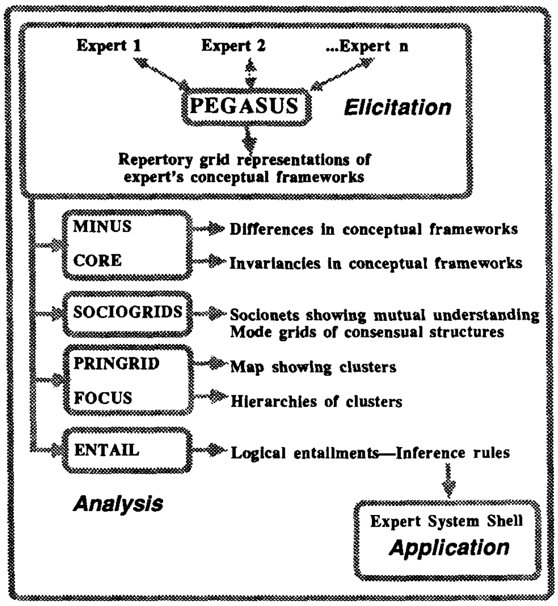

Fig. 1 Knowledge engineering facilities available in PLANET

The Expertise Transfer System (ETS) extends some of the techniques used in PLANET, and also contains a reasoning engine that allows the expert to test the grid knowledge incrementally by running consultations. The system has been in use in Boeing for more than three years and hundreds of prototypical knowledge-based systems have been generated by ETS. The system interviews experts to uncover key aspects of their problem-solving knowledge. It helps build very rapid prototypes (typically in less than two hours), assists the expert in analyzing the adequacy of the knowledge for solving the problem, and creates knowledge bases for several expert system shells (S.1, M.1, OPS5, KEE, and so on) from its own internal representation [1-4]. Figure 2 shows the naming of the expert's distinctions and some of the analysis.

INTEGRATED PROTOTYPING SYSTEMS: KITTEN AND AQUINAS

KITTEN [30] and AQUINAS [5] are major extensions of PLANET and ETS to provide knowledge acquisition tools which can encompass a diversity of forms of knowledge and relationships between knowledge. KITTEN can access a wide range of knowledge sources including text, interviews with experts, and observations of expert behavior. AQUINAS can present knowledge from multiple sources with clarity as to its derivation, consequences and structural relations. Both systems can encompass a diversity of perspectives including partial or contradictory input from different experts. Users of these knowledge acquisition tools are able to apply the knowledge in a variety of familiar domains and freely experiment with its implications.

Figure 3 shows the structure of KITTEN (Knowledge Initiation & Transfer Tools for Experts and Novices). KITTEN consists of a: knowledge base; various analytical tools for building and transforming the knowledge base; and a number of conversational tools for interacting with the knowledge base. The KITTEN implementation is written in Pascal and currently runs on a network of Apollo computers.

In addition to expert interviewing, KITTEN contains facilities for eliciting distinctions from text input or protocols. Text input of a manual or text book may be analyzed through the TEXAN

