Modeling as Framework for Knowledge Acquisition Methodologies and Tools

Brian R. Gaines, Mildred L. G. Shaw, and J. Brian Woodward
Knowledge Science Institute, University of Calgary, Calgary, Alberta, Canada T2N 1N4

This article develops a classification of the sources and types of models developed in knowledge engineering, and uses it to provide a framework within which knowledge acquisition methodologies and tools can be discussed and analyzed. Much of the early work on knowledge acquisition assumed that human expertise is based on "mental models" of domains and problem-solving techniques, and that these can be elicited and transferred to an expert system. The approach taken here is to focus instead on the knowledge engineer's modeling process, his or her conceptual models of systems associated with the expert's skill, and their sources and types. This leads to a comprehensive account of knowledge-based system development encompassing classical systems analysis, cognitive processes, linguistic representations, and the formalization of knowledge for computer application. © 1993 John Wiley & Sons, Inc.

I. INTRODUCTION

The literature on knowledge acquisition has grown explosively in the past decade as have the number of knowledge acquisition methodologies and tools.[1] Research and practice in knowledge acquisition is based on a "patchwork" of methods and techniques, imported from other disciplines and adapted pragmatically.[2] Often a technique has been imported to address a specific issue or problem and then its use has been generalized for other purposes. The sheer diversity of methodologies and associated theoretical framework and literatures is daunting to practitioners, theorists, and tool developers. One approach to coping with such diversity is to focus on one part of it and attempt to use, found, and support a single approach applicable to a wide range of requirements. Another approach is to attempt to combine a diversity of techniques within a "toolbox" framework, changing representations as required and supporting the management of a complex, heterogeneous process. Yet another approach is to attempt to develop an all encompassing theoretical framework and systems architecture that provides an overall "clean" solution.

All these approaches are valid in some sense, and all of them have both theoretical and practical limitations deriving from the essential anarchy of hu-

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 8, 155–168 (1993)
© 1993 John Wiley & Sons, Inc. CCC 0884-8173/93/020155-14

man knowledge processes.[3] What would be very useful at the current state of the art is to have some framework that provides a range of perspectives on the issues involved and the different approaches possible, within a reasonably unified system, without itself attempting the impossible of a general, in-depth, and complete theory. This article examines one possible such framework based on the analysis of the sources and types of conceptual model involved in the knowledge engineering process.

Clancey[4] has suggested that a common factor underlying all expert systems is that they contain qualitative world models, and that we can gain insights into the structure of knowledge bases and knowledge engineering by classifying the types of models involved. This suggests that a classification of the sources and types of models developed in knowledge engineering may be used to provide a framework within which knowledge acquisition methodologies and tools can be discussed and analyzed. A reasonable objective is to provide as unified a basis for discussion and comparison as possible, while not distorting essential differences that themselves reflect the diversity of human knowledge processes. In particular, the framework developed should encompass both "humanistic" approaches to knowledge acquisition, focusing on the elicitation of knowledge from human experts, and the "mechanistic" approaches focusing on the system analysis and software engineering involved in developing knowledge-based systems.

The approach taken in this article attempts to subsume and extend previous frameworks for knowledge acquisition rather than to show them "right" or "wrong," or develop different alternatives. Much of the early work on knowledge acquisition seems to be based on the presupposition that human expert performance is based on the existence of "mental models" of domains and problem-solving techniques, and that these can be elicited and transferred to an expert system. The following section addresses this preconception and the associated literature. The next section broadens the discussion to include the knowledge engineer's modeling process, his or her conceptual models of systems associated with the expert's skill, and their sources and types. The next section further develops this framework to structure these models and modeling processes within a hierarchical structure. This is then used as a framework for the discussion and characterization of various knowledge acquisition issues, methodologies, tools, and systems.

II. MENTAL MODELS

It is tempting to generalize from expert systems that are computer-based to the "expert systems" that are human experts, and extend Clancey's suggestion that all expert systems contain models, to suppose that human experts "contain models." From this perspective one might view the replication of human expertise in a knowledge-based system as involving the elicitation of the *mental models* of the human experts involved.[5] It is presumably this viewpoint that led to the metaphor of *expertise transfer* as the function of knowledge acquisition tools in the early years of knowledge engineering.[6]

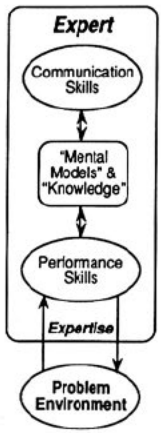

Figure 1. Modeling expertise.

