1988

SHELL-BASED EXPERT SYSTEMS IN BUSINESS: A RETURN ON INVESTMENT PERSPECTIVE

Stephen R . Ruth
George Mason University

Christopher K. Carlson
George Mason University

SHELL-BASED EXPERT SYSTEMS IN BUSINESS: A RETURN ON INVESTMENT PERSPECTIVE

Stephen R. Ruth
Christopher K. Carlson
Department of Decision Sciences
George Mason University

ABSTRACT

This paper examines an important issue emerging in information systems management--the decision to proceed with an expert system application in a business setting. The focus is knowledge based systems at the lower end of the complexity spectrum--small, very focused systems that can be implemented by the use of shell-based development environments. This group represents the majority of expert systems that are currently being implemented and has some characteristics quite different from the larger systems. A classification scheme is suggested to differentiate three levels of ES development, from multi-million dollar life cycle cost ES environments to those that are in the low five figure range. The Low End segment of the range, the focus of this paper, is characterized by lower unit costs, powerful development tools and a large number of small, successful applications. The important role of Low End systems is discussed, with particular emphasis on their relatively high yield in stand-alone applications. Such systems do not meet the AI demands of moderately or very complex problems but there is a surprising breadth in their use. A group of key success factors for Low End systems is proposed, based on a synthesis of the applications literature. To operationalize these factors, three actual cases using Low End technology--from marketing, government and agribusiness--are briefly described.

Low End systems are not all gain. Their low unit costs can often mask the risks of proceeding headlong into an application without careful examination of the variables that can predict successful results. An agenda for action is offered for specific management policies for the planning of knowledge-based applications.

1. INTRODUCTION

The implementation of Artificial Intelligence-based systems has become one of the most significant new areas of business ADP activity during the past few years. While the AI field is becoming a multi-billion dollar annual investment in the US (Davis 1986; Harmon and King 1985, p. 10), and growing rapidly, a clear trend within that growth has been the increasing emphasis of AI applications toward expert systems. This trend shows the early leaders in AI technology--perceptual systems such as robots, computer vision, and natural language systems--losing their proportion of the AI development activity and expert systems increasing towards half of the total AI market (Harmon, Maus and Morrissey 1988, pp. 22-23).

Expert systems are in essence computer software capable of replicating the decision behavior of an expert in some very focused area of application. They have been called "computer programs that apply substantial knowledge of specific areas of expertise to the problem solving process" (Bobrow, Mittal and Stefik 1986, p. 880) Typically expert systems are composed of a knowledge base, an inference engine and a dialog system. The knowledge base is a body of rules, data, experiences and definitions that contribute to the total body of knowledge. The inference engine is a representation of the way an expert manipulates the knowledge base to achieve expert behavior. In order to join the user and the expert system, a dialog component provides the question-answer link, enabling the expertise of the system to be available to the user. The process of determining the structure of the knowledge base, inference engine and dialog component is often called knowledge engineering.

An expanding literature in the use of expert systems in business indicates a broad variety of applications in areas such as audit (Graham and Steinbart 1986), financial assessments (Hart, Barzilay and Duda 1986), logistics (Allen 1986), materials handling design (Gabbert and Brown 1987), marketing (Conlin, et al. 1987), and decision support systems (Dhar and Croker 1988; Henderson 1987; Remus and Kotterman 1986; Turban and Watkins 1986). An interesting result of the ES explosion has been the availability of software to produce relatively simple, stand-alone expert systems efficiently. AI software products are now available to perform significant tasks on personal computers (Martorelli 1988; Williamson 1986).

2. HIGH PAYOFF ON SMALL EXPERT SYSTEMS: AN EXAMPLE

While there is little data on the total number of expert systems now in full operational use (a rough guess would be about 50,000 systems, mostly relatively small and highly focused in application), two things are clear: some of the largest companies are investing major resources in expert systems and the typical application is relatively small (Kupfer 1987). DuPont, perhaps the most enthusiastic user of expert systems technology of all United States corporations, has aimed their approach squarely at the Low End of the ES spectrum. The director of DuPont's Expert System effort claims about two thousand expert systems in stages from planning to prototype to actual implementation (*Computerworld* 1987). Investment expense is controlled in the range of $10,000 to $20,000 per system and the payoffs across the company are, on average, in the low to mid six figures for each system--a handsome return of 10 or 20 to one.

