IEMS Vol. 4, No. 2, pp. 207-217, December 2005.

# A Framework for Analysis of Systems Failure in Information Systems Integration

**HanGook Kim†**
Department of Industrial Engineering and Management
Tokyo Institute of Technology, Tokyo 152-8550, JAPAN
Tel: +81-3-5734-2587, Fax: +81-3-5734-2947, E-mail: korea712@me.titech.ac.jp

**Junichi Iijima**
Department of Industrial Engineering and Management
Tokyo Institute of Technology, Tokyo 152-8550, JAPAN
Tel: +81-3-5734-3942, Fax: +81-3-5734-2947, E-mail: iijima@me.titech.ac.jp

**Sho Ho**
Department of Industrial Engineering and Management
Tokyo Institute of Technology, Tokyo 152-8550, JAPAN
Tel: +81-3-5734-2361, Fax: +81-3-5734-2947, E-mail: s-hou@ml.me.titech.ac.jp

**Abstract.** Business mergers are a direct result of rapid changes in the current corporate environment. They are occurring in many industries, including financial institutions. As information systems prove increasingly indispensable in the integration of companies' business systems, information system integration is becoming increasingly necessary. However, in many cases such integration does not work well. Therefore, this paper proposes a new framework using both IS integration model and IS integration phases to analyze systems failure in IS integration.

**Keywords:** System Failure, Information System Integration, Failure Analysis Framework

## 1. INTRODUCTION

In today's dynamic business environment, business system integration has become necessary in various types of industries, including financial institutions. Such efforts in information system integration, unfortunately, are often unsuccessful. Integration involves more than simply joining computer applications to share data. It includes issues at the process level in areas such as (a) integration of processes across departments, (b) job specifications with system capabilities, and (c) manual processes with computerized processes; and at the business goals level in areas such as (a) business architecture, (b) values, and (c) performance management.

Coping with complex IT projects continues to be a central problem for information systems (Keider, 1984). While information technology has become an integral part of organizations, the failure rate of IT projects remains high (Cole, 1995). A project is usually deemed successful if it (a) meets requirements of functionality, reliability, maintainability, portability, efficiency, integration, and operability; (b) is delivered on time; and (c) stays within budget (Powell, 1996). Citing a 1995 Standish survey, May reported that only one-sixth of all projects were completed on time and within budget; one-third of all projects were cancelled; and over half were considered challenged (May, 1998).

In this paper, we propose a model with multiple components in order to explain more clearly IS integration. Moreover, we propose a framework that uses components of the IS integration model as horizontal axis, and phases of IS integration as vertical axis. This framework serves to analyze both failure factors and the relationships according to the IS integration model.

The structure of this paper is as follows. In section 2, we discuss conventional research on the causes of failure of information systems. In section 3, we propose a new model of systems integration and examine the processes by which obstacles to a successful system occur. Furthermore, a framework for analyzing obstacles

† : Corresponding Author

in systems integration is proposed using the phases of IS integration and the components involved. In section 4, we analyze two actual examples using the proposed analytical framework, and we consider what factors can lead to the aforementioned obstacles. Finally, the results of this research and direction for further work are presented.

## 2. IS INTEGRATION AND FAILURE

### 2.1 Definition of IS Failure

It is a general consensus among researchers that failure rates of IS are high. However, the many studies and surveys in this field present few clear definitions of failure, beyond subjective or personal interpretation. Since the 1970s, a number of frameworks have been proposed to explain the concept of IS failure (Lucas, 1975; Sauer, 1993). Two recent approaches seem particularly important because they relate IS failure to social and organizational context: (a) the concept of *expectation failure* by Lyytinen and Hirschheim, and (b) the concept of *termination failure* by Sauer.

Lyytinen and Hirschheim (1987) identified four major theoretical categories in IS failure:

1) Correspondence failure

This form of IS failure, most commonly discussed in the literature, typically reflects a management perspective on failure. Design objectives are specified in detail, followed by an evaluation of the information system in terms of these objectives. If a lack of correspondence exists between objectives and evaluation, the IS is regarded as a failure.

2) Process failure

This failure is characterized by unsatisfactory development performance. Either the IS development process cannot produce a workable system, or the development process produces a workable IS, but the project runs over budget.

