경영정보학연구
제8권 제2호
1998년 9월

A Classification Structue of Information Systems Failures:

An Empirical Investigation of IS developers' perception

Kim, Jong-Uk*

정보시스템 실패의 구조 규명을 위한 실증연구: 프로그래머를 중심으로

김 종 욱

Many cases of information systems (IS) failure have still continued to be reported ever since computer-based information systems were introduced to process business transactions in the early 1950s. Because an enormous amount of budgets is currently invested on information technology in many organizations, failures and problems of information systems may serve as key culprits to serious business problems which will face the organizations.

Thus, there have been a number of studies on IS failures which aimed to identify causes and reasons for such failures and reveal their inherent nature. Some studies developed conceptual frameworks to classify categories of diverse IS failure phenomena. However, little research performed an empirical study to investigate the underlying structure of IS failures perceived by IS professionals by measuring their perception. In this regard, the current study collected systems developers perceptual data towards IS failure phenomena to identify what constitute IS failure.

The data was analyzed using a multidimensional scaling program and ten categories of problems were identified to constitute the IS failure structure. It was found that most categories were related to problems with users, hardware, and systems quality.

* 영남대학교 상경대학 경영학부 조교수

Ⅰ. Introduction

Since the first business computer system was introduced in the early 1950s to process the payroll transaction, information systems (IS) failures have continued to be reported in the numerous literature [Alter & Ginzberg, 1978; Ewusi-Mensah & Przasnyski, 1991; GAO, 1992; Keider, 1978; Lucas, 1981; Lucas, 1975; Lyytinen & Hirschheim, 1987]. Widely known examples of these phenomena of IS failure are well presented in the two classical research of Brooks' [1974] "the mythical man-month" and Lucas' [1975] "why information systems fail".

Despite quantum improvements in technology and the introduction of new tools and techniques for systems development which in turn have led to improvements in efficiency, functionality, reliability, and ease-of-use, such IS failures are still continuing. Gladden's [1982] survey in 1982 indicates that 75% of all systems development undertaken is either never completed or never used if completed. Mowshowitz [1976] states that many, if not most, information systems are failures in one sense or another. These IS failures are continuing in the 1990s. Abdel-Hamid and Madnick[1990] cite a headline in the Wall Street Journal explicitly saying "Creating New Software was Agonizing Task for Mitch Kapor Firm", subtitled with "Despite Expert's Experience, Job Repeatedly Overrun Time and Cost Forecasts". Ewusi-Mensah and Przasnyski [1991] also identify multidimensional factors for IS project abandonment such as cost overruns, schedule delays, technological problems, and organizational, behavioral, or political issues.

These IS failures are manifested in a number of ways. In case of discretionary users of the system, they may not use the system or may use it infrequently [Lucas, 1975]; captive users, on the other hand, may resist the mandatory use of the IS [Frantz & Kahn, 1994; Markus, 1983] and may exhibit negative attitudes towards the system [Ginzberg, 1981]. Thus, in any case, the potential benefits of the IS may not be realized from the organization's perspective [Alter & Ginzberg, 1978].

IS failure can be fatal to the success of organizations due to the current considerable investments in information systems. During the 1980s, investments in information technology (IT) reached $ 190 billion [Keen, 1991], and US expenditures for software development and maintenance has been estimated to grow by 1995 to more than $ 225 billion domestically and more than $ 450 billion worldwide [Alter & Ginzberg, 1978]. For the remainder of 1990s, companies will continue to spend an enormous capital on building their information systems.

In this regard, it is important to understand in a clear and comprehensive way the concept of IS failures and their antecedent causes. Therefore, failures in IS development and implementation have been a subject of considerable discussion in the information systems literature [Bostrom & Heinen, 1977; Ewusi-Mensah & Przasnyski, 1991; Ginzberg, 1981; Kumar & Welke, 1990]. The most prominent work for IS failures is provided by Lyytinen and Hirschheim [1987]. They developed the most comprehensive classification framework for IS failures, identifying four ways of defining IS failures: correspondence failure, process failure, interaction failure, and expectation failure.