3. CHARACTERIZING EXPERT SYSTEMS APPROACHES

While many of the best known expert systems applications such as XCON, a system for configuring Digital Equipment's VAX computer (Bachant and McDermott 1984), are characterized by life cycle expense in the range of a million dollars or more, the typical expert system has a far smaller investment. It is possible to conceptualize the expert system spectrum as shown in Table 1, which differentiates three levels of investment in expert system technology, separated by a range three orders of magnitude in life cycle expense. The categories and costs are speculative, but based on aggregations of data furnished in the general literature, reports and some site visits.

The million dollar systems, called High End, often have tens of thousands of rules or rule equivalents and are typically developed using languages such as LISP and Prolog or AI environments such as ART, KEE and KnowledgeCraft. These systems require major investments in computer hardware, workstations, software, and knowledge engineering personnel, and often take several years to complete.

The intermediate approach, Mid-Range, with life cycle costs in the low six figures, uses high level languages but has fewer rules or rule equivalents. Some Mid-Range expert systems use a development environment called a "shell" to generate the system more quickly. Shells facilitate ES development by setting up a pre-defined structure that allows the programmer or knowledge engineer to fill in a template for the knowledge base, the inference engine and the dialog system, rather than having to develop detailed code for each segment using a generalized language.

Table 1. Comparison of Three Approaches to ES Development

	Developmental Approach		
	Low End	**Mid Range**	**High End**
Typical Total Life Cycle Cost	$10,000- $70,000	$100,000- $600,000	$1 million- $5 million
Typical Number of Rules or Rule Equivalents	50-400	350-2,000	1,000-100,000
Unit Cost of Rules or Rule Equivalents	$40- $100	$150- $400	$200- $500
Typical Development Environment	Shells	Shells; High Level Languages	High Level Languages Advanced AI Languages
Typical Language or Shell	EXSYS, 1STCLASS, Insight 2+, TI-PC EASY	GURU, M1 KSS, NEXPERT LISP, PROLOG	LISP, PROLOG ART, KEE, KNOWLEDGE-CRAFT
Typical Computer Hardware	Microcomputer	Supermicro; Minicomputer (LISP Machine)	Minicomputer; Main Frame
*Estimated ES Applications Share	80-90%	5-15%	2-5%

*Refers to each developmental approach's estimated share of the total number of operational expert systems currently developed.

The third category of expert system, the Low End, probably represents the great majority of those now in use in the United States. These smaller systems have a narrow scope and more focused body of knowledge to capture. The low-end knowledge-based systems, consisting typically of a few hundred rules and programmed with powerful but relatively inexpensive shells, such as EXSYS, Insight 2+ or 1STCLASS/ FUSION are being used by organizations that have opted for a larger number of relatively focused, stand-alone systems as opposed to the High End, large integrated systems. An interesting point associated with this level of expert system is also seen in Table 1: the *unit* life cycle cost of programming expert systems decreases from the million dollar approach to the shell approach. It has been estimated (Mahler 1986) that the unit cost of the LISP or PROLOG programs is nearly an order of magnitude greater than that of the shell-based systems at the low end of the ES spectrum. In this view, over the life cycle of a LISP or PROLOG ES the cost of a line of code is about one person day while for a shell system it is one person hour. Harmon and King (1985, p. 9) and Gevarter (1987, p. 24) make this same point in the context of the rapidly improving technology and the resultant declining costs of developing a knowledge system. As Gevarter states (1987, p. 40), "expert system building tools (ESBTs) have made possible productivity improvements of an order of magnitude or more in constructing expert systems. Current tools are only forerunners of ESBTs yet to come."

4. PICKING THE RIGHT APPLICATION FOR A SMALL ES

Selecting the right application for an expert system can be a significant issue, one affecting the company's long term competitiveness. We concentrate on that question and aim our discussion at the popular and somewhat under reported Low End of the ES development spectrum: shell-based systems of only a few hundred rules. These represent the most typical application in many companies and are the fastest growing segment of the market. They are low in unit cost, relatively quickly implemented and can be developed and run on a microcomputer. The literature on expert systems development describes dozens of possible factors which may determine the appropriateness of an expert systems application. A partial but representative list of these factors is presented in Table 2. When the expert system application is at the low end of the spectrum, these factors need to be defined more specifically. We describe below six factors that seem particularly crucial to the success of the smaller knowledge-based systems.