3) Interaction failure

Here, the emphasis shifts from either a mismatch of requirements and system or poor development performance to usage of a system. The argument is that a heavily used system constitutes success. But it is, in actuality, a failure if it is rarely used, or if significant problems affect its use.

4) Expectation failure

Lyytinen and Hirschheim describe expectation failure as a superset of the three types of failure described above. They also regard expectation failure as a more encompassing, political, and pluralistically informed view of IS failure than the other forms. This is because they characterize correspondence, process, and interaction failure as having one major theme in common. Each views an IS largely as a neutral technical artifact (Klein and Hirschheim, 1987). In contrast, however, they define expectation failure as the inability of an IS to meet a specific stakeholder group's expectations. IS failures signify a gap between an existing situation and a desired situation for members of a particular stakeholder group. Stakeholders are any group of people who share a pool of values that define both the desirable features of an IS and the methods by which they should be obtained.

Lyytinen (1988) broadens this analysis by making a useful distinction between *development failure and use failure*. Stakeholder groups face problems in IS in terms of either development or use. In the former case, the main concern is molding the IS to fit the stakeholders' future interests, while the focus is on aligning the IS with the stakeholders' ongoing concerns. In terms of development failure, Lyytinen (1988) lists categories of common problems: (a) goals, (b) technology, (c) economy, (d) organizational impact, (e) participation, (f) control of development, and (g) perception of development. In terms of use failures, the following categories are of significance: (a) technical material, (b) data, and (c) conceptual, reactionary, and complexity problems.

Sauer (1993) has criticized the model proposed by Lyytinen and Hirschheim for its plurality. Sauer's model proposes a more conservative definition of IS failure. According to his account, an IS should be deemed a failure only when development or operation stops, thereby leaving supporters dissatisfied with the systems ability to serve their interests. Thus, a system should not be considered a failure until all interests in the progression of an IS project have ceased. This definition of termination failure is hence stricter than Lyytinen and Hirschheim's concept of expectation failure.

Sauer develops a model of IS failure based on exchange relations. He describes the development of information systems as an innovative process based on three components: the project organization, the information system, and its supporters. Each component is arranged to form a triangle of dependencies working within the context of an environment. The information system depends on the project organization, which depends on its supporters, who, in turn, depend on the information system. The information system requires the efforts and expertise of the project organization for supporters, and the project organization is heavily dependent on the provision of supporters in the form of material resources and help in coping with contingencies. Finally, supporters require benefits from the IS. Figure 1 illustrates this triangle of dependencies.

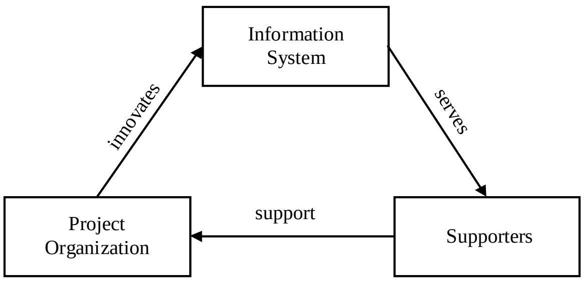


**Figure 1**. Sauer's model of IS development

Sauer (1993) distinguishes termination failure from expectation failure by his concept of flaw. Information systems are the products of processes open to flaws, and every information system is flawed in some way. However, flaws are different from failures, as flaws may be either corrected or accepted within any innovation process, albeit at a cost. Flaws are defined by the perception of stakeholders. Flaws may take the form of program bugs, hardware performance, or organizational changes. Without support available to deal with flaws, the capacity of an IS to serve its supporters will decrease, allowing further flaws to be introduced into the innovation process. At some point, the volume of flaws may trigger the decision to remove support and, hence, to terminate the project.

Ewusi-Mensah and Przasnyski (1994) distinguish between *IS project abandonment* and *IS failure*. They perceive IS failure as failure of usage and/or operation of the IS, whereas IS project abandonment involves the process of IS development. This approach is similar to Lyttinnen's distinction between development and use failure. Ewusi-Mensah and Przasnyski (1995) delineate three major types of project abandonment:

(1) Total abandonment: Complete termination of all activities on a project prior to full implementation.
(2) Substantial abandonment: Major truncation or simplification of a project to make it substantially different from the original specification prior to full implementation.
(3) Partial abandonment: Reduction of the original scope of the project without significant changes to the original specification of IS prior to full implementation.

