N90-27326

INDUCTIVE KNOWLEDGE ACQUISITION EXPERIENCE WITH COMMERCIAL TOOLS FOR SPACE SHUTTLE MAIN ENGINE TESTING

Dr. Kenneth L. Modesitt
Head, Department of Computer Science
Western Kentucky University
Bowling Green, KY 42101

Abstract: Since 1984, an effort has been underway at Rocketdyne, manufacturer of the Space Shuttle Main Engine (SSME), to automate much of the analysis procedure conducted after engine test firings. Previously published articles at national and international conferences have contained the context of and justification for this effort (Refs. 3, 7, 10, 11, 15, 16). Here, progress is reported in building the full system, including the extensions of integrating large databases with the system, known as "Scotty." Inductive knowledge acquisition has proven itself to be a key factor in the success of Scotty. The combination of a powerful inductive expert system building tool (ExTran), a relational data base management system (Reliance), and software engineering principles and Computer-Assisted software Engineering (CASE) tools makes for a practical, useful and state-of-the-art application of an expert system.

INTRODUCTION

Every time a Space Shuttle Main Engine (SSME) is test fired, hundreds of measurements are taken directly from a wide variety of sensors. Many more values are also calculated from these. All of these data values, when combined with previous engine and component performance, are used by the engineering staff at Rocketdyne, the propulsion division of Rockwell International, to determine the future tests. These outcomes can vary from all requirements being met, to a few minor events, to a rare significant event. As the SSME is the world's most complex reusable liquid-fuel (oxygen and hydrogen) rocket engine, Rocketdyne and NASA, the customer, conduct thorough investigations of each test firing by their most highly-trained engineering staff. The author is a former employee of the Rocketdyne division.

To continue its virtually perfect record of supporting shuttle flights, Rocketdyne is always looking for ways, both technical and organizational, to improve the quality of the product while working within customer guidelines. One of the major methods involves making the most accurate diagnosis, analysis, and recommendation possible for the the next engine test or shuttle flight. To perform this task, reliance has been on maximal use of sophisticated tools and the expertise of an engineering staff. This staff has accumulated experience dating back to 1975 and covering 1400+ SSME firings, plus numerous other ones: Apollo F-1, J-2, and Atlas engines.

Rocketdyne was confronted with a significant dilemma: how to improve the quality of the SSME test analysis in the face of diminishing senior staff. Several options to solve this dilemma were discussed in Ref. 7. It was decided to use a combination of staff, results from previous SSME tests, and automated software tools to build a prototype for automated corporate expertise related to reusable propulsion components.

Rocketdyne was far from alone in being confronted with the above problems. Indeed, the corporation had ample "company" in deciding to use a type of automated tool known as expert systems, part of the artificial intelligence technology. The company is certainly not the first to decide to concentrate initially on a diagnosis type of application, a type currently of considerable importance to industry despite being "old-hat" to the AI research community. So what <u>is</u> unique about Scotty, the name given to the automated system? There are two unusual aspects.

One such aspect is the incorporation of Scotty as "another", albeit advanced, software tool which must:

1. Meet corporate-wide software engineering development and quality guidelines.
2. Live in a distributed corporate environment,
3. Talk to large data bases,
4. Be maintained by existing engineering staff,
5. Execute on standard computers,
6. Be amenable to parallel processing hardware, and
7. Run with color graphics terminals,

The other unusual aspect is a technical one which increases the ease with which Scotty can be constructed. By use of a type of Expert System Building Tool (ESBT) known as inductive or example-based, the historical expertise now reposing in data bases, both in human and machine form, from the hundreds of SSME tests can be transformed into examples, and thence automatically into rules. These rules will, in turn, drive Scotty during normal day-to-day operation in future years.

Scotty: HISTORY

In 1984, the author was hired by Rocketdyne to assist in the construction of an automated tool for SSME test analysis. The employment was on a half-time basis, and was in addition to his position as Professor of Computer Science at California State University, Northridge. Within two months, a proof-of-concept model for a High Pressure Oxidizer Turbo Pump (HPOTP) had been built. This involved recommendation of an inductive ESBT, Expert Ease by Intelligent Terminals, Ltd (ITL), now known as Knowledgelink, in Glasgow, Scotland, and the first such PC-based ESBT commercially available. The tool was purchased and used, after minimal training time, by a mechanical engineer, to diagnose HPOTP anomalies, by specifying 42 examples and nine attributes. A 48 rule subsystem was automatically generated by Expert Ease. <u>No</u> rules were required of the engineer. This prototype and the problem context, rationale, and solution were described in an early paper (Ref. 7). A desirable tentative system configuration is shown in Figure 1.

