

## Industrial Strength Knowledge Bases: Issues and Experiences

Thomas P. Kehler, IntelliGenetics, Inc., Peter Friedland, Stanford University

Harry Pople, Univ. of Pittsburgh, Rene' Reboh, SRI International

Steve Rosenberg, Hewlett Packard Labs

Recently we have entered an era of intense government and industrial interest in AI systems. In the development of robust AI systems, the role of the knowledge base is extremely important. A significant investment in knowledge base development will be made over the next few years. Consequently the adequacy of current techniques for the development and maintenance of large knowledge bases must be given critical review. Like database development in the early sixties, knowledge base development issues are likely to dominate AI system development. These issues are likely to drive the development of practical techniques as well as the refinement theoretical approaches.

### Knowledge Base Stability

Development of a large AI system implies long term commitment to the knowledge base. The cost of knowledge base development represents a significant fraction of the total cost of implementing an expert system. Because such a significant investment is made in the knowledge base, stability over time is an important issue. One question to consider is the susceptibility of knowledge bases to innovations in AI technology.

The experience of expert system developers has been that successful applications have resulted from limiting the problem scope and using primarily experiential or judgmental knowledge. Limiting the scope of the system implies not only a domain limitation but also assumptions on how the system is going to be used. Knowledge bases intended for expert decision making may differ from those intended for tutorial use. An issue is whether changing the intended use of a system will adversely impact stability of the knowledge base.

Competing with the notion of knowledge base stability is the practical constraint that AI systems are developed by a process of evolutionary refinement. The two constraints of knowledge base stability and development by evolutionary refinement imply tension between the degree of commitment to specific techniques for representation and reasoning in knowledge based systems. These constraints in turn raise questions about the architecture of knowledge base development tools: Is the knowledge base development tool to be viewed as a "utility package" or as an "integrated applications system?" This particular issue has influenced design considerations in various systems.

### Memory: Uniform or Structured

A second issue which must be considered is that of memory management for knowledge bases. If the virtual memory address space offered by the hardware is smaller than that needed for the knowledge base, a structuring of the knowledge representation is needed. Only a few years ago, the limitations of timesharing virtual memories presented serious limitation to the development of robust knowledge bases. With the advent of personal Lisp machines, this limitation has been removed... for now.

Industrial strength knowledge bases are likely to grow large quickly. The need for extensive data in some applications intensifies the rate of knowledge base growth. Do we leave the size limit question to hardware architecture or should we structure the memory in some way? Clearly there are practical systems which require more than a 32 bit address space for a memory map. The virtual memory issue has largely been overlooked, with the hope that our knowledge bases will not exceed the address space of the machine.

### Integrity

From an engineering perspective, knowledge base integrity implies that interactions with the knowledge system will be consistent over the scope of the application. Inconsistency in the knowledge base becomes evident when a user enters territory not previously explored by the designer of the system. Scientifically, considerable attention is being given to establish knowledge base integrity in terms of a well-formed epistemology.

Most systems used today in industrial applications are built using an engineering approach. Explanation facilities are often used to debug knowledge bases. Knowledge base debugging sessions are driven by performance specifications for the system. Integrity of a system is reached by a step-wise process of testing and refinement. Truth maintenance techniques developed which provide computational support for lines of reasoning. These techniques allow the system to trace the dependency of one fact on other facts. Questions to be considered are: How can knowledge system building tools aid in achieving a consistent, robust knowledge base?

Current systems which use incremental development as a means of extending system capabilities suffer from a degree of fragility. When a user proposes an interaction not previously anticipated the system degrades in ungraceful ways. For a system to degrade gracefully the system must have a model of the limits of its expertise. In the past the semantics of a knowledge representation has been an issue even for small systems. Clearly developing domain semantics which can identify their own limits will be a major issue for industrial applications of knowledge based systems.

#### Knowledge Acquisition and the User Interface

In order to extend the capabilities of a knowledge system, current technology requires new knowledge to be entered and debugged by the knowledge engineer or domain expert. This process puts a particular stress on the user interface. The bit mapped graphics display integrated with LISP programming environments has made it possible to construct highly interactive interfaces to knowledge bases. Using graphics it is possible to represent knowledge in a way which is natural to the user, offering an extended vocabulary for representation. Commonly used graphical representations of concepts may be directly represented. Since the graphical representations are integrated with the Lisp environment, it is possible to make the representation active, items may be selected and manipulated using graphical pointing devices such as the mouse.

The use of these techniques can potentially improve knowledge base utility in a system. Knowledge base building tools can use this potential by providing integration of graphical perspectives on knowledge with knowledge representation techniques.

#### Joint Knowledge Base Development

Most prototype knowledge systems have been developed based on the knowledge of a single expert. An alternative to the single expert has been establishing a consensus among several experts. Broad use of commercial AI systems requires joint development knowledge bases by many experts. Knowledge base building tools may aid in the process of handling alternative sources of knowledge and recognizing conflicts in knowledge base refinement by several contributors.

While personal workstations have been a great aid in providing a large virtual address space and an improved user interface, they have introduced the complication of distributed knowledge bases. Working with distributed knowledge bases is of particular importance in the construction of expert systems where collaboration between experts and disciplines is an integral part of system use.

#### Interfaces to Databases

Expert problem solving often requires utilization of data organized in large data management systems. For example, a molecular geneticist planning an experiment will make reference to the NIH database on DNA sequence data. Many prototype systems have avoided the data management issue by limiting the scope of application. Interfacing to data management facilities is an integral part of many industrial AI systems. Consequently, a knowledge system is likely to require interaction with a database system. Facilities integrating the knowledge based system with conventional database management systems are a likely knowledge base building tool.