Their small survey study suggested that total abandonment is the most common type of development failure experienced in the U.S. They also found that organizational factors, particularly (a) the level of senior management involvement and (b) the degree of end-user participation in the project development, were the most widespread and dominant factors contributing to IS success or failure.

However, there were few or no papers about IS integration. In this paper, we focus on the issue of failure as it relates to IS integration. We maintain that the tendency to analyze IS failure solely from a technological standpoint is limiting, that the nature of the failure of IS integration is multi-faceted, and, consequently, must be analyzed in terms of components related to the project of IS integration.

## 2.2 Phases of IS integration

Although many procedures are used in information system development, the following three phases are common to all procedures (Nanjo and Hosaka, 1989) :

(1) *Plan*: Involves (a) deciding on construction and introduction, (b) establishing construction / introduction organization, (c) performing investigation analysis, and (d) completing outline design.
(2) Implementation: Involves concrete promotion of construction and introduction and selection of (a) application business, (b) a detailed design, and (c) program creation.
(3) Evaluation: Involves evaluating and improving on the information system's reliability, response, stability, adaptability, and economical efficiency.

When IS integration accompanies merging operations such as M&A, it is necessary to consider the integration analysis and design phases, though they are not seen in ordinary information system construction phases. Therefore, in this paper we divide the plan phase into two steps — an analysis phase and a design phase based on Yamamoto's work (2002).

1) Integration analysis

The integration analysis phase involves analysis of the influence of an IS on M&A, and the execution possibility of IS integration. Moreover, it utilizes a long-term view in deciding on the business integration strategy that can be performed. Business process and risk are analyzed, and a core function is clarified. It is necessary to (a) establish the range, especially regarding the field to be unified; (b) set up a process; (c) construct a system; and (d) determine priorities. Moreover, it is important to compose and examine a special professional team specifically for integration.

2) Integration design

In the integration design phase the base system after integration is chosen, and business processes and data models are designed. Moreover, it is important to set up a valuation basis as a design condition. The execution organization and the schedule of integration must be decided in advance of implementation. The influence on business, from the viewpoint of customers and employees, must be analyzed. It is necessary to clarify the business process as packages and to review the validity of design.

3) Integration implementation

In the integration implementation phase, required software is developed based on an integration design; data shift, application integration, network integration, and process integration are carried out; and the existing system is connected. Its influence on the whole M&A is large when integration is ineffective in this phase. Thus, it is of high importance to both build risk-management organization and aim for sufficient information sharing.

4) Integration evaluation

It is necessary to supervise the integration situation and to measure the effect of integration in the integration evaluation phase. Moreover, quality assessment of an integrative system is required. It is essential to prepare for system integration to the new M&A by continuing management of both risk and problem examples through operation of an integrated system.

These phases are used as a time division of the framework we propose in 3.3.

### 2.3 Four Options for IS Integration

With advantages and disadvantages for each, four main options in IS integration are as follows:

(1) Develop interface between both systems: Keeping both systems has the advantage of continuity, with little cost required for retraining people in the new system. The main problem with this approach is the duplication of support staff, meaning no reduction in cost.

(2) Develop a new system: This appears to be an attractive option, as one exceptional new system can replace two old systems. The clear advantage is that this option suits the long-term future of the organization. However, to enable the new system to take off, all developmental work on the existing systems must cease, allowing the new system to take over. The merged company will be forced to operate as two separate companies for a considerable time, delaying the benefits of the merger. If neither existing system is good, then building a new one is worthwhile.

(3) *Cherry-Picking* (select the best system): Each part of a merged organization has some state-of-the-art systems that they believe can be merged with those from the other side to make a more desirable system. This approach is technically flawed. Each organization, over a number of years, has produced a number of systems that operate together. However, it is nearly impossible for them to work with another organization's systems, mainly because of their overlapping functionality.

(4) *One-sided* (select one company's system): The last major option is to select one set of systems for use throughout the new merged company. The advantage here is that a guaranteed system is already in place, so all that has to be done is to (a) ensure adequate system growth to deal with increased business, (b) implement the change, and (c) train the staff from the new organization to use it. However, during the transfer to one system, little enhancement can take place. If a large firm is taking over a much smaller one, then selecting the larger company's system seems the best choice.

The best option depends on the present situation. Therefore, it is necessary to understand the features of each option in order to select the most appropriate one and perform a successful integration.