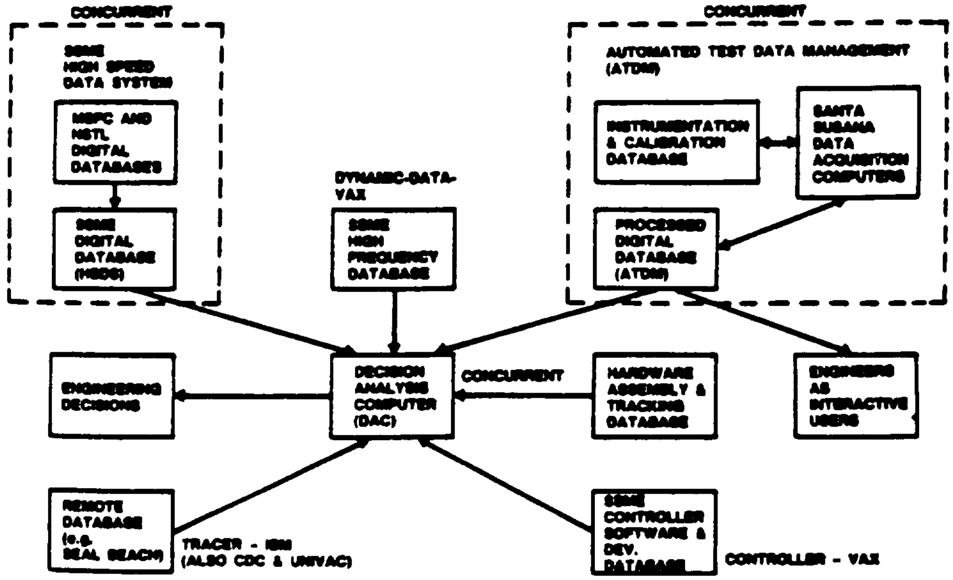

Figure 1. Scotty - Final System Configuration

During 1985 and 1986, the system (now named Scotty) underwent several extensions. From a tool viewpoint, a more powerful ESBT became available. ExTran 7, an industrial strength Fortran-based inductive ESBT from ITL which runs on a wide variety of machines from PCs to workstations to super-minis to mainframes, was recommended (Ref. 1). A process for using ExTran is given in Figure 2. ITL ported the product to the available Concurrent Computer Corporation 3260 super-mini at minimal cost. The HPOTP examples were immediately transported to ExTran and the resulting module was now a true, albeit simple, knowledge base system (KBS) utilizing "Why", "How", and "What if" type questions, history files, external interfaces, and all the other features usually associated with a KBS.

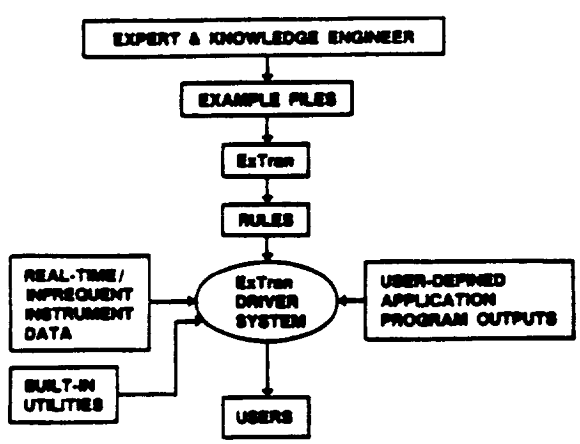

Figure 2. Inductive Expert System

Conceptually, Scotty was extended in several directions during this same time period. It was demonstrated that multiple problems could be run concurrently on the multiple processor Concurrent 3260. Graphics routines (PLOT-10 and GKS libraries) were tied to ExTran with a minimum interface. In-house statistical routines were easily linked to Scotty. Small Fortran routines were written to access SSME test files and output attribute values for input to Scotty sub-problems. Additional SSME component modules were specified. A major extension was the run-time interface between ExTran and the large data base managment system DMS/32 supplied by Concurrent, then known as Perkin-Elmer (Ref. 6). These are all described extensively in a paper presented in 1986 (Ref. 3).

