# CHARACTERISTICS AND ADOPTION OF GENERIC FINANCIAL EXPERT SYSTEMS: A CASE STUDY OF FAILURE

**SUNG J. SHIM**

FAIRLEIGH DICKINSON UNIVERSITY

## ABSTRACT

Several software vendors have attempted to develop and market "generic" financial expert system (GFES) products for multiple users across organizations. While these expert systems (ES) share certain similarities with other ES, they have their own special characteristics that may pose some difficult issues for potential adopters. This case study examines the characteristics of GFES and related adoption problems, based upon the experience of an electronics company with Palladian Software's Management Advisor. The results show that the system can provide superior analytic capability going well beyond straightforward spreadsheet or quantitative modeling approaches of conventional financial software. But they also show that the system's problems, particularly limited explanation capability and lack of compatibility and modifiability, are too serious to overcome. The lessons from this case study will be helpful to those involved in financial ES projects.

## INTRODUCTION

Financial decision-making problems have represented an important set of expert system (ES) applications. While many ES have been developed for financial applications, not every system has met with implementation success. Particularly, the attempts of several software vendors including Applied Expert Systems, Palladian Software, and Syntelligence towards "generic" financial ES (GFES) for multiple users across organizations (see Table 1) have not been so successful. Palladian Software went out of business after several years of struggling to sell Management Advisor (MA). Syntelligence's Lending Advisor, probably the best known GFES so far, has not penetrated more than a few banks [11], and Syntelligence has eventually filed for bankruptcy. Applied Expert Systems' PlanPower and Client Profiling System may be still operational in some places, but the vendor appears to have already curtailed its commitment to these systems. The instances of these GFES do not mean that GFES in general are not viable, as it is hard to determine how well other GFES have fared in the market. But they may suggest that such a generic approach to financial ES applications poses some difficult issues for potential adopters.

**Table 1. Examples of Generic Financial Expert Systems**

| Vender | System | Application Domain |
|---|---|---|
| Applied Expert Systems | Client Profiling System | Client account management |
| Applied Expert Systems | Plan Power | Financial Planning for Individuals |
| Paladian Software | Management Advisor | Capital budgeting |
| Syntelligence | Lending Advisor | Credit & loan analysis |
| Syntelligence | Underwriting Advisor | Commercial underwriting |

Note: In addition to the GFES listed above, there might have been other GFES. But the systems listed above have been actually released for commercial use and are rather extensively described in the press and professional journals. When Palladian Software went out of business, the technology license of the company's products including Management Advisor and Operations Advisor (another generic ES product in the operations area) was taken by Carnegie Group. Since the early 1990s, Crowe Chizek & Company has taken on the marketing of Syntelligence products in the banking industry, e.g., Lending Advisor as a companion product to Crowe Chizek's Financial Analyst's Management and Authoring Systems.

This case study examines the characteristics and related adoption problems of GFES, based upon the experience of an electronics company with Palladian

Software's MA. We first describe the characteristics of GFES in terms of three innovation characteristics identified as being most related to the innovation adoption: relative advantage, complexity, and compatibility. Then, we look at how these characteristics are perceived by users and related to their adoption decisions, using data obtained from MA users at the company. While they may not be generalized to other GFES or user organizations, the lessons learned from this case study can help better understand the plausibility of GFES from the viewpoint of users. We hope that the lessons will be useful to those involved in ES projects, particularly those considering a generic approach to financial ES applications.

## CHARACTERISTICS OF GENERIC FINANCIAL EXPERT SYSTEMS

As GFES are a relatively new information technology, it is useful to look at their characteristics in terms of the innovation characteristics identified as being related to the innovation adoption. Considerable research has been undertaken to describe the characteristics of an innovation and their relationship to the innovation adoption. Of the various characteristics, relative advantage, complexity, and compatibility are found to have the most consistent and significant relationship to an adoption [17]. Relative advantage is the degree to which an innovation is perceived as being better than the idea it supersedes [13]. Complexity is the degree to which an innovation is perceived as relatively difficult to understand and use [13]. Compatibility is the degree to which an innovation is perceived as being consistent with the existing values, past experiences, and the needs of the receiver [13]. It is generally agreed that both relative advantage and compatibility are positively related to adoption, whereas complexity is negatively related to adoption [17]. We describe below the relative advantage, complexity, and compatibility of GFES, grounded in the context of those GFES listed in Table 1.