## 3. ANALYSIS OF IS INTEGRATION FAILURE

### 3.1 IS Integration Model

Boehm and Ross (1989) point out that different stakeholders in a software development project have unique objectives that often conflict with one another. For instance, users require a robust, user-friendly system with functions that can support their tasks. At the same time, development team members hope to encounter interesting technical challenges. These differing expectations create fundamental conflicts when simultaneously approached, resulting in an unclear or misunderstood scope of the project. In light of these difficulties, we propose a new model (Figure 2) for analyzing obstacles in system integration, using Sauer's model as the base. The relationships between the components directly involved in IS integration are more clearly expressed by adding the *Integration Promotion Committee* (IPC), and by separating *users* from *supporters*.

IPC consists of representatives of supporters, project organizations, and users. Figure 2 shows that IPC is a superordinate group to these components (supporters, project organizations, and users). Their opinions, claims, and conflicts can be managed and adjusted by IPC that has the ultimate responsibility of the project. The IPC develops detailed integration plans toward the system integration, in conjunction with other components.

It is users who carry out their business with the information system. Users are an important element of the environment of IS. Ives and Olson (1984) review the literature on IS success through user involvement. They categorize IS failure primarily as organizational behavior problems and emphasize that user involvement is essential for avoiding a failed system.

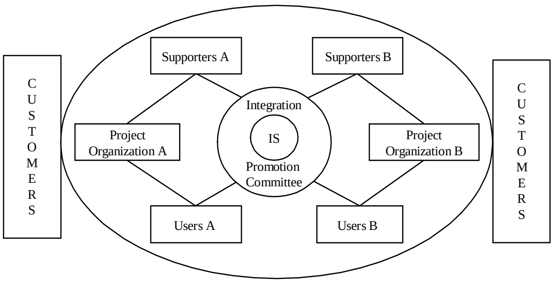


**Figure 2**. IS integration model

In this paper, failure in system integration means that, after integration, a malfunction of computer systems occurs as a result of a disagreement between the components involved in the IS integration.

### 3.2 Review of FTA

Fault tree analysis (FTA) is a deductive technique that focuses on one particular undesired event (US Nuclear Regulatory Commission, 1981). A tree is constructed with that event as the root. Each subsequent event is either a primary event, or an INTERMEDIATE event. INTERMEDIATE events are then developed by being connected to a gate, which is itself connected to other events.

Events within a fault tree are typically discrete occurrences of a faults or events that - while not faults in and of themselves - contribute to a fault higher up the tree. The fault tree then qualitatively illustrates how the undesired event resulted from the primary events, via zero or more INTERMEDIATE events. The symbols encountered in a fault tree are given in Figure 3. The symbols in the figures, as well as the descriptions that follow, were drawn entirely from the *Fault Tree Handbook*.

#### 3.2.1 Event symbols

The BASIC, UNDEVELOPED and EXTERNAL events are the *primary events*. They are not developed any further in the analysis, typically because they are outside the scope of the analysis.

- The BASIC event is an initiating event at the limit of resolution; this fault will not be developed further in this particular analysis.
- The UNDEVELOPED event is considered because of a lack of information, or because the event is of no consequence.
- The EXTERNAL event is expected to happen during the normal operation of the system.
- The INTERMEDIATE event is neither the top-most undesired event, nor a primary event. Further events are attached to the INTERMEDIATE events, usually via gates.

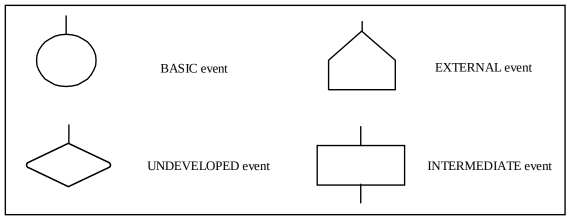


**Figure 3**. Fault tree event symbols

#### 3.2.2 Gate symbols

These gates (Figure 4) should be familiar to most readers in computing:

- The AND gate indicates that the output fault (drawn above the gate) occurs only if two (or more) input faults (drawn below the gate) occur.
- The OR gate indicates that the output fault occurs if at least one of the two (or more) input faults occurs.

In Section 3.2, we briefly reviewed the FTA technique. We use event symbols to express events and factors that have an influence on an initial undesired event, and to indicate relationships by *gate* symbols.