Scotty: CURRENT STATUS

As of mid-1988, Scotty underwent field-testing on a sub-system basis, using the taxonomy of Waterman (Ref. 19). Parts of Scotty were run in parallel with previous modes of operation to help determine the validity of the system, and to update its knowledge base. Scotty consists of far more than "just" an expert system, as is clearly shown in figure 3, but rather is one component in a fairly extensive software system. This reflects the strong belief that viable expert systems are most likely to succeed in a hybrid and integrated environment, where they must communicate easily with other standard existing and future sub-systems. This had been stressed by the author since the initial conception, contrary to the host of stand-alone KBSs being proposed in the early mid-80's, thanks to his 25 years of software engineering experience.

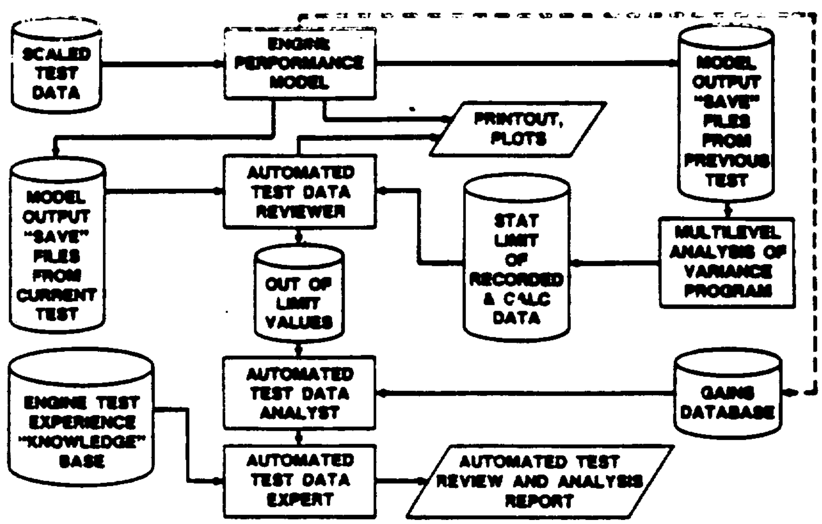

Figure 3. Context of Scotty - Automated Test Data Expert

Scotty, as of early 1988, consisted of 48 ExTran modules comprising 5400 lines of code (LOC) in Fortran. Supporting code required 7100 LOC. The ExTran generated code was derived automatically from approximately 1100 examples. Only 125 rules have involved any manual intervention to date. The other 1400 rules have been induced automatically.

KNOWLEDGE ACQUISITION ISSUES

The above numbers should be considered extremely carefully. Note that the knowledge acquisition task involves far more than simply eliciting examples from an expert or a data base. In fact, this component is relatively easy. The much more critical and difficult task revolves around the structuring of Scotty! Many in the AI field have become so enamored with the power of induction that they have forgotten some very basic software engineering principles. The top-down (divide-and-conquer) strategy has shown itself to be an extremely powerful one for thousands of years in the engineering field. Do not give it up just because a new powerful bottom-up technique is now possible!

The process of induction which turns an unordered set of examples (an operational specification of a task) into an ordered set of rules or code is a very powerful tool. This addition to existing computer-aided system engineering (CASE) tools would be welcome, and is probably on the horizon, based on recent press releases. However, the process is really only concerned with the generation of a software module. Most current research (Ref. 14) and the Scotty experience indicates that the majority of the expertise of an expert lies in her/his ability to structure the overall complex solution. Considerable work in the area of civil engineering at Wayne State University (Ref. 2) also substantiates this belief.

What good does it do (and what havoc can be wrought) to have one enormous module, derived from hundreds of examples with dozens of attributes? To be sure, the resulting rules probably execute with blazing speed and derive the "correct" answer. However, and this is a big caveat, who will be able to understand the resulting rule? Who would be willing to verify that the resulting rule set is accurate? When such a huge module is generated, experience to date shows that the expert finds the rules to be simply incomprehensible. What must the poor end user think? What has happened to the "transparency" of the underlying system, one of the most valuable additions of expert systems to the software field? Of what use is the much-touted explanation capability now? Why do some vendors promote that their tools can operate with thousands of examples and hundreds of attributes? ExTran, on the contrary, encourages the expert to break down her problem into sub-problems by issuing a warning whenever the length of a rule exceeds certain bounds. There are also various versions which differ in the maximum number of attributes per problem.