Table 2. Examples of Suggested Factors that Contribute to the Success of an Expert System Application

• Availability of expertise	• Facts stated as ideas
• Task of manageable size	• Task does not involve common sense
• Task requires cognitive, not physical skills	• Consistency of problem-solving is crucial
• Leverage points easy to determine	• Strong economic incentive exists
• Symbolic/heuristic, not algorithmic, task	• Expertise is typically expressed symbolically

4.1 Key Factor for Low End ES: Extremely Narrow Focus

Particularly in the case of problems adapted for Low End ES shells, it is crucial that the application be highly specific and its boundaries very narrow and focused. With too broad a focus, the system could appear to be functioning appropriately but could miss a crucial element of expertise. Such systems, as Sheil (1987, p. 94) states, "don't know what they don't know." Since a typical shell-based application will have only a few hundred rules, the knowledge base is likely to have only a few dozen premises. For example, in a credit analysis ES, each premise about the person's eligibility for credit would necessarily have many outcomes, each of which gives rise to a probability-based rule. Table 3 shows an example of one such rule and the user question from which the rule is derived. It can be seen that the single premise "Based on the previous loan history, the individual" gives rise to at least five rules, each similar in structure to rule 46 shown in Table 3. It could be argued that an expert system with only several dozen premises might not be particularly valuable, but there are indications that the behavior of many experts is often reducible to a few hundred rules within a specific application area.

Table 3. Sample of Five-Premise Screen in EXSYS for an Automobile Loan Processing Case

Shown below the user query is the embedded rule from the knowledge base that corresponds to the third premise.

Based on previous loan history, the individual
1 has never been disapproved for a loan
2 has one or two previous disapprovals
3 has several previous loan disapprovals
4 has a great deal of previous loan disapprovals
5 has never applied for a loan before

RULE NUMBER: 46

IF:

(1) Based on previous loan history, the individual has several previous loan disapprovals

THEN:

OFFER THE LOAN - Probability=6/10
and DENY THE LOAN - Probability=3/10

NOTE: **A large amount of previous loan disapprovals can indicate a financial problem.**

REFERENCE: **The Consumer Loan Department**

Source: Mr. Kurt Smith, NAVY Federal Credit Union, Vienna, Virginia

The accounting firm of Coopers and Lybrand developed *Expertax*, a powerful expert system used for tax planning. Kneales (1986, p 33), the director of the project, was surprised at first that experts truncate their decision behavior but found that a relatively small number of rules is often enough to capture certain types of expertise. In his words: "Suddenly all this bull about 'I've been doing this for twenty years and every case is different' disappears" (Kupfer 1987, p 78).

Table 4. Sample from Variable Screen for a 1STCLASS Implementation for Assigning US Navy Enlisted Personnel

SEX	RATE	MARITAL	PREG	#FROTAT	RESULT
MALE	OS	SINGLEND	PREG	OUTUS	#YNMALE
FEMALE	PN	SINGLED	SPPREG	CONUS	#YNFEMALE
	YN	MARRIED	NOTPREG		#OTHER
	BM	MARRIEDMIL			#OTHER1
	HT				#OTHER2
	PC				#OTHER3
	QM				#OSMALE
	RM				#PNMALE
	SK				#BMMALE
	MS				#HTMALE
					#PCMALE
					#QMMALE
					#RMMALE
					#SKMALE
					#MSMALE
					#OSFEMALE
					#PNFEMALE
					#BMFEMALE
					#HTFEMALE

Source: Mr. Ron Evans, School of Business Administration, George Mason University, Fairfax, Virginia

Table 5. Sample from Examples Screen for a 1STCLASS Implementation for Assigning US Navy Enlisted Personnel

SEX	RATE	MARITAL	PREG	#FROTAT	RESULT
MALE	OS	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	PN	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	YN	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	BM	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	HT	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	PC	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	QM	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	RM	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	SK	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	MS	MARRIED	SPPREG	OUTUS	#OTHER1
MALE	OS	MARRIED	NOTPREG	OUTUS	#OSMALE
MALE	PN	MARRIED	NOTPREG	OUTUS	#PNMALE
MALE	BM	MARRIED	NOTPREG	OUTUS	#BMMALE
MALE	HT	MARRIED	NOTPREG	OUTUS	#HTMALE
MALE	PC	MARRIED	NOTPREG	OUTUS	#PCMALE
MALE	QM	MARRIED	NOTPREG	OUTUS	#QMMALE
MALE	RM	MARRIED	NOTPREG	OUTUS	#RMMALE
MALE	SK	MARRIED	NOTPREG	OUTUS	#SKMALE
MALE	MS	MARRIED	NOTPREG	OUTUS	#MSMALE