### Relative Advantage

Several studies, e.g., [1][18], have examined the advantages or benefits of ES in organizations. They report that ES can provide such benefits as improved decision making, greater consistency and continuity in decision making, faster response time, operational cost savings, and more readily available information. The major advantage of ES over conventional computer programs comes from their ability to capture, represent and disseminate expert knowledge, particularly for problems that are not well structured. ES use symbolic processing, that is, they determine the results or decisions by rules of thumb or heuristics. By contrast, conventional computer programs employ numerical manipulation and provide algorithmic solutions, and so they do a better job with tasks that bear a clear structure. Because many business problems cannot be adequately solved by numerical manipulation alone, ES can provide a means to solve these otherwise inaccessible problems through the use of heuristics, often in conjunction with numerical manipulation. For example, we may perform such numerical manipulation as "what-if" analysis in designing investment plans. But when the number of possible investments is large (e.g., PlanPower looks at over 100 different types of investments [10]), doing what-if analysis on all possible combinations of investments may be impractical, and so our heuristics become a critical element of the investment planning process.

Most financial decision-making problems center on assessing the risks and rewards associated with alternative courses of action. Although they involve an analysis of large amounts of numerical data, these assessments typically involve unstructured tasks associated with uncertainty [6]. For example, an investment officer designing investment portfolios, a financial analyst budgeting capital projects, or a bank officer analyzing credits, cannot have complete information about the myriad of events that will govern the outcome of a decision over time. Of course, ES cannot solve all unstructured financial decision-making problems, but they are often the best tools for managing and solving them. It is also widely recognized that real-world usage of normative financial models is infrequent, due to such problems as complexity of decision-making environment, lack of necessary information, and difficulty of usage [15]. ES have the potential to solve these problems and to distribute modern financial models and expertise to those who otherwise would not have time, training or experience to use them effectively.

In addition, generic ES products provide several advantages over other ES developed in house. Two main problems often cited in regards to ES development are the lack of knowledge engineers and high development costs [1][14]. It is widely recognized in the artificial intelligence (AI) community that there is a marked scarcity of knowledge engineers and that it takes time to develop people with the skills needed to make ES work. Even with a team of people with the right mix of talent, including experts, knowledge engineers, programmers, and participating users, an ES project can take several months to several years to design and implement. But generic ES products, like other off-the-shelf software packages, are readily available to users.

### Technology and Knowledge Complexity

The complexity of an ES is largely determined by its embedded technology complexity and knowledge complexity. Technology complexity is the depth and scope of the programming effort, the user environment, and the technical efforts involved in building and implementing an ES [9]. Knowledge complexity is the degree of depth and specialization of the internal knowledge of human experts, the scope of the decision-making process, and the level of expertise required, including discipline-based knowledge incorporated into the ES application [9].

The first ES utilized a single AI method like "if-then" or production rules. With advances in AI technology, various AI concepts and methods have become utilized to build more powerful and efficient ES. GFES are composite systems in that they combine different AI approaches with each approach being employed where it is most effective. For example, MA utilizes a number of AI representation and control concepts including contingent inheritance hierarchies, object-oriented graphics, production rules, model-based reasoning, goal-directed control, and constraint-based inference [12]. Taken together, such a composite system is expected to provide far more power and efficiency than can be obtained by using any single method. But designing a composite system would involve more extensive programming and technical efforts than designing a simple rule-based system. PlanPower took approximately two years from design to delivery [4]. Palladian Software's MA and Operations Advisor also took about two years to develop, and the vendor invested about $18 million in the development of these two systems. Syntelligence invested more than 205 man-years and $20 million in the development of several GFES products including Lending Advisor and Underwriting Advisor [3].