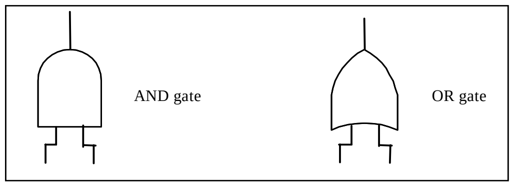


**Figure 4**. Fault tree gate symbols

In this paper, we use Fault Tree Analysis (FTA) to identify explicitly the relationship of events, including the root failure events. FTA is especially useful in applications involving monitoring systems, air traffic control, nuclear power station control, and manned space flights (Michael, 1988).

### 3.3 Framework for analysis of IS integration failure

Up to now, there were no frameworks or models in place to analyze failure in IS integrations. Hence our purpose in proposing the framework for analysis of IS integration failure.

Uniting the IS integration model with the Phases of IS integration (2.2) completes the framework (Figure 5) to analyze such obstacles.

This framework is a mechanism for analyzing (a) failure factors and (b) relationships according to the IS integration model — from the implementation phase to the analysis phase — when system-glitches occur. It uses components of *IS integration model (3.1)* for a horizontal axis, and *phases of IS integration (2.2)* for a vertical axis. We can make use of FTA to specify factors that negatively influence the failure, in accordance with

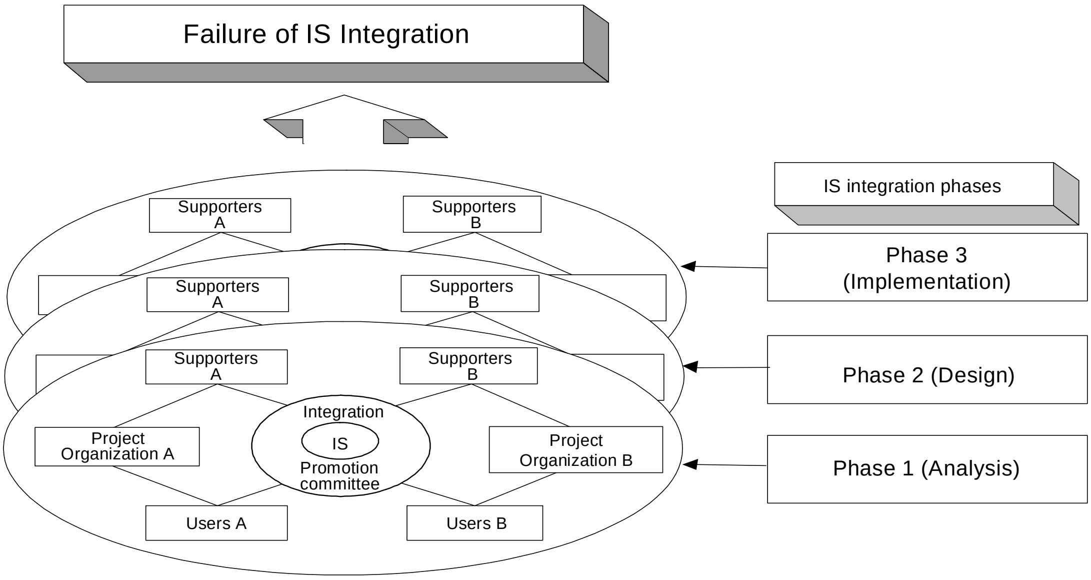


**Figure 5**. A new analysis framework for IS integration failure

the phases of IS integration. So, the framework and FTA can be used as deductive tools of analysis to grasp the factors and relationships which trigger the failure (top event).

Benbasat *et al*., (1987) describe case research as being particularly important for problems during their early, formative stages of research and theory. They see a key use of case studies in the generation of theory from practice. The topic of IS failure is a particularly formative research area and, therefore, one particularly amenable to the case-study approach.

We have interviewed several project managers, each of whom was selected from different IS integration projects in large companies in Japan. The evidence from these case studies suggests that several common failures must be avoided when IS integration is implemented.

In this paper, we analyze the system integration of both Mizuho Financial Group (MFG) and Sompo Japan.

## 4. DESCRIPTION AND REVIEW OF THE CASE STUDIES

The MFG project is a frequently cited Japanese example of recent IS integration failure. MFG ranks number one in Japan in individual asset holdings (at some ¥34 trillion) and conducts business with 70% of the country's listed companies.