Although a wealthy stream of research has existed on this issue [Ewusi-Mensah & Przasnyski, 1991; GAO, 1992; Ginzberg, 1981; Lucas, 1975; Lyytinen & Hirschheim, 1987; Markus, 1981; Robey & Markus, 1984], however, the concept of IS failure still remain ill-defined and poorly understood. A lot of previous research identified only one or two or, at most, several categories of IS failures from the individual cases of such failures. Some studies, although they recognized multiperspectives of IS failures, also attached too limiting causes to the IS failures. Therefore, we need to clarify the dimensions of IS failure in a more comprehensive manner to define and well understand what IS failure really is. Thus, in order to identify its dimensions, we ask the following question: What constitutes IS failures as developers "perceive" them. The answer to this question may serve as a sound base to identifying valid causes or reasons of IS failures, and developing ways to prevent such previous fatal IS failures. What has not been forthcoming is an empirical investigation, not a conceptual work, of the IS failure categories which developers conceive [Lyytinen & Hirschheim, 1987]. Therefore, the objective of this research is to examine empirically the classification of IS failures by identifying the underlying structure of IS failures as developers conceive them. IS failures may originate from diverse sources of the systems life cycle such as planning, development, operations, maintenance, etc., and thus, this study aims to investigate these multiperspectives of the IS failure phenomena.

Ⅱ. Previous Research

IS failure has been a subject of considerable discussion among many IS researchers. They investigated IS failures from diverse perspectives. A number of behavioral and implementation researchers suggest that lack of attention to socio-organizational issues may often be the cause of these IS disasters [Bostrom & Heinen, 1977; Friedman & Kahn, 1994; Levine & Rossmore, 1993; Lucas, 1975; Lyytinen & Hirschheim, 1987; Robey & Markus, 1984]. These studies have identified the following social and behavioral issues as antecedents to IS failure: user resistance [Frantz & Robey, 1984; Markus, 1983], implementation problems [Lucas, 1975; Lyytinen & Hirschheim, 1987], lack of user involvement [Ives & Olson, 1984], negative user attitudes and inadequate expectations [Ginzberg, 1981; Lyytinen & Hirschheim, 1987], and conflicts between diverse stakeholders [Markus, 1983]. Such behavioral reasons for IS failures suggest that, in order to provide socially acceptable as well as functionally complete systems, socio-organizational issues should be given the attention during systems development.

On the other hand, the lack of technical completeness and efficiency is also considered a key culprit to the failure to realize the benefits of the IS. From the technical perspecitve, such critical conditions as functional capability, high system reliability, and ease of use should be achieved first in order to develop successful systems. A wide variety of examples of systems failures which originated from the technical limitations and incompleteness are provided in the issues of the ACM SIGSOFT

Software Engineering Notes.

There is a viewpoint for IS failure from both the technical and social perspectives. Bostrom and Heinen [1977] suggested that neither techno-economic nor socio-organizational considerations can be ignored during systems development. They argue that technical and social aspects of IS development should be no longer seen as competing aspects in system development. Rather, it should be recognized that one cannot be complete without the other. Thus, systems analysts, in order to achieve the potential benefits of the IS and successfully accomplish its objectives, must possess a balanced view of the social and organizational, as well as technical aspects of information systems.

From the project management viewpoint, Genuchten [1991] investigated the reasons for schedule delays in software development and led to a conclusion that the distribution of reasons for delay varied widely from one department to another. And he recommended that the responsible department should reveal its reasons for delay in order to be able to take adequate actions for improvement. Abdel-Hamid and Madnick [1990] warned that, from a case study of a NASA project, because intuition alone is not sufficient to handle complex and dynamic interactions in the software project, managers could mislead to the wrong lesson. Thus, an effective postmortem diagnostic exercise to identify problems and their causes is essential in adequately identifying project deficiencies and thereby preventing repeated occurrences of the same errors on future projects. Thus, they argue that "the payoff from an effective postmortem is a smarter organization that truly learns from its failures(p. 47)".

Lyytinen and Hirschheim [1987] classified IS failures into four major categories. First, they identify three traditional classes of IS failures. The first category is "correspondence failure", which defines IS failures when original system design objectives are not met. This perspective focuses on system quality and performance in a technical sense to judge whether the system failed or not. However, as they point out, the concept of "correspondence failure" is often too idealistic and the system could end in a failure although the system achieves all the initial objectives if the original requirements are ambiguous.

The second category of defining IS failure is "process failure", which views failure as the inability to produce a workable system, or the ability to produce a workable system but with cost overruns and schedule delays. The third approach of failure is labeled as "interaction failure", which is originated largely from the perspective of the user. It is generally characterized as a low level of IS use, negative attitudes of users towards IS, and low user satisfaction.