Is it too much to ask that practicing software engineers and expert system developers actually work together? It just "may" be that each has something to offer the other. It is so frustrating to this author, after being in both fields, and in both industry and academia since 1961, to see such miniscule amounts of two-way communication between these two groups of professionals. Only recently have there been any hopeful signs, in terms of joint conferences.

Scotty: EXTENSIONS IN PROGRESS

Development is continuing on a number of fronts for Scotty. Included are: beta-testing of a new product jointly developed by Knowledgelink and Concurrent, augmenting the potential sources of existing data which can provide hidden or latent knowledge, and effectively utilizing graphics.

The major extension underway is the intention to use Reliance Expert (Ref. 5), which is the result of a joint project between Knowledgelink and Concurrent with roots in the earlier work at Rocketdyne (Ref. 3). This product extends the interface between ExTran and a powerful data base system to include the knowledge acquisition component of the former, as well as the run-time interface discussed in Ref. 3 (Figures 4, 5, and 6). This product is currently undergoing beta testing at Rocketdyne.

Basically, Reliance Expert permits any data, when represented as records in a relational DBMS to serve as a source of knowledge (usually hidden or latent) for the knowledge acquisition phase (induction) of ExTran. One of the uses for this portion of Reliance Expert would be to serve as an "expert" for historical knowledge of Scotty, as it can now be transformed automatically into examples and then to rules. So, once again, the knowledge acquisition bottleneck becomes less and less of an issue, as it will be possible to go directly from records in a DBMS to production rules in an expert system. Moreover, it is even possible for the expert system component to modify the DBMS, should that be desirable.