However, this perspective can be highly misleading since the nature and existence of mental models are themselves subject to debate. For example, Clancey himself would not accept the generalization above. He has argued against the expertise transfer viewpoint, seeing knowledge as a capacity to behave adaptively that cannot be reduced to representations.[7] In the context of visual imagery, Pylyshyn[8] has cast doubt on both the empirical evidence for the existence of certain types of mental model in perception, and on the possibility of such evidence. In a practical context, Dreyfus and Dreyfus[9] have argued that it is precisely the lack of correspondence between the basis for human skilled performance and its representation in communicable models that imposes the most severe constraints on developments in artificial intelligence.

Figure 1 illustrates these issues in terms of the role of the expert as a carrier of some performance skills whose emulation in a computer is desired. The existence of the capability for skilled performance is taken as a given, but the existence of the other processes shown within the expert is presuppositional. We may impute the existence of "mental models" and "knowledge" and suppose that the expert may have the communication skills to discuss them. We may be concerned to elicit the imputed structures without distortion, or that the expert does not have the vocabulary to describe them. However, this very framework and meta-vocabulary is itself based on existence assumptions that, if they are incorrect, may undermine the whole basis of our approach to knowledge acquisition. The representations made by the knowledge engineer are not isomorphic to structures in the mind of the expert.[10] For knowledge acquisition purposes it is misleading to assume that our tools "capture" or "transfer" the knowledge of the expert.

Norman[11] has addressed these issues by analyzing mental models not only within the context of a person performing a task but also within the context of the observer of both person and task. He notes that a psychologist's understand-

ing of expertise involves his or her own mental models of the task and of the mental model of the person performing it. He suggests that some of the problems in the transfer of knowledge can best be viewed in terms of the mental model that the person answering questions has of the person asking them. That is, we are dealing with multiple levels of interacting models involving the mental processes of both experts and knowledge engineers (and the presuppositions of tool designers if we attempt to take the knowledge engineer out of the loop).

Norman distinguishes as *conceptual models* those mental models that are concerned with communicating with other entities, and are represented in the communication primitives of language and behavior. The primary mental models affect the form and representation of the conceptual models and are, in turn, constrained by the language and behavioral representational system. It is the conceptual models that are available to others and which require further translation and representation by other modeling activities. As the mental models are unavailable to an observer, the conceptual models are taken to represent the models used to address the task demands of the target system. The conceptual models become the focus of further modeling efforts.

Within this framework, one can view knowledge engineers, or automated knowledge acquisition systems interacting with the expert, as accessing and developing the expert's conceptual models—that is, of those models which the expert imputes as underlying his or her expert behavior. Some parts of these models may have already been developed, particularly if the expert has a teaching role, but other parts will come into being as a result of the knowledge acquisition process. Thus this process is by no means an unbiased transfer of a mental model, or even of a conceptual model of an imputed mental model. It is itself a modeling process that provides its own framework through its underlying meta-models, which are often implicit in the elicitation procedures. The resultant conceptual models provide a set of data which is used in various analysis processes which, in turn, are directed by meta-models of their own.

III. CONCEPTUAL MODELS

The distinction that Norman introduces between mental models and conceptual models, and the dubious status of mental models in themselves, suggests that a useful framework for the analysis of knowledge engineering may be developed through the analysis of the sources and types of conceptual model available to the knowledge engineer rather than focusing only on the mental processes underlying expertise. The situation of the introspective expert who can communicate his or her "knowledge" well, may be treated as one where the "knowledge engineering" and "expert" roles are operating effectively together within the same person. The situation of the expert from whom knowledge is being "elicited" actually building a new model of the basis of his or her skills through the process of elicitation may be treated as one where the conceptual model is developed as part of the process of knowledge engineering. In adopting the conceptual modeling perspective we do not exclude previous viewpoints, but rather supplement them with complementary perspectives.