The last view of IS failure is "expectation failure". The success or failure of the system depends on the user's expectations, i.e., the beliefs and desires concerning how the IS will serve the group's interests. So, they define the expectation failure as the "inability of an IS to meet a specific stakeholder group's expectations". However, even this expectation failure has a limitation because there may be a conflicting situation which are not unambiguously defined by the concept of expectation failure. While one group of stakeholders benefit from the introduction of a new IS, the other groups of stakeholders may be disadvantageous, if they lose organizational power due to the new IS, for instance.

However, none of IS failure research provide sufficient views of IS failures covering the full range of IS failures. Although some studies recognized that IS failure consists of multiple perspectives, it is not clearly known that they provided a comprehensive view of IS failures. Furthermore, little empirical research has been performed to identify the underlying structure of IS failure by measuring a developer's perception. Thus, the objective of this research is to examine a structure or categories of IS failures as developers perceive them. The results of this research will provide the classification structure of the IS failure concept and an empirical foundation contributing to the conceptualization and measurement of IS failure in future studies.

Ⅲ. Research Methods

Data for the analysis was gathered from the sorting task where the subjects sorted IS failure factors based on the perceived similarity. The list of the total 35 factors was created by the author from the previous IS success and failure literature [Bailey & Pearson, 1983; DeLone & McLean, 1992; Ginzberg, 1981], and the results of the sorting task was analyzed with the multidimensional scaling (MDS) method. The 35 factors used in the sorting task is listed in <Table 1>.

3.1 Participants

Participants in this study were 30 programmers or systems analysts in the IS departments of a large bank and a department store located in the Southeast region of Korea. Their average age was 32.3 and their average number of years in systems development was 3.6 years. Their average experience of 3.6 years served in systems development seemed reasonable for the purpose of this study because most of them already performed systems development projects and, thus, they could understand meanings of the factors used in the sorting task.

3.2 Procedures and Sorting Task

Participants were individually contacted by the researcher and asked to participate in the study. When they agreed to participate, then they were instructed about the nature of the study and how to perform the sorting task.

A randomly ordered deck of 35 cards was given to each subject for the sorting task. Each card was typed with one of 35 factors which were believed to be largely related to IS failures, with its number on the top. The participants were provided with adequate space to perform the sorting task and they were instructed to divide the deck of cards, first of all, into two piles-one containing the factors that they thought were closely related to IS failure and the other they thought relatively irrelevant to IS failure. Then, they were told to make as many sub-piles as they thought appropriate based on similarity.

Finally, the participants were asked to rank order those sub-piles in terms of relevance to IS failures. It took about 20 minutes in completing the sorting task.

<Table 1> IS Failure Factors Used in the Sorting Task

Schedule delays
Lack of system reliability
Slow computer processing speed
Failure of backup and disaster recovery of the system
Cost overruns(Insufficient budgets)
Incompatibility with other interfacing systems
Insufficient user communication
Insufficient hardware capacity
Failure to secure information
Conflicting relationships between the user and IS departments
Conflicts/agreement of objectives between the user and IS departments
Wrong development plan
Insufficient user's knowledge to provide right user requirements
Inaccurate output information
Lack of data integrity
System that is difficult to modify
Lack of vendor support for hardware and software
Insufficient developer experience and skill(Technical competence of IS staffs)
Poor design of report formats and screens
Non-use of development tools and aids
User's unreasonably high expectation towards the system
Delayed response to user requests by IS department
Failure to adopt and learn new information technology
Insufficient training and education of end-users
Selection of inadequate hardware - I/O device, POS, disk, network device
Lack of system documentation and user manuals
Delayed provision of output information
Selection of inadequate software - database, programming languages
Program errors
Insufficient top-management support for IS
User dissatisfaction towards the IS
Failure to achieve expected results
Failure to realize the initial user requirements
User's refusal to use the system
Insignificant increase of user productivity

The use of sorting data in identifying the underlying structure of IS failure was borrowed from psychological research originating from Bruner, Goodnow, and Austin [1956]. As Walsh [1988] indicates, the sorting procedure has some strengths. Subjects are able to categorize a set of factors which is independent of the experimenter's own category system and the method requires a much smaller number of comparison for estimating psychological distance between objects than Torgerson's [1958] method which requires subjects to compare all pairwise similarities in a factor set.