Built upon composite AI concepts and methods, GFES also contain comprehensive expert knowledge in their specialized domains. It is worthwhile to note that those GFES products in Table 1 have been developed by specialized AI software vendors utilizing world-class professionals in finance as well as AI. Several renowned academicians served as domain or technology experts in the development of MA. A group of executives and academicians with expertise in financial planning services were key players in developing PlanPower [8][16]. Syntelligence's GFES products also incorporate almost the same level of expert knowledge from individuals who have developed ES track records [10]. Incorporating such a great deal of expert knowledge, those systems are among the largest in scale and the most comprehensive in analytic capability in their respective financial domains. MA, which consists of about 200,000 lines of codes, analyzes not only the financial value but also the risk, competition, timing, flexibility, and overall business impact of capital budgeting projects [12]. PlanPower incorporates about 1,200 chunks (roughly equivalent to 6,000 simple rules) of heuristic procedure in about 250,000 lines of codes [16]. PlanPower provides coverage of all the areas involved in the creation of financial plans: estate planning, insurance, tax advice, and so forth [8]. Syntelligence's GFES products also have comprehensive financial-risk-assessment knowledge bases and range in size from 800 to over 2,000 nodes with chains more than 100 functions deep and relations containing over 100,000 values [3].

### Task Compatibility

Compatibility is an important factor for success in information systems implementation [2]. Like other information systems, ES are supposed to be compatible with user tasks. Particularly, the conceptual model or heuristic procedure underlying an ES should be congruent with user tasks. In this regard, generic ES products developed by outside vendors may be less compatible with specific user tasks than ES developed in house. While generic ES contain comprehensive domain knowledge as described above, outside vendors may not understand user problems and needs as closely as the users themselves. Instead, generic ES intend to provide a certain body of generic domain knowledge relevant over broad tasks, as they cut across organizations and specific users and applications.

Generic knowledge may be useful by itself, but it needs to be applied to specific user tasks. There are simple financial models applicable to a broad range of tasks such as net present value calculations. But these simple models, which are widely used in conventional spreadsheets and financial systems, are usually only for certain parts of the comprehensive analysis in GFES. Beyond these standard models, GFES contain complex models and knowledge based upon vendor's point of view or set of beliefs and try to apply them to broad tasks in a certain domain. But users may have different procedures and practices for the tasks targeted by the proposed system. For example, MA's financial input to capital budgeting analysis was initially built based upon cash flow statements, as the vendor believed it as a theoretically more correct method [7]. But some early adopters of the system were found to do their capital budgeting analyses based upon accounting statements, and after all, the vendor had to go through an extensive redesign of the system to provide an input side dealing with accounting terms as well as cash flow terms [7]. Thus, GFES may be less compatible with user tasks than other ES developed by users or at least with some participation of users.

## ELECTRONCO'S EXPERIENCE WITH MANAGEMENT ADVISOR

As described above, GFES have their own special characteristics, while sharing similarities with other ES. Adopting a GFES would entail both advantages and problems largely associated with these special characteristics. In this section, we look at how these characteristics are perceived by users and related to the adoption, based upon the experience of Electronco (the company's name is disguised at its request) with Palladian Software's MA. Electronco is a large producer of various technology products ranging from consumer electronics to industrial automation systems. Electronco is generally thought to be innovative in adopting AI technology in that it has been reportedly involved in a number of AI activities throughout the company. MA is a GFES product designed to solve capital budgeting problems, such as new business investments in plants, equipment, products, and so on (see [12] for functional and technical details of MA).

The following description of Electronco's experience with MA is based upon data obtained from unstructured telephone interviews with three MA users at Electronco and two of Palladian Software's customer support staff who worked with Electronco (the names of these interviewees are disguised at their requests). Five people at least tried out MA for their tasks at Electronco, but two of them left the company and we were unable to locate them. The remaining three users have used MA on test projects for several months but eventually stopped using the system, while we could not determine to what extent the other two users used the system.