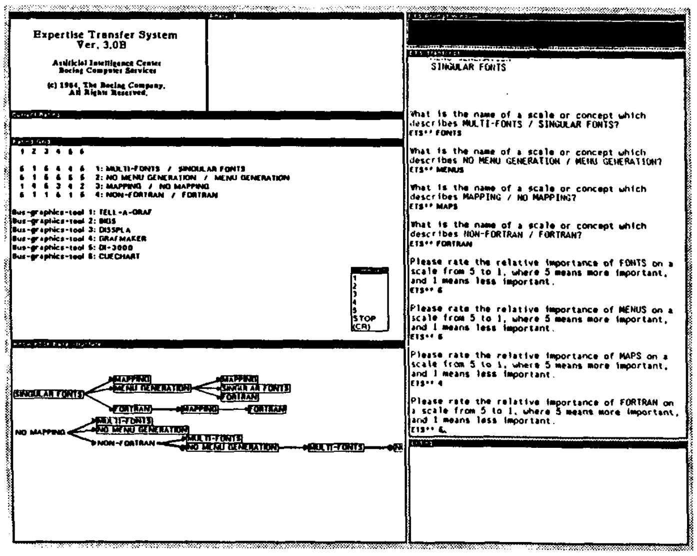

Fig. 2 Naming distinctions and rating their importance in ETS

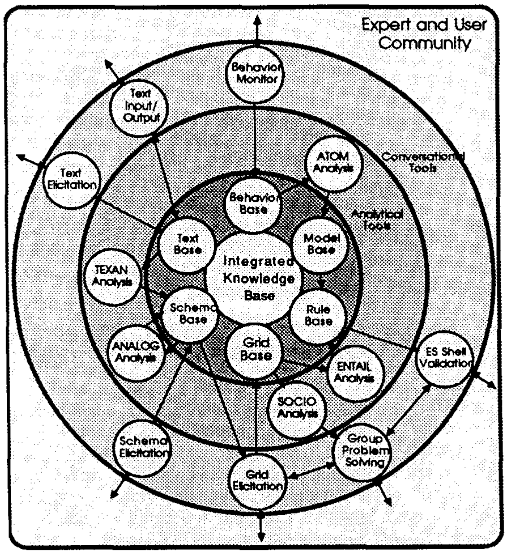

Fig. 3 KITTEN — Knowledge Initiation and Transfer Tools for Experts and Novices

procedure which clusters associated words. The text is fully indexed by all non-noise words grouped by their stems, and a coupling matrix of word associations is calculated using a simple distance-in-text measure. This leads to a schema from which the expert can select related elements and initial constructs with which to commence grid elicitation. Once a grid has been built, it can be analyzed by the ENTAIL procedure [13] which interprets the distinctions as fuzzy predicates. Asymmetric implications are derived so that one can infer how a distinction might be applied to a new item given other distinctions [17]. These provide rules for input to an expert system shell [19, 2, 3].

An alternative route is to monitor the expert's behavior through a verbal protocol giving information used and the resulting decisions. This protocol is analyzed through the ATOM procedure which induces structure from behavior using a search over a model space ordered by complexity and goodness of fit. From this data, production rules may be generated [14].

Data gathered from expert interviewing, text analysis, and protocol analysis can be combined within a single framework. KITTEN attempts to make each stage as explicit as possible. The knowledge base is presented to the expert in a more intelligible form than in typical rule-based systems.

Rules generated by KITTEN may be loaded into the commercial shell Nexpert [25]. Nexpert can display the rule base in unique graphical presentations that enable the expert to see the impact of different fragments of knowledge. The logging and explanation facilities of Nexpert allow the expert to track down and revise spurious inferences that may arise with the rules derived by KITTEN.