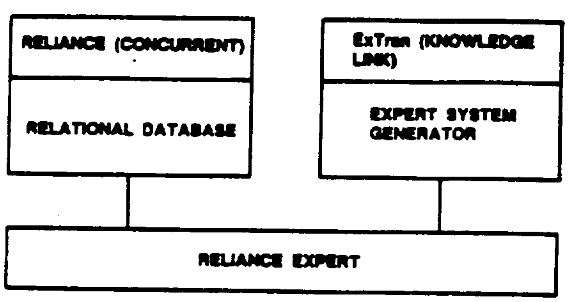

Figure 4. Reliance Expert Structure

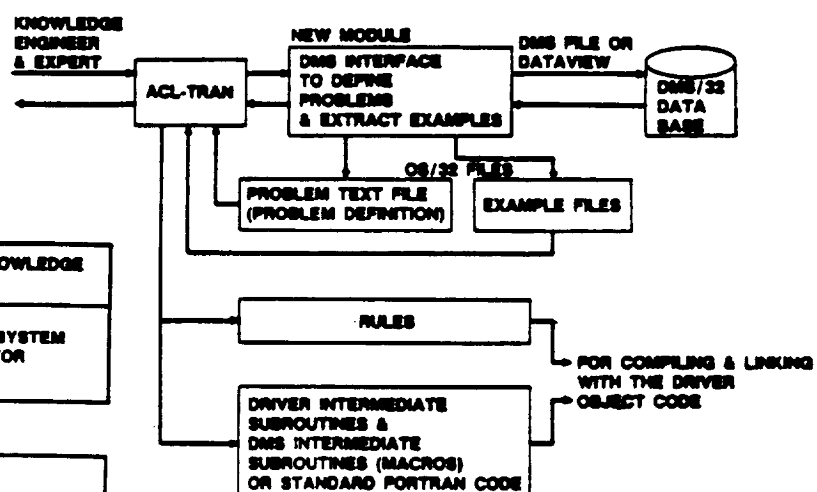

Figure 5. Reliance Expert Development Phase

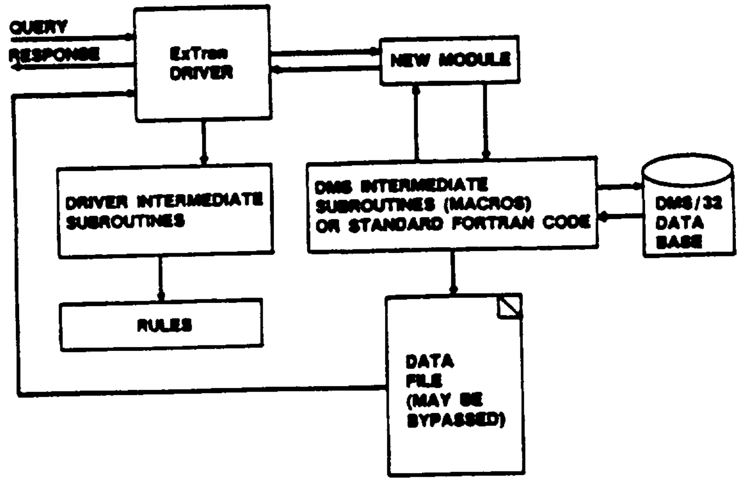

Figure 6. Reliance Expert Run-time Phase

There is a wide selection of existing data bases which could lend themselves to exercising the Reliance Expert product. Anomaly data from SSME testing is one source among several that also include Failure Modes and Effects Analysis (FMEA), turbopump build and history, and hazard tree data. Anomaly data, although primarily hardware-oriented, is a useful source of information. It provides a starting point for converting much of the SSME testing expertise repository into machine readable form. Some efforts are underway to use this source to augment the experience now encapsulated in the heads of senior engineering staff. Each anomaly data sheet consists of three major fields: problem (symptoms), analysis (causes), action for next test and other recommendations. Zero or more anomalies are recorded for each test, usually very minor ones. By carefully reviewing each anomaly and any back-up plots/tables, it is possible to convert each one into an example format consisting of a set of attribute-values and decisions.

Graphics is also being included in future versions of Scotty. A SSME instrumentation chart, now taped to the walls of hundreds of Rocketdyne engineering offices, has been converted to a dynamic color computer graphics form. The graphics subsystem has capabilities to zoom, highlight problem areas (according to actual test data measurements), and depict flow. This is not CAD/CAM, although there are a few common themes, nor is it extensive CFD modeling of the National Aerospace Plane (NASP) using multi-million dollar CRAY 2s. It is a practical and feasible use of moderate color resolution on the readily available super-mini and terminals. Engineers on the floor, as would be expected, are very pleased to see in graphical form what they have hitherto had to dig out of static tables and plots.

FUTURE DEVELOPMENT ISSUES

Further in the future are several concerns. There is an interest in each as a potential contributor to improving the quality of SSME test analysis. Obviously, Rocketdyne is keenly concerned also about technology transfer to other types of engines, in addition to the SSME. The company is deeply committed to supply the power system for Space Station Freedom, as a result of being named the prime contractor. The National Aerospace Plane (NASP) engines are also likely candidates for Rocketdyne. Expendable Launch Vehicles (ELV), the Advanced Launch System (ALS), Orbital Transfer Vehicles (OTV), and other propulsion and energy systems are also promising areas.

These further-reaching concerns are concentrated both in application and technical areas. On the application side, Rocketdyne would like to investigate the potential of extending Scotty to handle a limited subset of the measurement data for flight engines. The incorporation of health and test monitoring is also of high interest. Design of modified and new engines is a challenging option. This could perhaps involve using the current computer model for SSME test analysis to help generate examples for potential design consideration. A recent paper gives some insights on such proposals (Ref. 4). An obvious application is to enlarge the context of Scotty to include new hire training on SSME test analysis.

On the tool side, the issue of dealing with uncertain and/or noisy example data is significant. Real engineering problems involve uncertain and incomplete information. A noted nuclear engineer, Dr. Billy Koen at the University of Texas in Austin, has gone so far as to define the engineering method as "the use of heuristics to cause the best change in a poorly understood or uncertain situation within the available resources" (Ref. 9).

It is apparent, based on recent IJCAI, AAAI and IEEE conferences that induction is receiving considerable attention, so fuzzy induction is probably just around the corner. A recent U.S. based inductive workshop (Ref. 2), just on the heels of an international conference on induction and the founding of an International Special Interest Group on Inductive Programming in 1987, all bodes well for this extremely active area of research. We will see additional and powerful tools on the market which offer such practical features. Recent work at the University of Tennessee Space Institute holds considerable promise for dealing with both qualitative and temporal issues relevant to rocket engine testing (Ref. 8). Abductive reasoning for diagnosis also appears to hold some promise (Ref. 13).

CONCLUSIONS

Since 1984, effort has been underway at Rocketdyne, manufacturer of the Space Shuttle Main Engine (SSME), to automate much of the analysis procedure conducted after test firings. We thus report on progress in building the full Scotty system, after a noted 23rd century rocket propulsion expert.