### Advantage over Spreadsheet Programs

All the MA users at Electronco were in finance areas where they performed such tasks as capital budgeting, investment evaluation, and analysis of business opportunities. Before MA, they had used Lotus 1-2-3 spreadsheet programs for their tasks. When MA was first introduced to them, they were very interested in its advanced analytic capability. John Evanston, a financial analyst, recalled the first impression of MA at the company.

> Most of the people who came to the demonstration viewed it (MA) as a fairly sophisticated model with a lot of bells and whistles, a lot of good theories to take into account. It did something that we had wanted to do for a long time with spreadsheets or special programming.... When submitting a large project (at the company), presentations had to be done beginning with an investment proposal submitted to the capital committee for the division. It then moved to the corporate capital committee, the finance subcommittee of the board, and finally the board. At any point along the way, the proposal needed to be revised to take into account new positions. We felt that MA could simplify this process by quickly incorporating changes.

Mark McCall, a financial manager who initiated the MA project at Electronco, was particularly interested in the Competitors' Impact module, one of the analytic modules of the system.

> What caught my eye was its Competitors' Impact module, which went through a check-list of questions, explained our cost base versus our competitors' and showed the kind of technology or latest market window that might help us to get a product out before the competition.

Susan Johnson, Palladian Software's manager of MA account at Electronco, further described the workings of the Competitors' Impact module.

> The Competitors' Impact module is based on the microeconomics of competition and net present value as the criterion of worth. It gives an estimate of the level to which competition drives price. It also gives an idea of how fast price falls. It does two more things: translate the effect of a revised price forecast into value terms, and help understand the nature of competition in the industry and help probe the limits of the user's ability to forecast prices. It also links a project with industry competition by asking the user a series of questions de-

> signed to accomplish three tasks: estimate industry average cost; explore why the user's long-run price may deviate from the competitive price; and guess competitors' decision criteria.... The user can see the original price forecasts, the industry equilibrium price (i.e., the price that just covers the industry long-run costs), and a suggestion for how price moves from the user's original forecast to the industry equilibrium price.

All the three users agreed that the main reason for their interests in MA was such sophisticated analytic capabilities as the Competitors' Impact module that went well beyond conventional spreadsheet programs. After several demonstrations, Electronco purchased four site licenses of MA with an option to purchase more at a reduced price. At the time Electronco purchased MA, the system cost about $65,000 for the software, plus about $50,000 for the workstation.

**Complexity and Limited Explanation Capability**

The users perceived that MA incorporated very comprehensive knowledge to the application and it could perform complex analyses. John Evanston described:

> MA appeared to have a thorough knowledge, using many complex and interdependent finance and accounting relationships. For example, when I provided profit and loss information, it could convert to cash flows, and vice versa. When I forecasted revenue, cost and investment, it could determine cash taxes and book taxes. In between, it could account for depreciation, working capital, asset recovery, land appreciation, and a host of other relationships. MA could perform that complex set of calculations.

While the users were impressed with the MA's comprehensive knowledge and analytic capabilities, they said that they were not really aware of the operation of the system's composite AI technology base as it worked in the background. Instead, they were more concerned with the knowledge and reasoning used in the system. To help users in this regard, MA provided several explanation features including the Help system, the Road Map, and the Glossary. Susan Johnson of Palladian Software described these features:

> The Help topics contain information specific to the user's place in the system or to the task that the user attempts. For example, in a forecasting table, the Help system displays a list of Help topics pertaining to entering data in the forecasting table. It also provides an explanation of how the entry in a data cell is calculated, e.g., gross revenue.... On the Road Map, the system shows the ways a number is arrived at, by indicating the relevant items for which data exist. For example, when the user has not completed all the steps necessary to proceed to one of the MA's analytic capabilities, it tells the user the type of information the user had to provide to get there.... The system's Glossary provides a list of terms, definitions and examples when appropriate.