For example, the paradigm shift from a classical system analysis perspective to that of an expert systems approach is readily analyzed in terms of what is being modeled in Fig. 1, the problem environment or the performance skill. Classical systems analysis emphasizes the formation of observed system models, that is in terms of Fig. 1, the formation of a model whose functional dependency is:

Problem Environment Output = System Model (Problem Environment Input)

The expert systems approach might be seen to be characterized by developing the inverse model whose functional dependency is:

Problem Environment Input = Task Model (Problem Environment Output)

That is, we model the decision making or control system rather than the system about which decisions are being made.[12]

In the early days of expert systems development, it was assumed that the direct communication of knowledge between expert and knowledge engineer was the preferred method. In a classic experiment, Michalski and Chilausky[13] showed that this was not necessarily so and inductive behavior modeling, in which the expert is observed in action and his or her activities modeled, may lead to a better knowledge-based system. This is an example of the second paradigm above, of modeling the expert as opposed to modeling the system. However, it is rarely a purely behavioral paradigm since the knowledge engineer may not be able to discriminate the inputs that the expert is using and will normally rely on verbal reporting by the expert for a description of the inputs and outputs. It should also be noted that the models of system and skills obtained through input–output modeling are mutually contaminated—we model the system in the context of an expert, and the expert skill in the context of a particular situation. Generalizing either model may lead to erroneous conclusions.

It is customary in expert system development, to assume that the expert has already constructed such models or may be in a privileged position to do so through self-observation and introspection, and these may be elicited by direct communication between knowledge engineer and expert. Additionally, the knowledge engineer may derive models from other experts, from the literature, and from the application of principles allowing performance skills to be derived from deep knowledge. The final knowledge-based system development involves the synthesis of these many models and the encoding of them to become an operational knowledge-based system emulating the desired expertise.

Thus, the knowledge engineer, or knowledge engineering team and tools, has access to multiple sources of data through various channels and uses these to develop a variety of conceptual models. Figure 2 shows the major conceptual models that may be developed in knowledge engineering, distinguished by their sources, and indicating some of the knowledge engineering processes and skills involved. This figure attempts to be comprehensive, showing knowledge sources not only in association with the expert and his or her behavior, but also knowl-

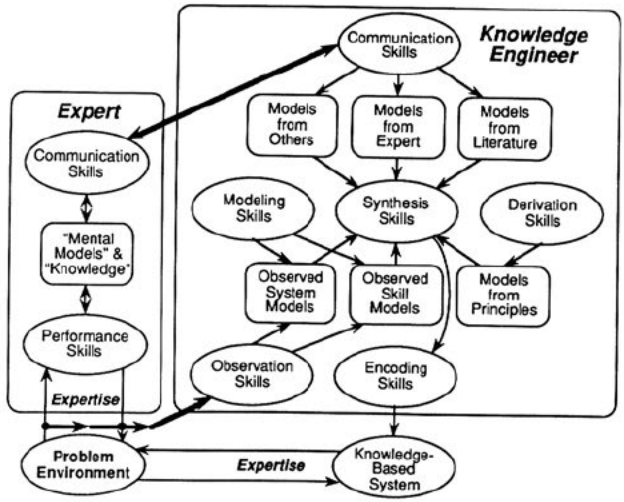

Figure 2. Modeling processes in knowledge engineering.

edge derived from others, the literature, and through the application of laws and principles.

Many issues in knowledge engineering are already apparent in Fig. 2. There may be conflicts between the models. If the models are each developed in depth and then conflicts are resolved this is only a system design problem. However, if the availability of one model drives the elicitation of another, then inadequate models may result—for example, if the knowledge engineer uses background knowledge to interpret expert communication that is inconsistent with the expert's conceptual framework. The variety of skills demanded of the knowledge engineer seem unlikely to exist in all but a few people and suggest the need for a team approach, training programs, and support tools.

As emphasized already, another significant aspect of the knowledge engineering task that is apparent in Fig. 2 is that the final model developed may bear little resemblance to the mental models assumed to exist within the expert. There is no reason to require that such models should be independent of the knowledge engineering process, or even exist before it is commenced. In a definition of knowledge engineering from the early days of expert system developments, Hayes-Roth, Waterman, and Lenat[14] note that "One of the most difficult aspects of the knowledge engineer's task is helping the expert to structure the domain knowledge, to identify and formalize the domain concepts." In terms of system development it does not matter how the final conceptual model for skill emulation is constructed provided it is effective. Methodologies and tools for knowledge acquisition may have legitimate concerns about biasing the expert, but they may also have legitimate objectives of building mental models where none previously existed and strengthening or superseding those that do exist.