Source: Mr. Ron Evans, School of Business Administration, George Mason University, Fairfax, Virginia

4.2 Key Factor in Low End ES: Example-Based Knowledge Structures

The importance of focused rules is even more evident when the shell environment is based on the expert describing examples, not rules. These so-called inductive shells are useful when the domain expert has difficulty in describing rules but can readily give examples of typical decision behavior. The expert need only provide samples of his or her behavior and the shell is able to induce the rule structure from the examples. 1STCLASS and TIMM are examples of inductive shells.

With inductive shells, the number of variables must be very few since there is a multiplicative effect as variables are added. Table 4 describes five variables in a personnel selection expert system for assigning the next duty station for United States Navy enlisted personnel. The personnel expert, called a "detailer" in the Navy parlance, is confronted with a relatively large number of choices for some variables and at least two for all variables. The total number of possible cases needed to complete the expertise matrix equals the product of the number of all the possible values of each variable. The practical result of this effect is that most shell-based inductive systems need to be limited to a relatively small number of variables, usually in the range of six or seven, to stay within reasonable size constraints. This is reminiscent of Miller's landmark article, which provided evidence that this is also a practical limit common to many other sensory processes (Miller 1957). Table 5 shows samples of the hundreds of examples required to complete this expert system. Leaving gaps in the examples is a frequent strategy for dealing with this combinatorial problem. The shell, like many domain experts, does not always have an answer.

4.3 Key Factor in Low End ES: Experts and Knowledge Engineers

The apparent ease of use of the typical shell packages and their increasing availability in university curricula (Ruth 1988) may mask the fact that knowledge engineers and domain experts, not the ES shell programs, are the key to good expert systems development no matter what the developmental environment. Typical development cycles for shell-based ES are characterized by knowledge engineering being the major expense (Cupello and Mishelevich 1988; Harmon and King 1985, p. 195). Once the rules, variables, examples, dialog vocabulary and other key issues are examined by the knowledge engineer, the shell facilitates the translation to an expert system. The availability of qualified knowledge engineers is equally crucial in shell-based environments as in more complex applications. Fortunately, the relative simplicity of the Low End systems often makes it possible for the domain expert to do his or her own knowledge engineering, reducing expense and elapsed time significantly.

The organizations that have been successful in implementing shell-based ES have been characterized by careful attention to the selection of knowledge engineering personnel. The specialists can be trained from within or may be outside consultants. The DuPont approach, where most knowledge engineering specialists are company employees working in the functional area, is successful but the approach used by the American Express Company, where outside knowledge engineers are often employed, has also yielded positive results (Kupfer 1987, p. 74; Silverman 1987, pp. 8-9; Newquist 1987). Other organizations are undertaking experimental computer technology efforts in their main line activities by providing knowledge engineering support from specialized subsidiaries, as in the case of Security Pacific National Bank's Automation Company (Eliot 1988) and NCR's Advanced Systems Development (Rolandi 1988).

The insight from large as well as small ES development experiences is the same: the major expense and time commitment must be dedicated to the knowledge engineering phase of the ES development life cycle. There may be additional categories of knowledge engineering expense areas to be considered. Fried (1987) examined a large number of ES implementations, including a representative sample of Low End applications, and concluded that many frequently fail to plan for the added expenses of end user implementation and user interface. Incidentally, the increasing cost of trained knowledge engineers, often exceeding the salary of directors of EDP or engineering, may be a good reason for training in-house persons in this emerging specialty.

4.4 Key Factor in Low End ES: Hardware/Software Availability

Expert system development in a Low End shell-based environment does not depend on the availability of minicomputers or main frame resources; it is characterized exclusively by the use of microcomputers. Since the typical application has only a few hundred rules, the microcomputer is easily capable of accommodating the knowledge base, inference engine and dialog system no matter what the shell selected. In cases where the upper limits of the shell's capability are tested, micros are usually still quite capable of achieving the required performance, although PC-AT class machines or equivalents may be required.