But the users felt that these explanation features were not sufficient for understanding the system's reasoning behind a certain result. According to Mark Goldman, a financial analyst:

> Although MA could explain its reasoning, it could not provide sufficient explanations in many cases. I wanted to be able to click a field and get the calculations and any relevant assumptions. It showed some calculations but not all the relevant formula and assumptions I needed.... For example, MA had a module for analyzing the margin for error and the risk associated with investment projects. It could pinpoint those factors critical to the success of the project, such as unit price, unit sales, unit cost, SG&A cost, and investment. It could present the margin for error of each factor. But it could not explain how the critical values were selected. It could not explain what other factors and assumptions were considered in performing the calculations. I needed explanations for the results, as I had to bear responsibility for them.
>
> While Palladian Software's customer support staff made an effort to educate us about the knowledge behind the system, the complex reasoning used in MA required a great deal of expert knowledge. Even few of the customer support staff seemed to understand or be able to explain how all that knowledge worked.... They often had to go

> back and validate the knowledge with their experts.

The users agreed that it was difficult for them to understand the complex knowledge and reasoning behind MA and that the problem was further aggravated by the limited explanation capability of the system.

**Lack of Compatibility and Modifiability**

The users also felt that MA was not flexible enough to describe their problems as they saw them or to accommodate their specific situations, although they liked the comprehensiveness of its generic knowledge. Moreover, the almost closed structure of MA made it very difficult for them to modify the system to address their specific needs. Mark McCall described the frustration felt in this regard.

> What caught us with MA was the lack of flexibility, the lack of ability to tailor it to the way we thought about financial analyses at our company.... For example, we did not name "revenue" at the time. We were calling it "net sales bill." We wanted a flexibility to use our own terminology. In fact, they (Palladian Software) did come up with a way to do that, not initially but eventually. In order to get such flexibility, we had to work with them quite a bit of time.... Another thing was disaggregation of line items, specifically for revenue that was made up of revenues from various products. This was how we looked at it for the business plan. We wanted to do it by entering multiple line items. But MA did not allow us to enter multiple line items but multiple scenarios. To run multiple scenarios in MA, however, we had to recalculate and consolidate data. We worked with them (Palladian Software) on line disaggregation and time disaggregation, but the system never got to the points we desired.
>
> MA was almost a closed system. When we encountered a situation the system could not accommodate, there was generally no way we could modify the system to address the situation.... It was just too rigid in its set-up, and therefore, it would have been next to impossible to fan out as a general management tool, unless we had been willing to change the way we thought about financial analyses.... But there was too much history and culture in the way we thought about things for an outside software package. We could not go against the grain and implement different terminology or methodology. It was not flexible enough to be able to put things in our terms and methodologies.

In fact, MA included the Tailoring module to match the system's generic knowledge to the needs of specific users and problems. But the users felt that it was not sufficient for accommodating all the terms and methodologies that they were using. Lisa Smith, Palladian Software's customer support person, explained some reasons for these limits on the tailoring capability of MA.

> MA is not a simple rule-based expert system. It is a large-scale composite expert system. It utilizes various knowledge representation techniques. MA also uses a host of complex and interdependent relationships in finance and accounting. It cycles through these when appropriate to make sure all the information is consistent. Much of the system's sophisticated analytic capability that users put a high value on is achieved by its complex knowledge base built upon composite AI technology.
>
> While it might be desirable to users, we could not give users the ability to make unrestricted changes in the complex knowledge base.... If the system does not work right after some changes, users would assume that the vendor is at fault. To give users the ability to make unrestricted changes in the system is to give them the ability to corrupt it. Thus, in order to guarantee functional correctness of the complex system, we had to restrict the tailoring capability within manageable bounds....Using the Tailoring module, however, users can employ their ways of thinking about business valuation problems, for example, access authorization, cell formats, financial rates, label names, report names, and so on.... The tailoring should be explicit and well controlled so that we can maintain the integrity of the complex knowledge base and at the same time to allow for the divergence.