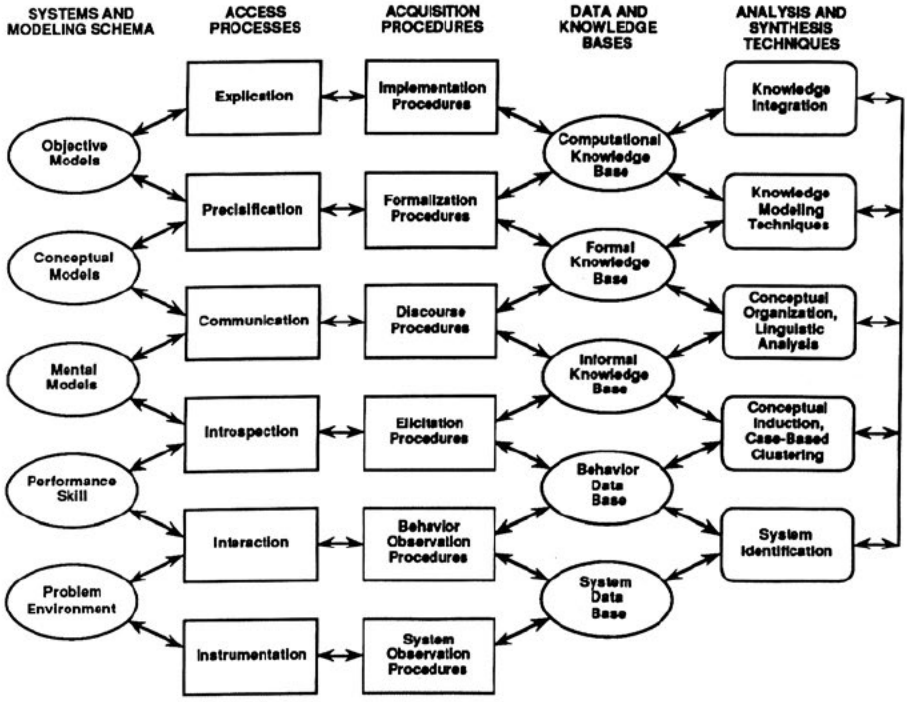

Figure 3. A hierarchical framework for knowledge acquisition.

IV. A MODELING ARCHITECTURE FOR KNOWLEDGE ACQUISITION

The discussion of the preceding sections and the range of modeling processes shown in Fig. 2 provide a framework for knowledge engineering in terms of the sources and types of models involved. Some of the flows and structures involved are apparent in Fig. 2, and it is interesting to see if these can be organized into a well-defined "architecture" for knowledge engineering. However, it should be noted that there is a danger in so doing that too much structure will be imposed, excluding some phenomena in actual knowledge engineering practice, and also imposing a normative framework apparently constituting "good" knowledge engineering. This section develops a hierarchical architecture for knowledge engineering methodologies and tools that can provide a basis for system organization, but it is not intended to claim that this basis is "correct," "unique," or "best," only that it is a useful framework for a very complex situation.

Figure 3 presents a modeling framework for knowledge acquisition methodologies, techniques, and tools based on the distinctions already discussed and the incorporation of system analysis and knowledge engineering procedures. In the leftmost column are the knowledge sources in terms of systems and modeling

schema already discussed with the addition, at the top, of "objective models" as a term for the formally specified operational models that characterize positive science. In the column to the right of this are the processes giving access to these models. These processes are shown as mediating between the systems and models involved, deriving from the generating, the hierarchical relation between the systems and models in the leftmost column.

In the next column on the right are shown the knowledge acquisition procedures appropriate to each of the access processes. These generate data and knowledge bases as shown to their right, which are in one-to-one correspondence with the original systems and models in the leftmost column. In the rightmost column are shown analysis techniques that draw on these databases to generate the computational knowledge base, and also mediate between them generating one form of data or knowledge from another. These combine with synthesis techniques that integrate the results of analysis and of derivations from various knowledge sources to synthesize a computational knowledge base.

