The Knowledge Engineering Review

Trends in the marketing of expert system technology

LELAND TESCHLER
1533 Arthur Avenue, Lakewood, Ohio 44107, USA

As recently as a year ago, trade journal articles that discussed expert systems often carried a great deal of hyperbole and inflated claims. Many otherwise sophisticated technologists put expert-system techniques in the same category with such far-off ideas as orbiting missile defense systems and thinking machines.

This situation has changed dramatically. As expert system technology has been increasingly comercialized, the expectations of potential users are quickly coming down to earth. Companies that are typically among the early adopters of new technologies now look at expert system projects as capital expenditues that must be cost-justified in conventional ways. Case histories of successful expert system projects are much easier to come by today than even a few months ago. Reports of how expert programs have benefitted early adopters, coupled with the availability of low-cost expert shells, are helping the technology to diffuse throughout industry.

In all, the image of expert systems is changing from a "gee whiz" technology to that of a useful tool, true to the predictions of practitioners in the field. However, the product strategies that analysts have forecast for expert systems are not unfolding quite as expected.

A number of observers have anticipated that a huge market will eventually emerge for pre-packaged, domain-specific expert programs. Marketing research firms have suggested that such systems will ultimately be found handling chores in fields ranging from investment banking to manufacturing. What analysts have by and large not considered, however, is the character of a firm that would be able to bring such products to market. What kind of company would be able to market expert systems at, say, high level finance? The answer to this question could determine the dynamics of the expert system market in the next decade.

On this point, the picture is starting to clear. It now looks as though firms with tremendous resources in both expert system technology and in the underlying domain of interest will be the most likely purveyors of pre-packaged expert systems.

Texas Instruments Inc. (TI) presents a good example. The company has developed a number of relatively large expert systems for use in house, and is now trying to decide whether to release some of these packages commercially. The systems address applications ranging from the justification of capital expenditures to the diagnosing of faults in installations of IBM mainframes.

A key point is the character of the effort that led to these systems. TI computer laboratories have been working on artificial intelligence technology since 1978. The company is a major user of expert systems, as well as a substantial vendor of expert system shells.

The expert systems that TI is now considering as potential products are generally products of appreciable development efforts. These programs comprise typically between several hundred and a thousand rules. Most have continued to evolve as they have been used. Some have been second efforts, total rewrites of first attempts that were either too slow or had other terminal defects.

Would-be expert-system marketers should ask themselves whether TI's resources are a prototype of what it takes to successfully develop and sell domain-specific expert systems. (TI marketers say that the systems they are considering bringing to market would be 80% solutions—some 80% of the domain-specific knowledge would be in the package, and the buyer would supply the other 20% to customize the software to specific end uses.)

Two main assets give TI the option of selling complete systems—an experienced cadre of artificial intelligence software professionals, and an equally competent contingent of domain experts who are prone to adapt and work with new technologies. What kind of companies can draw on these two ingredients? For now, the answer seems to be Fortune 500 companies that are deeply involved in software technology.

One conclusion that comes from this analysis is that the small entrepreneurial start-up company may not play a large role in bringing complete domain-specific systems to market. It would be difficult for such companies to maintain the breadth of knowledge necessary for both software development and in-depth domain expertise. It is likely that the basic domain-knowledge typified large expert systems goes beyond what a handful of human experts at a typical start-up can provide.

Moreover, it is becoming clear that even small expert programs require a significant amount of "breaking in" use by experts and novices in real situations, before they can serve as useful tools. This type of de-bugging may be hard for small companies to provide, even through co-operation with benevolent large companies.

The resource requirements for developing and commercializing full-blown expert packages loom large enough that we may not see many such programs any time soon. Developers are coming to appreciate the effort required to deliver useable and relatively bug-free software. A safe approach is to allow expert software to pay for itself first as an in-house project before commercialization. This may be the preferred method for wringing out the bulk of domain-specific experts commercialized in the next few years.

Though domain-specific programs that are more than 80% solutions seem to be at least a few years away, domain-specific expert shells are already at hand. The first such shells are designed for scanning sensors and controlling real-time processes such as electric utilities and distillation towers. At least two shells, G2 and Personal Consultant OnLine, have been released for such uses.

The key traits that allow the programs to handle monitoring is the ability to consider the history of events or variables when firing rules, coupled with event-scheduling mechanisms normally associated with a real-time operating system.

Within the next year, shells are also expected to be optimized for uses in diagnostic and scheduling applications. These two areas are probably considered the most straightforward to develop today. The reason for optimizing shells for such applications, of course, is to reduce the amount of knowledge developers need to successfully construct expert systems.

Other advances are coming in the handling of human interfaces to expert systems. In particular, developers are coming to understand the need for on-going expert system maintenance. They have discovered that, not surprisingly, users lose confidence in an expert system if the program continually gives the wrong answer for specific kinds of queries. In simple terms, this means that any expert program must be changed as quickly as possible when bugs in its logic emerge. The addition of mechanisms that facilitate such changes are receiving a high priority among shell developers.

Nevertheless, despite improvements and refinements in shell technology, indications are that the development of a medium-to-large expert system will demand a considerable amount of expertise on the part of developers. Techniques chosen for knowledge representation will be an important component in the success of projects for a long time to come. In addition, there are several issues that may prevent the technology from diffusing quickly beyond applications in simple diagnosis and scheduling. A few of these issues are topics of current research that many think will fall to analysis only after a great deal of further study.

For example, many kinds of configuration problems fit into the category loosely defined by Mark Fox, a noted AI researcher at Carnegie-Mellon University, as demanding second generation expert systems. Second generation systems, to simplify somewhat, are those where the program obtains answers as good or better than those of human experts only if the rules are considered in a certain special order that depends on the problem at hand. Second generation systems are thus typified by clever rule selection techniques that are basically empirical.

One problem is that it often takes experienced developers to find the sorts of clever rule selection techniques associated with second generation systems. Unfortunately, the number of problems that

demand second generation techniques could be large. A third problem is that because of the empirical nature of the rule selection mechanisms in such programs, it may be hard to build them into expert shells in forms that would be palatable to relatively inexperienced shell users.

A second category of problem that is currently difficult to attack is typified by a need for both deep and shallow knowledge. In general, low-cost shells stop short of providing an obvious means of integrating shallow knowledge extracted from human experts with deep knowledge about inherent processes taken from sources such as physics, chemistry, or economics text books.

The typical use for deep knowledge is as a last resort, to provide a possible solution path when experience fails to yield satisfactory results. The problem is that the links between deep and shallow knowledge are often complicated. As in the case of second generation systems, the effective use of deep knowledge requires appreciable skill on the part of developers.

In light of these intellectual hurdles, it seems safe to say that the development of medium to large scale expert systems will remain in the realm of the software specialist for at least the next few years. Serious shell users will continue to need much expertise on knowledge representation techniques and similar issues. If nothing else, this should provide a growth market for expert system newsletters and textbook publishers.

Vendors say that within the next two years, shell technology will advance to a point where knowledge representation may cease to be a concern of the shell user. If this prediction comes true, it will mean a quantum jump in simplification for the average expert system developer. But consider this: it has been over five years since the personal computer first became commercialized on a large scale. Even now, a PC is not a device that can be lightly entrusted to completely untrained users, unless the said users happen to be degreed engineers.

It is not unthinkable for the expert system tool market to follow a similar pattern. In expert systems, we may indeed be at the bottom end of an exponential growth curve. However, if expert system technology follows the road laid out by PCs, the next few years will still see a growth market for human experts on expert systems.