Ⅳ. Results

4.1 Multidimensional Scaling (MDS)

To analyze the data in identifying the underlying dimensions of IS failure as perceived by systems developers, a multidimensional scaling (MDS) program was employed. A modified version of multidimensional scaling, the MDSORT program [Takane, 1981], was used to analyze the result data of the sorting task. MDS generally "allows the investigator to examine and evaluate the underlying dimensions or criteria that people use in formulating perceptions about the similarities among products and services" [Elliot & El Sheshai, 1979]. Because MDS recovers the underlying structures or dimensions of products or services which are hidden among consumers, it has been widely used in marketing analysis. The program identifies the underlying structure of IS failures as developers perceive by analyzing the data gathered from the 30 programmers and analysts' sorting tasks. Takane [1981] specifically indicates that it provides significant dimensions of a concept such as IS failure by developing "a configuration of stimulus points in a multidimensional Euclidean space in such a way that the sum of squared intercluster distances averaged over subjects is a maximum under suitable normalization restrictions on the configuration(p. 698)".

4.2 The Dimensions of IS Failures

The first task analyzing output was to determine how many dimensions best fit the data. From a scree test of eigenvalues, the five dimensional solution was found best fitting data. This five dimensional solution is known as a group trait space [Walsh, 1988], which represents "a common set of dimensions along which individuals are thought to perceive stimuli(p. 881)". The dimension also represents the aggregate knowledge structure regarding the similarity of the 35 factors for the 30 systems developers.

Table 2 shows the five dimensional MDSORT solution including loadings for each dimension. The total variance explained by these five dimensions was 26.4%. The variance explained for each underlying dimension ranged from 6.3% for dimension one to 4.1% for dimension five. The results of the MDSORT program provide both positive and negative polarities for each dimension since clusters of stimuli anchor the ends of each dimension. The title of clusters for each dimension was named by the author according to the factors loaded. Thus, for instance, the negative polarity for dimension one was titled as "user dissatisfaction" following such factors loaded as 'failure to achieve expected results', 'users dissatisfaction toward the IS', etc.

The MDSORT results in <Table 2> show that factors related each other are well grouped together at each dimension. For example, factors loaded on the negative polarity of dimension two are associated with problems of inadequate system design and reliability and thus the group was named as "poor system quality". <Table 3> summarizes the ten underlying categories of IS failures identified from the analysis of the sorting data.

V. Discussion

The primary goal of this study was to identify empirically the underlying dimensions of IS failure as perceived by systems practitioners. The findings suggested directions for identifying the structure of IS failure through analyzing the empirical

<Table 2> The Dimensions of IS Failure

Dimensions	Positive Polarity: Factors in the Dimension	Loadings	Negative Polarity: Factors in the Dimension	Loadings
One	**Incompatibility**		**User Dissatisfaction**	
	Incompatibility with other interfacing systems	0.368	Failure to achieve expected results (monetary savings, etc.)	-0.363
	Lack of system reliability	0.207	User dissatisfaction towards the IS	-0.277
	Selection of inadequate software - database, programming languages	0.205	Conflicts/agreement of objectives between the user and IS departments	-0.262
			Insufficient top-management support for IS	-0.244
Two	**Lack of New Technology and Tools**		**Poor System Quality**	
	Non-use of development tools and aids	0.406	Program errors	-0.298
	Failure to adopt and learn new information technology	0.322	Poor design of report formats and screens	-0.292
			Insufficient hardware capacity	-0.254
			Lack of system reliability	-0.240
Three	**User Resistance**		**Operation Problems**	
	User's refusal to use the system	0.271	Failure to backup and recover disaster of the system	-0.375
	Insufficient user communication	0.163	Non-use of development tools and aids	-0.354
	Program errors	0.158	Selection of inadequate hardware - I/O device, POS, disk, network device	-0.287
	Insignificant increase of user productivity	0154	Failure to achieve expected results (monetary savings, etc.)	-0.285
			System that is difficult to modify	-0.281
Four	**User Problems**		**Cost/Hardware Problems**	
	Insufficient user's knowledge to provide right user requirements	0.306	Cost overruns(Insufficient budgets)	-0.410
			Lack of vendor support for hardware and software	-0.252
	Inaccurate output information	0.271	Lack of system documentation and user manuals	0.241
	User dissatisfaction towards the IS	0.251	Failure to achieve expected results	0.209
Five	**Irrlesponsiveness to Users**		**Hardware Problems**	
	Non-use of development tools and aids	0.320	Insufficient hardware capacity	-0.301
	Delayed response to user requests by IS department	0.305	Slow computer processing speed	-0.297
			Lack of vendor support for hardware and software	-0.260
	Failure to realize the initial user requirements	0.280		
	Selection of inadequate software - database, programming languages	0.277		
	Program errors	0.267		