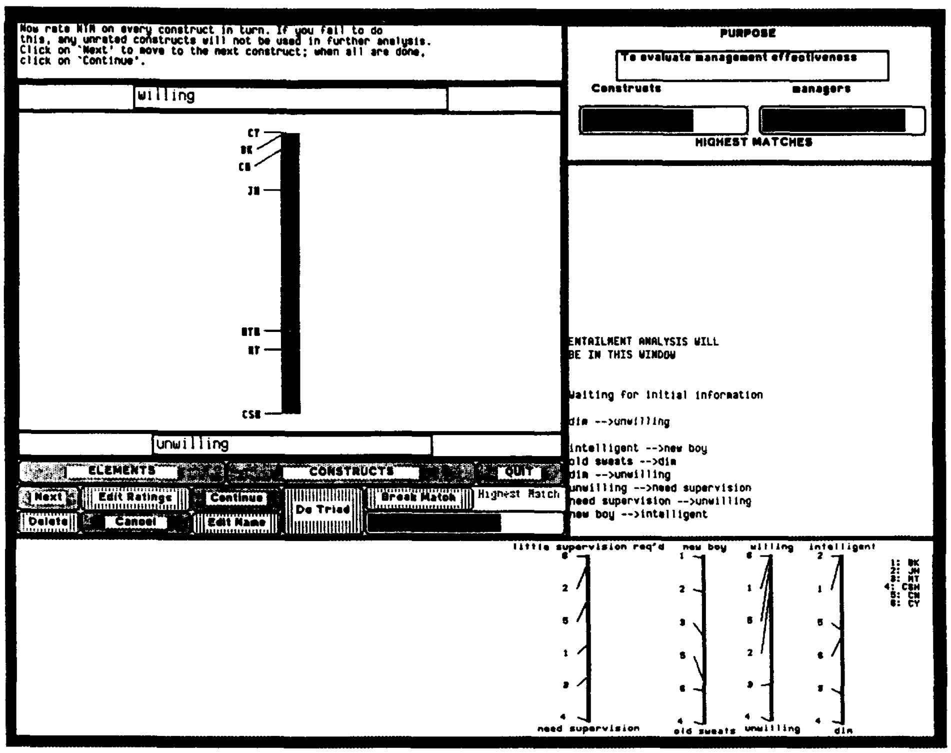

Fig. 4 A KITTEN screen

The group problem-solving component of KITTEN is particularly important because it goes beyond the stereotype of an "expert" and "users", and allows the system to be used to support an interactive community in their acquisition and transfer of knowledge and mutual understanding. Much expertise only resides within the social context of cooperating individuals and requires elicitation across the group. The SOCIO analysis program supports group elicitation techniques in which the conceptual frameworks of a number of experts are compared and allows members of a community to explore their agreement and understanding with other members, and to make overt the knowledge network involved [26, 27]. For example, **exchange** grids can be used in which A attempts to "see the world" through the eyes of B by rating B's elements on B's constructs either as A would have done (to explore **agreement**) or as he thinks B did (to explore **understanding**).

The KITTEN implementation is an initial prototype offering a workbench with minimal integration of the knowledge base, but each of tools has already proven effective, and their combination is proving very powerful in stimulating experts to think of the knowledge externalization process from a number of different perspectives. Figure 4 shows a distinction being elicited using the natural click-and-drag techniques of the window and mouse interaction.

AQUINAS is organized around a collection of integrated tool sets that share a common user interface and underlying knowledge representation and data base. It is written in Interlisp and runs on the Xerox family of Lisp machines. Subsets of AQUINAS also run in an Interlisp version on the DEC Vax and a C/UNIX-based portable version. The AQUINAS screen is divided into a typescript window, map windows showing hierarchies, rating grid windows, and analysis windows as shown in Figure 5. Experts interact with AQUINAS by text entry or by mouse through pop-up menus.

It is often difficult to represent complex problems in a single grid. Hierarchical tools in AQUINAS help the expert build, edit, and analyze knowledge in hierarchies and lattices. These allow the expert to break up complex problems into chunks of convenient size and similar levels of abstraction. Hierarchies in AQUINAS are organized around *solutions*, *traits*, *knowledge sources* (experts or external databases), and *cases*. Nodes in the four hierarchies combine to form rating grids. The expert defines the current rating grid by selecting appropriate nodes in the hierarchies. In the most simple case, the children of a node in a solution hierarchy supply the solutions along the top of a grid; the children of a node in a trait hierarchy supply the traits down the side of a grid as shown in Figure 5.

A major limitation of most current knowledge engineering tools is that they do not allow experts to specify how specific pieces of knowledge should be combined [21]. With respect to evidence combination, most tools either tend to use fixed, global numeric functions or restrict the expert to purely symbolic representations of uncertainty. In the domain of preferences, most tools propagate preferences heuristically, making the explicit representation of tradeoffs next to impossible. In AQUINAS, the structured nature of the knowledge base allows knowledge to be combined using locally-specifiable applications of several different methods (e.g., probabilistic, certainty-factor calculus for evidence; Analytic Hierarchy Process, absolute reasoning for preferences; Bradshaw & Boose, in press). Experts select methods based on the cost of elicitation, the precision of the knowledge needed, convenience, and the expert's preference.