A caveat is required here: Some of the shell environments are open ended enough to allow the application to grow well beyond the applications levels we are emphasizing in this article. For example, one of the most powerful rule based ES shells, EXSYS, has an upper limit of 5000 rules. Even though the typical EXSYS program uses only a few hundred rules, it is possible to group a series of EXSYS applications into a several thousand rule ES. This size system requires a much more expensive approach and could easily require a supermicro, minicomputer or even a main frame to run effectively.

The problems inherent in linked and cooperating knowledge bases are only recently being addressed in work on expert system shells (Bobrow, Mittal and Stefik 1986; Stefik 1986). Considerable effort is occurring in the area of integrating the new ES technology with existing computing bases (Kerschberg 1987; Pedersen 1988). There is a clear indication that stand alone ES, like end user computing in general (Huff, Munro and Martin 1988), will also need to pass through several stages of application maturity growth.

The software at the low end is diverse and powerful. Similar tothe problem of selecting a company-wide word processing package, there are a number of very good shells to choose from. Unlike the word processing decision, it is not a serious problem to select three or four. For example, many companies are selecting both an inductive and a rule based shell. For rule-based shells, they might choose a low cost package such as EXSYS or Insight 2+ (about $500 each) and a more expensive but more powerful package such as GURU or M.1 (both about $5,000). For the inductive shells, DuPont has selected both 1STCLASS and TIMM (both in the $500 to $1000) range and INSIGHT 2+ for the rule-based expert systems (Fersko-Weiss 1986).

4.5 Key Factor in Low End ES: Determination of Leverage Points

Because Low End ES development is low in unit cost, there is a possibility that an organization may consider developing ES applications without considering the feasibility, size of gain, strategic value and other issues that should be part of any information systems decision. While the questions may be somewhat different than those for information systems, the importance of careful analysis of the points of leverage made possible by the ES application is still crucial (Leonard-Barton and Sviokla 1988). Some typical questions that could be asked to determine whether the proposed Low End application will have some degree of leverage in the company are:

- Will the spreading of expertise in this very limited area enable us to reduce contingency funding?
- Are we in danger of losing this expertise?
- Do we need a more consistent approach company-wide in this decision area?
- Is the information codified somewhere else but hard to access?
- Will this ES affect our competitiveness in our industry?
- Will this application broaden the skills and potential of our employees? (For example, will the salesmen be more skilled and successful with a new ES that assists them in describing price breaks for a complex product line?)

4.6 Key Factor in Low End ES: Examining the ROI

One should be prepared for some surprising discontinuities when considering the potential payoffs of shell-based ES. Earlier we mentioned the estimate that there may be an order of magnitude advantage in the unit cost of program development in favor of the shell based systems. This means that an ES developed in KEE, ART, LISP or PROLOG with, say, 25,000 rule equivalents might cost about $400 per instruction or rule equivalent, while a shell-based program of, say 400 rules, would cost perhaps $60 per rule. Low End applications, by definition, are inherently less complex, easier to program and simpler to implement.

A businessperson would be quick to point out that higher expenditures can often yield higher total benefits. There might be hundreds, perhaps thousands, of small ES applications in a large company that might yield 10 or 20 times the investment, but very few that would leverage a half million dollar investment in the same way. Hence, only a handful of 10 to 1 or 20 to 1 return on investment successes in small ES could quickly have the same effect on the bottom line as a full recovery of costs on an expensive High End ES application. Small systems may offer a needed degree of risk diversification lacking in High End systems. There are many other cost issues beyond simply unit expense of development. A judicious mix of small scale and large scale projects is a good solution. The apparent unit cost and unit profit benefits of the small scale projects are often overlooked when organizations begin to move into AI as an element of corporate strategy.

The small scale projects also offer an ideal entry point to learn the best applications for future AI investment, allowing the organization to get its corporate feet wet before making a major commitment.

4.7 Low End Expert Systems in Action: Three Brief Examples

DuPont: Marketing MYLAR. DuPont produces a wide variety of film in the MYLAR family, products that are used in industrial processes, construction and in the home. An ES was developed using the Insight 2+ shell to aid the sales representative in selling these products, showing price breaks, special features, recommended substitutes, etc. The ES has proven popular and is often used on sales calls with the salesman employing a laptop computer to access the ES while dealing with the client. The shell-based system paid for its $10,000 initial investment within three months of use. Total benefits have been computed to be in the low six figures (Mahler 1988).