<Table 3> The Underlying Categories of IS Failure

Incompatibility	User dissatisfaction
Lack of new technology and tools	Poor system quality
User resistance	Operation problems
User problems	Cost/Hardware problems
Irresponsiveness to users	Hardware problems

results, and provided a guideline for conceptualizing IS failure for future studies in a manner consistent with what practitioners perceived.

There were found five MDS dimensions most appropriate in explaining the structure of IS failure and problems. However, because the MDS algorithm provides two different clusters of concepts at the each end of the individual dimension, i.e., positive and negative polarities, the ten categories were found to form the classification structure of IS failure.

As listed in <Table 3> we found that such categories as user dissatisfaction and resistance, poor system quality, hardware problems, lack of up-to-date technology/development tools formulated the structure of IS failure. In summary, the ten categories labeled largely express problems with users, hardware, system quality, and use of new technology.

It was interesting, among four types of IS failure defined by Lyytinen and Hirschheim [1987], process failure was not explicitly defined from our empirical results. In other words, problems of project management problems such as budget overruns and schedule delays were not explicitly perceived as the failures by subjects. A possible explanation for this is that most subjects who participated in the sorting task were programmers or low-level systems analysts who were not much concerned about project management decisions because those decisions are usually made by the upper-level managers.

It was also interesting to find that the results of the MDSORT data indicated that concerns of developers were largely limited to hardware, system quality, and problems with users. They did not consider them as a serious failure not to satisfy the initial user requirements and not to align organizational and IS objectives. This result suggests that the classification structure of IS failure be diverse depending on the developer's work experiences or other backgrounds. Considering about two-thirds of the subjects participated in this study were programmers, the limited perception of IS failures on users, hardware, systems quality seems to originate from their limited work experiences - programming related backgrounds. Major tasks of the most subjects were programming, system design jobs, or communications with users to solve users' dissatisfaction towards the system while they operated the system. Furthermore, because participants in this study were chosen from a bank and a department store, the type of systems which most participants had experiences with in the past were usually transaction processing systems.

In this regard, the structure of IS failure possessed by IS managers may be different from that possessed by programmers or low-level systems analysts. Furthermore, the underlying dimensions of IS failure defined by end users may be also different from those perceived by systems developers.

⟨REFERENCES⟩

[1] Abdel-Hamid, T. & Madnick, S., "The Elusive Silver Lining: How we fail to learn from software development failures," *Sloan Management Review*, Vol. 31, 1990, pp. 39-48.

[2] Alter, S., & Ginzberg, M., "Managing Uncertainty in MIS Implementation," *Sloan Management Review*, Vol. 19, 1978, pp. 23-31.

[3] Bailey, James and Sammy Pearson, "Development of a Tool for Measuring and Analyzing Computer User Satisfaction," *Management Science*, Vol. 29, No. 5, May 1983, pp. 530-545.

[4] Bostrom, R.P., & Heinen, J.S., "MIS Problems and Failure: A Sociotechnical Perspective, Part I: The causes," *MIS Quarterly*, Vol. 1, No. 3, 1977, pp. 17-32.

[5] Brooks, F. *The Mythical Man- Month.* Reading, Mass.: Addison-Wesley, 1974.

[6] Bruner, J., Goodnow, J., & Austin, G., *A Study of Thinking*, New York: John Wiley & Sons, 1956.

[7] Delone, W.H. and E.R. McLean, "Information Systems Success: the Quest for the Dependent Variable," *ISR*, Vol. 3, No. 1, March 1992, pp. 60-95.

[8] Elliott, M. & El Sheshai, K., *Introduction to Multidimensional Scaling and Conjoint Measurement for Marketing Analysis*. Georgia State University, Atlanta, Georgia, 1979.

[9] Ewusi-Mensah, K & Przasnyski, Z, "On Information Systems Project Abandonment: An exploratory study of organizational practices," *MIS Quarterly*, Vol. 15, 1991, pp. 67-86.