A mixed-initiative reasoning engine within AQUINAS supports consultations for heuristic classification problems. Many such problems may be solved by abstracting data, heuristically mapping higher level problem descriptions onto solution models, and then refining these models until specific solutions are found [9, 8]. In AQUINAS, data abstraction is carried out within hierarchies of traits, and solutions are refined as information is propagated through solution hierarchies [5, 6].

Knowledge acquisition tools can increase their leverage by suggesting appropriate expansions and refinements of the knowledge based on partial information already provided by the expert [12]. For large knowledge bases it would be desirable to derive many of the rating values rather than require the expert to assign each one directly. This is done in AQUINAS by propagating values and evidence through the hierarchies. The hierarchical organization of AQUINAS allows users to specify and weight knowledge sources. The reasoning engine uses this knowledge to give consensual and dissenting opinions.

The AQUINAS *dialog manager* makes decisions about general classes of actions and then recommends one or more specific actions providing comments and explanation if desired. This knowledge is represented as rules within the dialog manager. The hierachical structuring of the knowledge base allows the dialog manager to perform sophisticated debugging. A session history is recorded so that temporal reasoning and learning may be performed [23].

COMPARISON WITH OTHER SYSTEMS

MOLE [12] is an expert system shell that has been used effectively to build heuristic classification systems. Like KITTEN and AQUINAS, it interviews domain experts, exploits assumptions of exhaustiveness and exclusivity to determine the most likely candidates among competing hypotheses, and has many methods for analyzing and refining the knowledge base for consistency and adequacy. It distinguishes between three types of knowledge—covering, differentiating, and combining knowledge—in its problem-solving, and uses a system of "support values" to represent uncertainty.

CSRL [8] is another successful tool with a similar problem-solving approach. It handles a wider range of problem structures than are currently practical in KITTEN and AQUINAS. It is, however, more difficult for domain experts to learn and use. Gruber and Cohen [21] have developed a system called MUM that embodies architectural principles that facilitate knowledge acquisition. Like AQUINAS, it allows experts to specify local evidence combining knowledge; however it is limited to a symbolic representation of uncertainty. It allows a more flexible representation of control knowledge than is possible in current versions of KITTEN and AQUINAS.

An innovative learning apprentice program, ODYSSEUS, has been developed by Wilkins [31] to help experts refine and debug knowledge bases for the HERACLES heuristic classification shell. It addresses a task similar to that of the dialog manager in AQUINAS, but is quite different in its approach. Unlike the dialog manager, its use is appropriate only during the knowledge acquisition "end-game"; that is, it requires that a reasonable knowledge base has already been created.

Presently KITTEN and AQUINAS work best on those problems whose solutions can be comfortably enumerated such as analytic or structured selection problems such as classification or diagnosis, as opposed to problems whose solutions are built up from components such as synthetic or constructive problems like configuration or planning. SALT [24] is a particularly promising system that has been used effectively to build knowledge bases for certain types of constructive problems —its first use was to configure elevators.

Facilities for combining knowledge from multiple sources and for simultaneous group problem solving are unique to KITTEN