Major progress has occured on a technical front. Since the very inception of the program, it has been strongly believed that the intrinsic nature of SSME test analysis and character of inductive-based ESBTs represents an excellent match of problem and tool. The intuition has been confirmed by the relative ease with which expertise has been transformed to a structured system of modules composed of examples and thence to effective production rules. The structuring relies upon well-known software engineering techniques, and is aided by commercial CASE tools. The transformation from records in a data base to examples to production rules is accomplished automatically with Reliance Expert, a product combining a RDBMS and an inductive tool. The engineering staff responsible for building (and eventually maintaining) Scotty has consistently used examples as input. The knowledge-acquisition "bottleneck" is thus much wider than for most previously-reported expert systems. The end result is a software system which meets the real needs of Rocketdyne, and is deliverable in a cost-effective manner with less than usual maintenance requirements.

REFERENCES:

1. A-Razzak, R., T. Hassan, A. Ahmed, ExTran 7.2 Users Manual, Intelligent Terminals Ltd., Glasgow, Scotland, 1986.
2. Arciszewski, T., "Potential Applications of Induction in Engineering," Proceedings of the International Workshop on Inductive Programing, Inductive Programming Special Interest Group (IPSIG), Detroit, MI, 1989.
3. Asgari, D. and K. Modesitt, "Space Shuttle Main Engine Test Analysis: A Case Study for Inductive Knowledge-Based Systems Involving Very Large Data Bases," IEEE International Conference on Computer Software & Applications (COMPSAC), 1986, pp. 66-71.
4. Chen, K. and S. Lu, "A Machine Learning Approach to the Automatic Synthesis of Mechanistic Knowledge for Engineering Decision-Making," IEEE Conference on Artificial Intelligence Applications, 1988, pp. 306-311.
5. Concurrent Computer Corporation, Reliance Expert, Version 2 48-xxx F00 R00, Slough, England, 1988.
6. Concurrent Computer Corporation, Reliance DBMS, 04-338 F01 M99 R08, Oceanport, NJ, 1987.
7. Daumann, A. and K. Modesitt, "Space Shuttle Main Engine Performance Analysis Using Knowledge-Based Systems," ASME International Conference on Computers in Engineering, 1985, pp. 55-62.
8. Dietz, W. and M. Ali, "Qualitative and Temporal Reasoning in Engine Behavior Analysis and Fault Diagnosis," NASA Conference on Artificial Intelligence for Space Applications, 1987.
9. Koen, B., Definition of The Engineering Method. American Society for Engineering Education, Washington, D.C., 1985.

10. Modesitt, K., "Space Shuttle Main Engine Anomaly Data and Inductive Knowledge-based Systems: Automated Corporate Expertise," NASA Conference on Artificial Intelligence for Space Applications, 1987, pp. 203-212.
11. Modesitt, K., "Experience with Commercial Tools Involving Induction on Large Data Bases for Space Shuttle Main Engine Testing," Proceedings of the Fourth Expert Systems International Conference, London, 1988, pp. 219-230.
12. Modesitt, K., "Experts: Human and Otherwise," Proceedings of the Third Expert Systems International Conference, London, 1987, pp. 333-341.
13. Nau, D. and J. Reggia, "Relationships between Deductive and Abductive Inference in Knowledge-based Diagnostic Problem Solving," Proceedings of the First Expert Database Systems International Workshop, Benjamin- Cummings, 1986. pp. 549-558.
14. Shapiro, A., Structured Induction in Expert Systems. Addison-Wesley, 1987.
15. Sopp, G., "Scotty: Beaming Up a New Look at SSME Performance," Threshold, Rocketdyne Division, Rockwell International, Canoga Park CA, Number 2, 1987.
16. Tuckwell, R., "Scotty's Enterprise," Computer Systems, Bromley, U.K., August, 1987, pp. 18-20.
17. Warman, D. and K. Modesitt, "A Student's View: Learning in an Introductory Expert System Course," Expert Systems, An International Journal of Knowledge Engineering, Spring, 1988, pp. 30-39.
18. Warman, D. and K. Modesitt, "Learning in an Introductory Expert System Course," IEEE Expert, Spring, 1989, pp. 45-49.
19. Waterman, D., A Guide to Expert Systems. Addison-Wesley, 1986.