[10] Frantz, C. R., & Robey, D., "An Investigation of User-led System Design: Rational and political perspectives," *Communications of the ACM*, Vol. 27, No. 12, 1984, pp. 1202-1209.

[11] Friedman, B., & Kahn, P., "Educating Computer Scientists: Linking the social and the technical," *Communications of the ACM*, Vol. 37, No. 1, 1994, pp. 65-70.

[12] GAO(United States General Accounting Office), *Information Resources: Summary of federal agencies' information resources management problems*, GAO/IMTEC -92-13FS, 1992.

[13] Genuchten, Michiel van, "Why is Software Late? An empirical study of reasons for delay in software development," *IEEE Transactions on Software Engineering*, Vol. 17, No. 6, 1991, pp. 582-590.

[14] Ginzberg, M., "Early Diagnosis of MIS Implementation Failure," *Management Science*, Vol. 28, 1981, pp. 459-478.

[15] Gladden, G., "Stop the Life-cycle, I Want to Get Off," *Software Engineering Notes*,

Vol. 7, No. 2, 1982, pp. 35-39.

[16] Ives, B., & Olson, M.H., "User Involvement in Information Systems Development: A Review of Research," *Management Science*, Vol. 30, No. 5, 1984, pp. 586-603.

[17] Keen, P.G.W., *Shaping the Future: Business design through information technology*, Boston: Harvard Business School Press, 1991.

[18] Keider, S., "Why MIS Projects Fail," *Datamation*, December 1978.

[19] Kumar, K, and Welke, R.J. Methodology Engineering: A method for situation specific methodology construction. In W.W. Cotterman and J.A. Senn (Eds.), *Systems analysis and design: A research agenda*, Chichester: John Wiley & Sons, 1990.

[20] Levine, H., and Rossmore, D., "Diagnosing the Human Threats to Information Technology Implementation: A missing factor in systems analysis illustrated in a case study," *Journal of Management Information Systems*, Vol. 10, No. 12, 1993, pp. 55-73.

[21] Lucas, H.C., *Implementation: The Key to Successful Information Systems*, New York: Columbia University, 1981.

[22] Lucas, H.C., *Why Information Systems Fail*, New York: Columbia University, 1975.

[23] Lyytinen, K. & Hirschheim, R., "Information Systems Failure: A survey and classification of the empirical literature," *Oxford Surveys in Information Technology*, Vol. 4, 1987, pp. 257-309.

[24] Markus, M.L., "Power, Politics, and MIS Implementation," *Communications of the ACM*, Vol. 26, No. 6, 1983, pp. 430-444.

[25] Markus, M.L., "Implementation Politics: Top management support and user involvement," *Systems, Objectives, and Solutions*, Vol. 1, No. 4, 1981, pp. 203-215.

[26] Mowshowitz, A., *Information Processing in Human Affairs*. Reading, MA: Addison-Wesley, 1976.

[27] Robey, D., and Markus, M. L., "Rituals in Systems Design," *MIS Quarterly*, Vol. 8, No. 1, 1984, pp. 5-15.

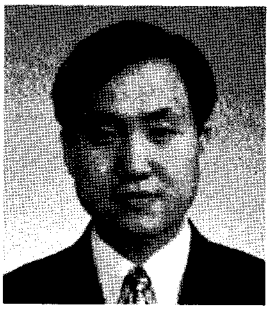
[28] Takane, Y., "MDSORT: A special purpose multidimensional scaling program for sorting data," *Behavior Research Methods & Instrumentation*, Vol. 13, 1981, 698.

[29] Torgerson, W., *Theory and Methods of Scaling*, New York, Wiley, 1958.

[30] Walsh, J., "Selectivity and selective perception: An investigation of manager's belief: Structures and information processing," *Academy of Management Journal*, Vol. 31, No. 4, 1988, pp. 873-896.

◆ 이 논문은 1998년 3월 26일 접수하여 1차 수정을 거쳐 1998년 9월 15일 게재확정되었습니다.

◆ 저자소개 ◆



김 종 욱 (Kim, Jong-Uk)

서울대학교 산업공학과(1982)를 졸업하고, Georgia State University에서 경영정보학 박사학위(1994)를 취득한 뒤, 현재 영남대학교 경영학부에 재직중이다. 쌍용정보통신, BASF Korea Ltd., Clark Atlanta University에서 근무했고, 관심분야는 interorganizational systems, electronic commerce, 정보시스템 평가 등이다.