Navy Supply Systems Command. This organization is responsible for the United States Navy's logistics management. It has developed an expert advisory system for evaluating decisions in supply contract termination actions (NAVSUP letter 1988). The system, which cost about $15,000, may be able to pay for itself during the first month of use. The leverage, and high payoff, in this application arises from the large number of users who will operate this system in their daily work activities. Several hundred contract item managers across the United States will employ the advisory system in considering recommendations for termination. Each potential termination action may be evaluated in minutes using the system as opposed to hours manually. The system will further serve to enhance consistency and standardization of practices across the large and diverse body of users. Finally, the advisory system provides requisite documentation of processing results in both hard copy and electronic form. The documentation supports subsequent processing actions and provides input for later studies by Navy cost analysts.

PEANUT. This EXSYS-based expert system has been developed by the Department of Agriculture to aid in one of the most crucial decisions in agribusiness: when to irrigate. While the focus is only one kind of peanut in a particular region of the United States, the *florunner* in Georgia, it is representative of many others. Data on potential return on investment is not yet available, but an insight from a micro-based expert system used for cotton growing, called COMAX, is useful. Lemmon (1986) reported that a southern farmer admitted that he lost over $400,000 in his 1986 cotton harvest because he ignored COMAX's recommendation to harvest in early September rather than at the end of the month. As the article concluded "The grower now believes that the maturity date of 1 September was correct and that, if the harvest had begun on that day, cotton production would have been increased by approximately 4.3 million pounds and the quality would have been improved by an amount worth an additional $0.11 per pound" (Lemmon 1986).

5. AGENDA FOR ACTION

The theme of this article is that some of the major advantages to be obtained in AI are emerging in expert systems, especially the most widely used branch of ES: the shell-based applications of several hundred rules which we have characterized as the Low End approach. We recommend that prospective users of ES technology consider the criteria we have synthesized in any ES decision. In many cases it will be obvious that the best way to begin with ES is at the Low End. Aside from having a very favorable entry cost, Low End system technology can become a part of the organization's competitive strategy in partnership with larger systems.

The most promising strategy would seem to be an ES portfolio that has, after a period of maturation, a mix of many Low End, some Mid-Range and a few High End applications. The important challenge for the manager who seeks to achieve the high yield opportunities inherent in this technology is to dedicate most of the expense and the attention to the process of extracting the knowledge and rules. Knowledge engineering, if done well, will always be the most reliable predictor for ES success.

6. REFERENCES

Allen, M. "The Development of an Artificial Intelligent System for Inventory Management." Unpublished Doctoral Dissertation, Ohio State University, 1986.

Bachant, J., and McDermott, J. "R1 Revisited: Four Years in the Trenches." *AI Magazine*, Vol. 5, No. 3, Fall 1984, pp. 21-32.

Bobrow, D.; Mittal, S.; and Stefik, M. "Expert Systems: Perils and Promise." *Communications of the ACM*, Vol. 29, No. 9, September 1986, pp. 880-894.

Computerworld. "Interview, Dr. Ed Mahler." Vol. 21, No. 47, November 23, 1987, pp. S9-S11.

Conlin, M.; Strohm, G.; Sathi, A.; Pinkus, A.; and Dumont, F. "A Goal Directed Approach to Data Access Organization in Telemarketing." In L. Kirschberg (ed.), *Proceedings of the First Annual Conference on Expert Systems in Business*, 1987, pp. 45-54.

Cupello, J., and Mishelevich, D. "Managing Prototype Knowledge/Expert System Projects." *Communications of the ACM*, Vol. 31, No. 5, May 1988, pp. 534-541.

Davis, D. "Artificial Intelligence Enters the Main Stream." *High Technology*, July 1986, pp. 16-23.

Dhar, V., and Croker, A. "Knowledge-Based Decision Support in Business: Issues and a Solution." *IEEE Expert*, Vol. 3, No. 1, Spring 1988, pp. 53-62.

Eliot, L. "Security Pacific's RESRA: A Case Study." *AI Expert*, Vol. 3, No. 4, April 1988, pp. 48-59.

Fersko-Weiss, H. "AI Update." *Personal Computing*, December 1986, p. 105.

Fried, L. "The Dangers of Dabbling in Expert Systems." *Computerworld*, Vol. 21, No. 30, June 29, 1987, p. 71.

Gabbert, P., and Brown, D. "MAHDE: An Intelligent Materials Handling Design System." In L. Kerschberg (ed.), *Proceedings of the First Annual Conference on Expert Systems in Business*, 1987, pp 87-94.