Thus the overall schema consists of five types of component:

(1) *Systems and modeling schema:* the problem environment, performance skill to be emulated, expert's mental models, knowledge engineer's conceptual models, and, possibly, objective models.
(2) *Access processes:* instrumentation of the target system, the expert's interaction with it, his or her introspection about the skill, communication about it, and in Carnap's[15] terminology its "precisification" in formal terms, possibly resulting in the kind of "explication" accepted as the basis for explanation and justification through objective knowledge.
(3) *Knowledge acquisition procedures:* observation of the target system, observation of the expert's behavior, elicitation procedures, discourse procedures, formalization procedures, and implementation procedures.
(4) *Data and knowledge bases:* database of system data; database of behavioral data; informal knowledge base; formal knowledge base; computational knowledge base; objective models.
(5) *Analysis and synthesis procedures:* classical system identification can be used to build system models from observation data; empirical induction and case-based clustering can be used to build skill models from behavioral data; conceptual organization and linguistic analysis techniques can be used to build a formal, or structured, knowledge base from an informal, or intermediate, one; knowledge modeling techniques can be used to represent the formal knowledge base in computational form; and logical deduction from laws and principles may be used to provide some knowledge about a system and this, together with the results of data analyses from various sources needs to be integrated to form a computational knowledge base.

All the earlier stages of analysis are shown as normally creating data at the next level but also as potentially creating computational systems in their own right. For example, in the development of autopilots it is possible to go directly from system identification of aircraft flight data through optimization to a working autopilot, but it is more realistic to note that the flight data is taken when the system is under pilot control and hence behavioral data is also involved, and that the results will be subject to subjective, engineering, and scientific validation before being incorporated in a working design.

The architecture of Fig. 3 makes no presuppositions relating to the nature of and existence of mental models, their relations to performance skills, their accessibility through language, or the status of computer emulations of these skills as cognitive models. It provides the distinctions necessary for these issues to be discussed but takes no sides. Clearly, practical knowledge acquisition procedures and analysis techniques do involve strong presuppositions, and by placing methods, techniques, and tools within this framework it is possible to clarify the assumptions made, and also identify the available and missing functionality.

It is also worth noting that Fig. 3 illustrates the way in which knowledge engineering as a system design methodology is sandwiched between two classical approaches to system engineering. At the bottom of the figure is the path to system design through instrumentation, data collection, and system identification. At the top of the figure is the path to system design through existing objective knowledge of the physical world allowing explication of particular requirements to lead directly to implementation. The middle layers represent the enrichment of the design process when we draw on human skills as exemplars of the system to be designed. Such a process has been common informally in engineering design, and knowledge engineering may be seen as formalizing it now that computer technology makes it feasible to develop knowledge-based systems operationalizing human expertise.

V. KNOWLEDGE ACQUISITION ISSUES IN TERMS OF THE FRAMEWORK

We have previously analyzed a wide range of knowledge engineering methodologies, techniques, and tools at a level of detail which is not possible here,[2] and there is now a wide range of books describing them[16,17] and knowledge engineering in general.[18] This section focuses on some of the major classes of knowledge engineering methods now in use, and discusses them within the framework developed.

First, it is now obvious from Figs. 2 and 3 why the knowledge acquisition literature involves such a "patchwork" of approaches. It is difficult to envision any modeling methodology from any discipline that might not have a role to play within the wide range of processes shown in these figures. The requirements range from the modeling of physical systems, through psychological systems, to formal systems, that is, through all aspects of Popper's[19] characterization of knowledge processes in terms of "3 worlds".[20] We should expect the diversity, incoherence, and gaps that constitute the patchwork, but should also be able to organize the pieces that form the patchwork within the levels and columns shown in Fig. 3. The relations between components, their interactions and mutual support, and the missing components necessary to provide integration should then be apparent.

Second, it is clear that a catchall term such as "interviewing" does not designate a monolithic technique in terms of the framework of Fig. 2. When we interview an expert we may be operating at any level of the hierarchy and may

be supporting any one of the many processes shown. All that we can say about interviewing in general is that a flow of linguistic information is involved—it is the content of that flow that determines the type of knowledge engineering involved. The expert may provide observations of the system, observations of his or her own problem solving behavior, introspection about aspects of his or her mental models, statements about his or her conceptual models of any aspect of the situation, and statements of formal or even computational models relating to the situation.