Mizuho Bank and Mizuho Corporate Bank were launched on April 1, 2002, under Mizuho Holdings Inc., following the reorganization of the group's three former core banks — Dai-Ichi Kangyo Bank (DKB), Fuji Bank, and the Industrial Bank of Japan (IBJ) — into the two Mizuho banks (Figure 6).

The Sompo Japan project was considered a success, as it fulfilled the desired objectives. As Figure 7 (GIAJ, 2002) depicts, Sompo Japan was established on July 1, 2002, through a merger between Yasuda Fire & Marine Insurance Co., Ltd. and Nissan Fire & Marine Insurance Co., Ltd. There were additional mergers with Dai-ichi Property and Casualty Insurance Co., Ltd. in April 2002, and with Taisei Fire & Marine in December 2002, respectively.

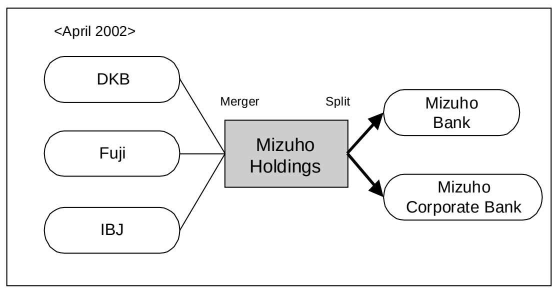


**Figure 6**. Merger of MFG

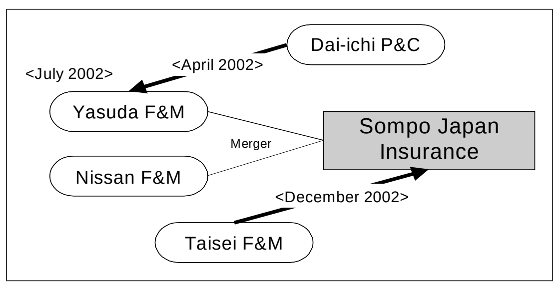


**Figure 7**. Merger of Sompo Japan

In order to analyze and compare the cases, we constructed a framework (Figure 8) by conducting 3 steps.