But the users felt that the MA's tailoring features were not sufficient for addressing their specific situations. Consequently, using the system required substantial recalculating, consolidating, and reformatting data on the part of the users. John Evanston described this inconvenient and lengthy process that he had to go through:

> The line structure (of MA) was very generic in terms of how certain lines were classified and calculated, for example, revenues, costs, assets, and so forth. But it was not in the way we were used to. So I had to first convert data that we had, so that MA could treat all the lines in the correct manner. Once I ran my scenarios in MA, then I had to translate the results in such a way that our managers could understand and use to make their decisions, that is, in the format that they were used to looking at. That added an enormous amount of cycle time to the process and overwhelmed the benefits that I got from MA in terms of analytic capability, completeness, and the like....It was not so much the fact that MA could not interface to data on our computers. But it was the fact that the way we looked at different costs, assets, and so forth, and the way we did financial analyses and reporting, were not so compatible with the way MA looked at things.

The users agreed that the lack of flexibility and modifiability and task incompatibility were the major problems that hindered their interests in MA and further adoption of the system. In fact, these difficulties overwhelmed the benefits that could be gained from the system, as the users eventually stopped using the system. Given a choice between limited analytic power in a flexible and modifiable system structure, and highly sophisticated analysis in an inflexible and closed system structure, the users chose less sophisticated but more flexible and modifiable spreadsheets and similar analytic programs, e.g., Lotus 1-2-3 and spreadsheet add-on program @RISK.

## LESSONS AND CONCLUSION

We have examined the characteristics and related adoption problems of GFES, focusing on their relative advantage, complexity, and compatibility. These off-the-shelf ES products provide certain advantages over conventional spreadsheets and financial systems. They utilize various AI concepts and methods and embed complex knowledge in their respective financial domains. They intend to apply a common body of generic domain knowledge across a broad range of user problems. Clearly, these characteristics of GFES are mutually dependent on each other, and as such bring about unique benefits as well as problems. The experience of Electronco with Palladian Software's MA shows that the system provides superior analytic capability transcending the limits of conventional spreadsheet programs and it is a major factor for getting potential users to become interested in the system. But it also shows that the system's problems, particularly limited explanation capability and lack of modifiability coupled with knowledge complexity and task incompatibility, prove to be too serious to overcome.

Certainly, many ES have been successfully developed for financial applications. In fact, the finance sector including banking, insurance, investment, and corporate finance has been found as one of the largest application areas of ES technology in a recent survey of ES adopters [14] as well as in an extensive review of literature on ES applications [19]. Built with more sophisticated tools and techniques, they provide many benefits or advantages over conventional financial systems and overcome some of the problems found in this case study. Both the finance and AI communities have become aware of the data necessary to meet the needs of financial ES applications, and the advances of computer technology have solved much of the system incompatibility problem. But it should be noted that many well-publicized ES have proved to be pure hypes or product failures [5]. Also, it is not unusual that a financial ES that has received a considerable press attention disappears entirely because of inadequate performance or proprietary nature [10].

Given the declining trend of ES, the kinds of problems described in this account of Electronco's experience with MA are important for understanding the plausibility of GFES as well as ES in general in the real world. The problem of task incompatibility is largely specific to generic ES, and so it can be avoided by a customized approach. Most ES are now focused on specific tasks and they are generally embedded systems. But they are still liable to such problems as limited explanation capability and lack of modifiability. The ability of an ES to explain its knowledge and reasoning has been assumed as a very attractive feature for users. However, designing an explanation subsystem is getting more difficult as the knowledge and the techniques to represent it become richer. Also, an ES has been thought to be modular, allowing the user to modify the system's knowledge. But it is difficult to modify the knowledge in a composite ES, since only some of the knowledge is represented in rules and much of the rest is represented in other techniques. Even in a simple rule-based system, adding a new rule may require restructuring and re-testing of the knowledge base so that the new rule can operate correctly with the rest of rule set. Taken together, these problems of ES and related design issues are more complex than they appear, but they are well worth trying to understand. We believe the lessons learned from this case study will be useful in developing more viable ES applications. We hope they will prove helpful to those involved in developing and implementing ES, particularly for financial applications.

## ACKNOWLEDGEMENTS

I thank the senior editor, two anonymous reviewers, and Walter Reitman for their comments and suggestions on an earlier draft of this paper. I also thank Richard Ottaway for editorial assistance. Finally, I am grateful to those who participated in the interviews for this case study.