In particular, at the initial stages of knowledge engineering, interviews with experts may be used to characterize the sources and types of models involved encompassing many aspects of Fig. 3, that is, in an initial project management mode. This leads to interviews being characterized as unstructured or structured, specific or general, probing or expansive, and so on. While such characterizations are useful in discussing the styles and structures of interviews involved, they do not in themselves constitute a logic of interviewing. This arises from the needs to establish the overall form of the system as shown in Fig. 3 in a specific situation, and then to fill in the detail from available and effective sources.

Third, specific knowledge acquisition techniques are characterized by their vertical and horizontal locations within the framework. For example, protocol analysis involves data collection for the behavior database through observation of interaction at one level or elicitation of introspection at the next. The behavior database is then subject to statistical system identification or to conceptual induction and clustering. The data collection methodology in protocol analysis may easily slip into the elicitation of not just a protocol but also an explanatory commentary which belongs in the informal knowledge base and is subject to linguistic analysis. Thus, applications of protocol analysis may involve multiple levels and activities that are confusing unless seen as organized within the framework.

Fourth, analytical tools such as induction and clustering algorithms have a well-defined location in the framework as analysis techniques providing a model creation technology. Their differentiation comes from what level, or levels, they can accept data, and at what level, or levels, they create data. A major focus in machine learning research for several years has been to create models at the knowledge level, conceptual structures rather than rules.[21] To the extent that all the analytic techniques involved do this, the problem becomes one of integration of conceptual structures. However, it is more usual to find that the analytic tools create data or knowledge at different levels and further processing is required before integration is possible.

Fifth, methodologies such as KADS[22] that provide a structured software engineering approach to knowledge engineering are focused at the penultimate level of applying formalization procedures to derive a formal knowledge base through the precisification of conceptual models. The KADS "levels" correspond to the different types and sources of conceptual models, classified in terms of the:

- *domain level* corresponding to the problem environment models in Fig. 2;
- *inference level* corresponding to a computational decomposition of skills into canonical operations, *role-limiting methods*[23] or *generic tasks*[24];
- *task level* corresponding to the skill model in Fig. 2;
- *strategic level* corresponding to meta-problem-solving processes of choosing and changing a problem-solving approach.

Thus KADS focuses on the detailed structure of a formal problem-solving architecture within which to operationalize the results of knowledge acquisition rather than on the processes of knowledge acquisition themselves. It may be seen as providing a formally specified "virtual machine" well-suited to the range of system developments that have come to be classified under the heading of "knowledge-based systems." Less formally, one can say that it provides a "high-level language" in contrast to the "machine languages" provided by expert system shells.

Sixth, methodologies such as those stemming from personal construct psychology[25] that are based on a cognitive model of intelligent agents are focused on the middle levels in Fig. 3, modeling the way in which mental models mediate between conceptual models and performance skills. Clearly any well-founded cognitive psychology has a potential role to play in knowledge acquisition that is strictly within the "expert systems" paradigm of modeling the expert rather than the system. However, to be useful the psychology must result in operational models on the one hand and support methodologies giving access to its hidden variables on the other. Personal construct psychology has been particularly attractive in these respects because, even though it is a constructivist model, it takes a positivist, axiomatic approach based on a few well-defined primitives that correspond to a formal intensional logic.[26] Extended "repertory grid" methodologies, eliciting hierarchical relations between concepts as well as definitional properties, provide a methodology to elicit these primitives.[27]

Seventh, the interface between cognition and formalization for people is mediated through language and knowledge acquisition support is required for the communication and discourse procedures and analysis level in Fig. 3. Ideally this should be based on a comprehensive theory of language arising out of the cognitive psychology, and this is beginning to appear feasible but still some way from realization.[28] Current knowledge acquisition tools addressing this level range from those focusing on the intertranslation of restricted natural language and knowledge representation frames such as SNOWY,[29] to those providing support for human classification of natural language components in terms of knowledge level primitives such as Cognosys.[30] Improved natural language processing must have a very high priority in the support of the complete range of knowledge acquisition processes in the framework of Fig. 3.

Eighth, classical system analysis focuses on the collection and analysis of system and behavior data at the lower levels of Fig. 3. In complex system development the other levels play their part, but the basic assumption has been that the final system design is grounded in accurate models of the environment in which the system is to operate and in precise "requirements specifications"

corresponding to the top level goals of the human agents involved. The implementation is quite separate from the system analysis and design because conventional programming languages do not provide knowledge-level constructs supporting human understanding of their operation. In this respect, the framework of Fig. 3 may be seen as an extension to classical system analysis appropriate to knowledge-based systems where very high level languages at the "knowledge level" are being used for the implementation to provide this support of human understanding.