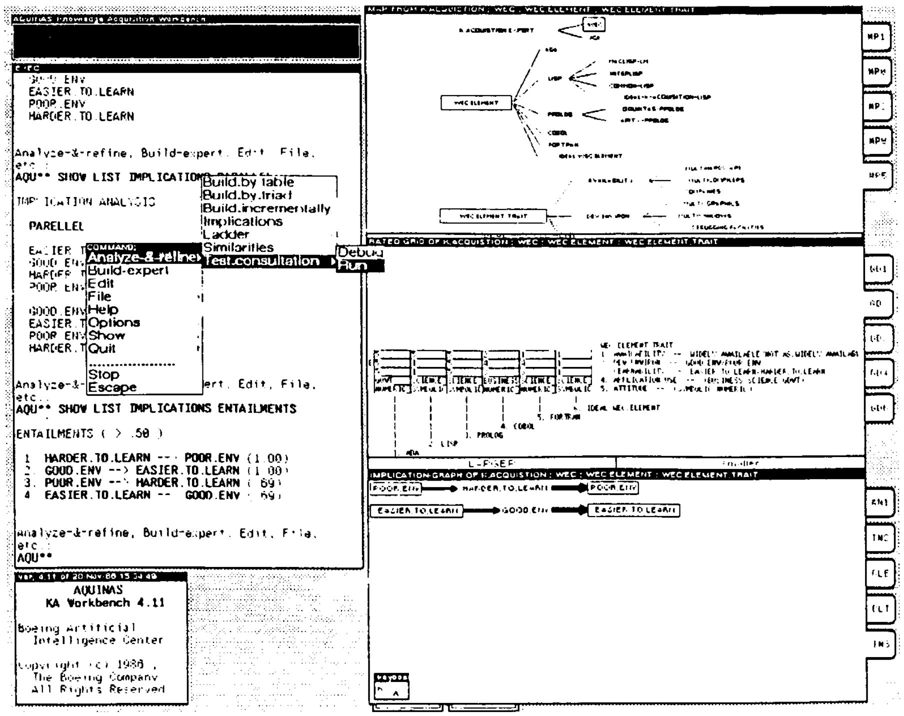

Fig. 5 An AQUINAS screen

and AQUINAS. These tools are also unique in the level of support they provide in the selection and definition of useful domain attributes in the initial exploratory stages of model building.

CONCLUSIONS

The initial generation of rapid prototyping tools for expert system development, represented by PLANET and ETS, elicited an expert's conceptual structure, and has proved useful in a wide range of practical developments. However, this generation was limited to cohesive sub-domains and single experts, proving most effective for small structured selection problems. The next generation, represented by KITTEN and AQUINAS, allows a diversity of sources to be used, including text, multiple experts, and decision-making behavior; structures the elicited material in hierarchies; and integrates tools and structures more closely with expert system shells. These systems offer the capability to expedite the prototyping stage of complex knowledge-based systems, motivating the experts to be closely involved in all aspects of the system development by giving them a supportive and comprehensible environment.

ACKNOWLEDGEMENTS

Financial assistance for the KITTEN work has been made available by the National Sciences and Engineering Research Council of Canada. AQUINAS was developed at the Knowledge Systems Laboratory, Advanced Technology Center, Boeing Computer Services in Seattle, Washington; thanks to Roger Beeman, Miroslav Benda, Kathleen Bradshaw, William Clancey, Cathy Kitto, Ted Kitzmiller, Art Nagai, Doug Schuler, David Shema, Lisle Tinglof-Boose, and Bruce Wilson for their contributions and support.

REFERENCES

[1] Boose, J.H. (1984). Personal construct theory and the transfer of human expertise. **Proceedings AAAI-84**, 27-33. California: American Association for Artificial Intelligence.

[2] Boose, J.H. (1985). A knowledge acquisition program for expert systems based on personal construct psychology. **International Journal of Man-Machine Studies 20**(1), 21-43 (January).

[3] Boose, J.H. (1986a). **Expertise Transfer for Expert Systems.** Amsterdam: Elsevier.

[4] Boose, J.H. (1986b). Rapid acquisition and combination of knowledge from multiple experts in the same domain. **Future Computing Systems 1**(2), 191-216.

[5] Boose, J.H. & Bradshaw, J.M. (1987a). 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.

[6] Boose, J. H. & Bradshaw, J. M. (1987b). AQUINAS: A knowledge acquisition workbench for building knowledge-based systems. **Proceedings of the First European Workshop on Knowledge Acquisition for Knowledge-Based Systems,** Reading, England.

[7] Bradshaw, J.M. & Boose, J.H. (in press). Decision analytic techniques for knowledge acquisition: Combining information and preferences using Aquinas. **International Journal of Man-Machine Studies, 27.**

[8] Bylander, T. & Mittal, S. (1986). CSRL: A language for classificatory problem-solving and uncertainty handling, **AI Magazine** (August).