Gevarter, W. "The Nature and Evaluation of Expert System Building Tools." *Computer*, Vol. 20, No. 5, May 1987, pp. 24-41.

Graham, G., and Steinbart, P. "The Use of Rule Based Expert Systems to Investigate the Effects of Experience in Audit Judgments." In L. Maggi, R. Zmud, and J. Wetherbe (eds.), *Proceedings of the Seventh International Conference on Information Systems*, San Diego California, December 15-17, 1986, pp. 206-213.

Harmon, P., and King, D. *Expert Systems: AI in Business*. New York: Wiley, 1985.

Harmon, P.; Maus, R.; and Morrissey, W. *Expert Systems: Tools and Applications*. New York: Wiley, 1988.

Hart, P.; Barzilay, A.; and Duda, R. "Qualitative Reasoning for Financial Assessments: A Prospectus." *AI Magazine*, Vol. 7, No. 1, Spring 1986, pp. 62-68.

Henderson, J. "Finding Synergy Between Decision Support Systems and Expert Systems Research." *Decision Sciences*, Vol 18, No. 3, Summer 1987, pp. 333-349.

Huff, S.; Munro, M.; and Martin, B. "Growth Stages of End User Computing." *Communications of the ACM*, Vol. 31, No. 5, May 1988, pp. 542-550.

Kerschberg, L. (ed.). *Proceedings from the First International Conference on Expert Database Systems*, Menlo Park CA: Benjamin/Cummings, 1987.

Kneales, D. "How Coopers and Lybrand Put Expertise Into Its Computers." *Wall Street Journal*, November 14, 1986, p. 33.

Kupfer, A. "Now, Live Experts on a Floppy Disk." *Fortune*, Vol. 116, No. 8, October 12, 1987, pp. 69-82.

Lemmon, H. "COMAX: An Expert System for Cotton Crop Management." *Science*, Vol. 233, July 4, 1986 pp. 33.

Leonard-Barton, D., and Sviokla, J. "Putting Expert Systems to Work." *Harvard Business Review*, Vol. 66, No. 2, March-April 1988, pp. 91-98.

Mahler, E. Private communications with authors. June-July 1988.

Mahler, E. "The Business Needs Approach." Lecture on Texas Instruments' Second AI Satellite Seminar - Knowledge-Based Systems, Spring 1986. (Available on VCR tape from Texas Instruments Company, Dallas, Texas.)

Martorelli, W. "PC-Based Expert Systems Arrive." *Datamation*, Vol. 34, No. 7, April 1, 1988, pp. 56-66.

Miller, G. "The Magic Number Seven, Plus or Minus Two: Some Limits on Our Capacity For Processing Information." *Psychological Review*, Vol. 63, No. 2, pp. 81-97.

NAVSUP letter. "Termination Policy." 25 February 1988, internal document. U. S. Navy Supply Systems Command, Washington, D. C.

Newquist, H. "American Express and AI." *AI Expert*, Vol. 2, No. 4, April 1987, pp. 63-65.

Pedersen, K. "Connecting Expert Systems and Conventional Environments." *AI Expert*, Vol. 3, No. 5, May 1988, pp. 26-35.

Remus, W., and Kottemann, J. "Toward Intelligent Decision Support Systems: An Artificially Intelligent Statistician." *MIS Quarterly*, Vol. 10, No. 4, December 1986, pp. 403-418.

Rolandi, W. "A Practical Approach to Knowledge Engineering." *AI Expert*, Vol. 3, No. 4, April 1988, pp. 60-65.

Ruth, S. "Introducing Expert Systems in the Business School: A Shell Game." *Interface*, Vol. 9, No. 4, Winter 1988, pp. 42-49.

Sheil, B. "Thinking About Artificial Intelligence." *Harvard Business Review*, Vol. 65, No. 4, July-August 1987, pp. 91-97.

Silverman, B. (ed.). *Expert Systems for Business*. Reading MA: Addison-Wesley, 1987.

Stefik, M. "The Next Knowledge Medium." *AI Magazine*, Vol. 7, No. 1, Spring 1986, pp. 34-46.

Turban, E., and Watkins, P. "Integrating Expert Systems and Decision Support Systems." *MIS Quarterly*, Vol. 10, No. 4, December 1986, pp. 403-418.

Williamson, M. *Artificial Intelligence for Microcomputers*. New York, NY: Brady, 1986.