Ninth, points five through eight may be brought together to suggest that a complete account of knowledge acquisition for knowledge-based systems requires the integration of classical system analysis, cognitive modeling of intelligent agents, linguistic analysis of text and discourse, and a rich formal language at the knowledge level. This integration would provide us with an extended system development methodology adequate to cope with the increased expectations of those specifying requirements for knowledge-based systems. In particular, the rich formal language at the knowledge level is necessary to overcome the problem reported in much knowledge acquisition research to date that the mediating representations and intermediate knowledge forms developed in acquisition have to be excessively distorted to operate in existing knowledge-based system shells.

However, note that the knowledge level language alone is only a target for specification. On the one hand it needs to be made operational as computational knowledge. On the other it needs to maintain an effective ongoing relation with the knowledge processes that drive it, many of which are those of active human agents forming an essential component of the ongoing system operation. Knowledge acquisition should not be seen as part of the system design process only. Knowledge is dynamic and changing, and acquisition, maintenance, and upgrading must merge into one process that is fully supported as an ongoing system operation. In particular, the cognitive aspects of much of the knowledge must continue to be recognized and supported in the ongoing system operation. Formalization cannot be at the expense of human understanding. On the contrary, effective formalization should lead to enhanced human understanding. This is the greatest challenge in the development of an effective knowledge-based systems technology. The objective is not just emulation of isolated human peak performance, but rather the emulation of the total human ability to develop, adapt, and maintain that performance in a dynamic and uncertain environment. In general, this is a social matter also, and most knowledge-based systems will require the effective integration of human and machine capabilities.

VI. CONCLUSIONS

Research and practice in knowledge acquisition has been based on a "patchwork" of methods and techniques, imported from other disciplines and adapted pragmatically to accommodate knowledge acquisition methodologies and tools. In this article we have tried to subsume and extend previous attempts to provide a framework for knowledge acquisition by developing a range of perspectives

on the issues involved and the different approaches possible. Much of the early work on knowledge acquisition was based on the presupposition that human expert performance is predicated on the existence of "mental models" of domains and problem-solving techniques, and that these can be elicited and transferred to an expert system. This article has focused instead on the knowledge engineer's modeling process, his or her conceptual models of systems associated with the expert's skill, and their sources and types. A framework was developed of these models and modeling processes within a hierarchical structure. This has been used as a framework for the discussion and characterization of various knowledge acquisition issues, methodologies, tools, and systems.

Knowledge is dynamic and changing; and acquisition, maintenance, and upgrading must merge into one process that is fully supported as an ongoing system operation. In particular, the cognitive aspects of much of the knowledge must continue to be recognized and supported in the ongoing system operation. The greatest challenge in the development of knowledge-based systems is that effective formalization should lead to enhanced human understanding. The objective is the emulation of the total human ability to develop, adapt, and maintain skilled performance in a dynamic and uncertain environment. This article has shown that a modeling perspective provides a framework for knowledge acquisition methodologies and tools supporting the effective integration of knowledge from many sources both human and machine.

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada. We are particularly grateful to Bill Clancey and other colleagues at the Knowledge Acquisition Workshops for discussions and critical comments that have improved the framework presented in this article.