## REFERENCES

[1] Ansari, A. and Modarress, B., "Commercial Use of Expert Systems in the U.S.," *Journal of Systems Management*, Volume 41, Number 12, 1990, pp. 10-13; 32.

[2] Cooper, R. B. and Zmud, R. W., "Information Technology Implementation Research: A Technological Diffusion Approach," *Management Science*, Volume 36, Number 2, 1990, pp. 123-139.

[3] Duda, R. O., Hart, P. E., Reboh, R., Reiter, J. and Risch, T., "Syntel: Using a Functional Language for Financial Risk Assessment," *IEEE Expert*, Volume 2, Number 3, 1987, pp. 18-31.

[4] Firdman, H. E., "Expert Systems: Are You Already Behind?" *Computerworld*, Volume 22, Number 16, April 18, 1988, pp. 99-105.

[5] Gill, T. G., "Early Expert Systems: Where Are They Now," *MIS Quarterly*, March 1995, pp. 51-81.

[6] Hart, P. E., "Financial Expert Systems," *Proceedings of the American Association of Artificial Intelligence*, Volume 2, 1986, p. 1150.

[7] Lamp, E. D., "The Strategic Evolution of a Young Software Company," Masters Thesis, Sloan School of Management, Massachusetts Institute of Technology, 1988.

[8] McKenney, J. L. and Sviokla, J. J., "PlanPower: The Financial Planning Expert System," Harvard Business School Case Number 9-186-293, 1988.

[9] Meyer, M. H. and Curley, K. F., "An Applied Framework for Classifying the Complexity of Knowledge-Based Systems," *MIS Quarterly*, December 1991, pp. 455-472.

[10] Miller, R. M., *Computer-Aided Financial Analysis*, Addison-Wesley, Reading, Massachusetts, 1990.

[11] Radding, A., "Cloning Loan Experts in the Struggle to Limit Risk," *Bank Management*, Volume 67, Number 7, 1991, pp. 48-50.

[12] Reitman, W., "Managing the Development of Generic Expert System Products," in D. A. DeSalvo and J. Liebowitz (eds.), *Managing Artificial Intelligence and Expert Systems*, Yordon, Englewood Cliffs, New Jersey, 1990, pp. 146-169.

[13] Rogers, E., *Diffusion of Innovation*, Fourth Edition, The Free Press, New York, New York, 1995.

[14] Shim, S. J. and Min, B. K., "Organizational Factors Contrasting Adopters and Non-adopters of Expert Systems," *Journal of Information Technology Management*, Volume 7, Number 3&4, 1997, pp. 45-58.

[15] Srinivasan, V. and Kim, Y. H., "Designing Expert Financial Systems: A Case Study of Corporate Credit Management," *Financial Management*, Volume 17, Number 3, 1988, pp. 32-43.

[16] Stansfield, J. L. and Greenfeld, N. R., "PlanPower: A Comprehensive Financial Planner," *IEEE Expert*, Volume 2, Number 3, 1987, pp. 51-60.

[17] Tornatzky, L. and Fleischer, M., *The Processes of Technological Innovation*, Lextington Books, Lexington, Massachusetts, 1990.

[18] Tyran, C. K. and George, J. F., "The Implementation of Expert Systems: A Survey of Successful Implementations," *Data Base*, Volume 24, Number 4, 1993, pp. 5-15.

[19] Wong, B. K. and Monaco, J. A., "A Bibliography of Expert System Applications for Business (1984-1992)," *European Journal of Operations Research*, Volume 85, Number 2, 1995, pp. 416-432.

## AUTHOR'S BIOGRAPHY

***Professor Sung J. Shim*** *is an assistant professor of management information systems in the Silberman College of Business Administration at Fairleigh Dickinson University. He received a Ph.D. in management from Rensselaer Polytechnic Institute and an M.S. in information management from George Washington University. His research interests include expert systems implementation, robotics product strategy, and applications of information systems.*