[9] Clancey, W. (1986). Heuristic classification. J. Kowalik, Ed., **Knowledge-Based Problem-Solving**, New York: Prentice-Hall.

[10] Diederich, J., Ruhmann, I. & May, M. (1987). KRITON: A knowledge acquisition tool for expert systems. **International Journal of Man-Machine Studies, 26**(1).

[11] Diederich, J., Linster, M., Ruhmann, I. & Uthmann, T. (1987). A methodology for integrating knowledge acquisition techniques. **Proceedings of the First European Workshop on Knowledge Acquisition for Knowledge-Based Systems,** Reading, England.

[12] Eshelman, L., Ehret, D., McDermott, J. & Tan, M. (1987). MOLE: A tenacious knowledge acquisition tool. **International Journal of Man-Machine Studies,** in press.

[13] Gaines, B.R. (1976). Behaviour/structure transformations under uncertainty. **International Journal of Man-Machine Studies, 8**(3), 337-365 (May).

[14] Gaines, B.R. (1977). System identification, approximation and complexity. **International Journal of General Systems, 3**, 145-174.

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

[16] Gaines, B. R. (1987b) How do experts acquire expertise? **Proceedings of the first European Workshop on Knowledge Acquisition for Knowledge-based Systems,** Reading, England.

[17] Gaines, B.R. & Shaw, M.L.G. (1981). New directions in the analysis and interactive elicitation of personal construct systems. Shaw, M.L.G., Ed. **Recent Advances in Personal Construct Technology.** 147-182. London: Academic Press.

[18] Gaines, B.R. & Shaw, M.L.G. (1981). A programme for the development of a systems methodology of knowledge and action. Reckmeyer, W.J., Ed. **General Systems Research and Design: Precursors and Futures**. 255-264. Society for General Systems Research (January).

[19] Gaines, B.R. & Shaw, M.L.G. (1986). Induction of inference rules for expert systems. **Fuzzy Sets and Systems,** 8(3), 315-328 (April).

[20] Garg-Janardan, C. & Salvendy, G. A conceptual framework for knowledge elicitation. **International Journal of Man-Machine Studies, 27(1).**

[21] Gruber, T. & Cohen, P. (1987). Design for acquisition: Principles of knowledge system design to facilitate knowledge acquisition. **International Journal of Man-Machine Studies,** in press.

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

[23] Kitto, C. & Boose, J.H. (1987). Heuristics for expertise transfer: The automatic management of complex knowledge acquisition dialogs. **International Journal of Man-Machine Studies,** in press.

[24] Marcus, S. (1987). Taking backtracking with a grain of SALT. **International Journal of Man-Machine Studies,** in press.

[25] Rappaport (1987). Multiple problem spaces in the knowledge design process. **International Journal of Man-Machine Studies,** in press.

[26] Shaw, M.L.G. (1980). **On Becoming a Personal Scientist**. London: Academic Press.

[27] Shaw, M.L.G. (1981). Conversational heuristics for eliciting shared understanding. Shaw, M.L.G., Ed. **Recent Advances in Personal Construct Technology**, 31-44. London: Academic Press.

[28] Shaw, M.L.G. (1982). PLANET: some experience in creating an integrated system for repertory grid applications on a microcomputer. **International Journal of Man-Machine Studies, 17**(3), 345-360.

[29] Shaw, M.L.G. & Gaines, B.R. (1986). Interactive elicitation of knowledge from experts. **Future Computing Systems, 1**(2), 151-190.

[30] Shaw, M.L.G. & Gaines, B.R. (1987). Techniques for knowledge acquisition and transfer. **International Journal of Man-Machine Studies,** in press.

[31] Wilkins, D.C. (1987). Knowledge base debugging using apprenticeship learning techniques. **International Journal of Man-Machine Studies,** in press.