References

1. J.H. Boose, "A survey of knowledge acquisition techniques and tools," *Knowl. Acquisition* **1,** 39–58 (1989).
2. M.L.G. Shaw and B. Woodward, "Mental models in the knowledge acquisition process," *Knowl. Acquisition* **2,** 179–206 (1990).
3. P. Feyerabend (Ed.) *Against Method* (NLB, London, 1975).
4. W.J. Clancey, "Viewing knowledge bases as qualitative models," *IEEE Expert* **4,** 9–23 (1989).
5. D. Gentner and A. Stevens (ed.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).
6. J.H. Boose, *Expertise Transfer for Expert Systems* (Elsevier, Amsterdam, 1986).
7. W.J. Clancey and J. Roschelle, "Situated cognition: How representations are created and given meaning," *Educat. Psychologist*, to appear (1991).
8. Z.W. Pylyshyn, *Computation and Cognition* (MIT Press, Cambridge, MA, 1984).
9. H.L. Dreyfus and S.E. Dreyfus, *Mind over Machine: The Power of Human Intuition and Expertise in the Era of the Computer* (Free Press, New York, 1986).
10. P. Compton and R. Jansen, "A philosophical basis for knowledge acquisition," *Knowl. Acquisition* **2,** 241–258 (1990).
11. D.A. Norman, "Some observations on mental models," in *Mental Models*, D.D. Gentner and A. Stevens (Eds.) (Erlbaum, Hillsdale, NJ, 1983).
12. B.R. Gaines and M.L.G. Shaw, "From fuzzy sets to expert systems," *Fuzzy Sets Syst.* **36,** 5–16 (1985).
13. R.S. Michalski and R.L. Chilausky, "Knowledge acquisition by encoding expert

rules versus computer induction from examples—A case study involving soyabean pathology," *Int. J. Man-Mach. Stud.* **12,** 63–87 (1980).
14. F. Hayes-Roth, D.A. Waterman, and D.B. Lenat (eds.), *Building Expert Systems* (Addison-Wesley, Reading, MA, 1983).
15. R. Carnap, *Logical Foundations of Probability* (Routledge & Kegan-Paul, London, 1950).
16. J.H. Boose and B.R. Gaines, *Knowledge Acquisition Tools for Expert Systems* (Academic Press, London, 1988).
17. B.R. Gaines and J.H. Boose, *Knowledge Acquisition for Knowledge-based Systems* (Academic Press, London, 1988).
18. H. Adeli, *Knowledge Engineering: Vol. 1, Fundamentals* (McGraw-Hill, New York, 1990).
19. K.R. Popper, "Epistemology without a knowing subject," in *Logic, Methodology and Philosophy of Science III*, B. Van Rootselaar (Ed.) (North-Holland, Amsterdam, 1968), pp. 333–373.
20. B.R. Gaines, "Social and cognitive processes in knowledge acquisition," *Knowl. Acquisition* **1,** 251–280 (1989).
21. S.J. Hanson, "Conceptual clustering and categorization: Bridging the gap between induction and causal models," in *Machine Learning An Artificial Intelligence Approach, Volume III*, Y. Michalski and R. Kodratoff (Eds.) (Morgan Kauffman, San Mateo, CA, 1990), pp. 235–268.
22. H. Akkermans, F. van Harmelen, G. Shreiber, and B. Wielinga, "A formulisation of knowledge-level models for knowledge acquisition," *Int. J. Intell. Syst.* **8,** 169–207 (1993).
23. J. McDermott, "Preliminary steps toward a taxonomy of problem solving methods," in *Automating Knowledge Acquisition for Expert Systems*, S. Marcus (Ed.), (Kluwer, Boston, 1988), pp. 225–256.
24. B. Chandrasekaran, "Generic tasks as building blocks for knowledge-based systems: The diagnosis and routine design examples," *The Knowl. Eng. Rev.* **3,** 183–211 (1988).
25. G.A. Kelly, "Aldous: The personable computer," in *Computer Simulation of Personality*, S.S. Tomkins and S. Messick (Eds.) (Wiley, New York, 1963), pp. 221–229.
26. B.R. Gaines and M.L.G. Shaw, "Cognitive and logical foundations of knowledge acquisition," in *Proceedings of the Fifth AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop*, J.H. Boose and B.R. Gaines (University of Calgary, Banff, Canada, 1990), 9-1-9-24.
27. J.H. Boose and J.M. Bradshaw, "Expertise transfer and complex problems: Using AQUINAS as a knowledge acquisition workbench for knowledge-based systems," *Int. J. Man-Mach. Stud.* **26,** 3–28 (1987).
28. R.W. Langacker, *Foundations of Cognitive Grammar: Theoretical Prerequisites, Volume 1* (Stanford University Press, Stanford, CA, 1987).
29. F. Gomez and C. Segami, "Knowledge acquisition from natural language for expert systems based on classification problem-solving methods," *Knowl. Acquisition* **2,** 107–128 (1990).
30. B. Woodward, "Knowledge engineering at the front-end: Defining the domain," *Knowl. Acquisition* **2,** 73–94 (1990).