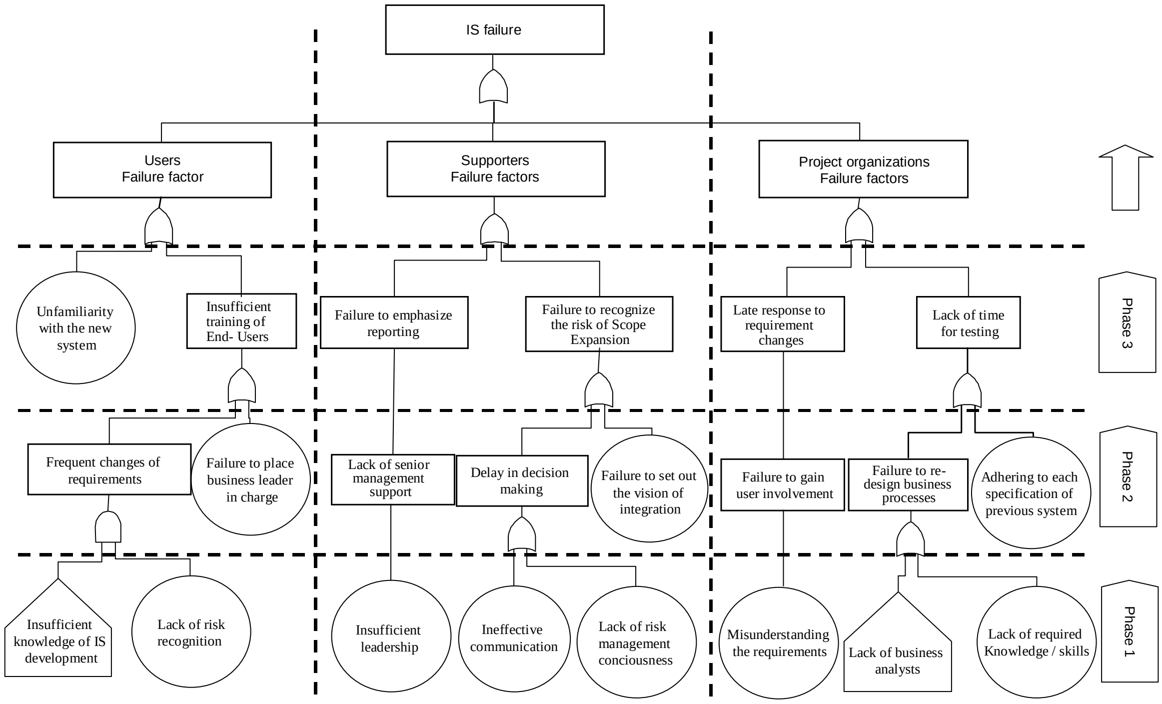


**Figure 8**. Failure factors in IS integration

In the First step, we prepared 3x3 cells based on Figure 5 that used a horizontal axis to represent components of *IS integration model* (3.1), and a vertical axis to represent *phases of IS integration* (2.2). In the next step, so as to collect critical failure factors in IS integration, we interviewed several project managers. Each of them was selected from different projects of IS integration in some large companies in Japan, including MFG and Sompo Japan. Finally, we made use of FTA to specify relationships between these factors that have negative influence on the failure, then allocated these factors to 3x3 cells in accordance with the phases of IS integration.

Figure 8 takes the organizational view point. The organizational view point captures the nature and contents of organizational roles by which organizational activities and tasks are carried out. We focused on behavior expectations, authority, and task performance of organization's component in this paper. Components in IS integration projects have their own roles and agenda to achieve in each phase of the project. We try to point out where and which problems exist in each component by using Figure 8.

Figure 8 can be used for analyzing failure and success factors of IS integration projects. In failed projects, we can explain why the project failed and where the failure factors appeared. Also, by comparing these failure factors, we can identify what critical success factors worked in projects considered successes.

In short, Figure 8 shows which facts may influence failure in the phases of IS integration. Our opinion is that it is necessary to concentrate on the failure factors of components, so as to analyze the ultimate failure. Also, Figure 8 can be used as a checklist of IS integration for failure prevention by IPC, which would carry out the IS integration from now.

## 4.1 MFG Project

On April 1, 2002, Mizuho Bank experienced major computer problems, causing most of its 7,000 automated teller machines (ATMs) to malfunction across Japan. The bank, the retail banking arm of the MFG, was also troubled with delays in money transfers for customers' utility bills and other payments. The crisis included a massive computer failure, whereby customers of Mizuho Bank and Mizuho Corporate Bank were double-billed for utility charges.

The company publicly announced that all the reported problems were the result of a poorly managed integration effort (Mark and Guidi, 2002). The Financial Services Agency (FSA), having investigated the situation, concluded that the primary cause of the problems was a complete lack of cultural integration. Employees followed the requests of their bosses from the original banks, to the detriment of the new company as a whole. Additionally, management was unable to agree upon a single computer system to support the new bank, resulting in the fiasco that was Mizuho's rollout.

One remote cause for the MFG trouble, discovered in the analysis phase, was power struggles among the three constituent banks prior to the integration. It was indeed a major project for them to reorganize into two banks under new concepts. One of the largest challenges was integrating the three banks' different computer systems. In developing their systems, DKB cooperated with Fujitsu Ltd., Fuji Bank with IBM Corp., and IBJ with Hitachi Ltd. In December 1999, four months after the announcement of the three-way merger, the banks decided that the merged retail bank would adopt DKB's computer system. However, the plan was nullified in November 2000, as a result of strong opposition from Fuji Bank. Fuji Bank was concerned that DKB would, if their computer systems were adopted, assume the leadership role in developing the retail banking system. Thus, the three banks agreed to install relay computers to connect the three existing systems, while keeping the existing systems for one year past the April 2002 launch before integrating them fully.

The integration plan in itself had fundamental problems such as delays in decision-making and insufficient computer load testing, in the implementation phase. The launch date of April 1 - the first day of the fiscal year when heavy data processing is normally required - was a poor decision. Additionally, MFG's management erroneously gave the go-ahead for the April 1 launch while cognizant of delays in money transfers toward the end of March. It was further revealed that MFG had turned down requests by Tokyo Electric Power Co. to conduct computer tests beforehand. Such series of oversights indicates MFG's lack of a clear-cut information technology strategy within the framework of the overall management plan.

### 4.2 Sompo Japan Project

Since the end of World War II, the Japanese non-life insurance industry had been enjoying steady growth with stable profitability, until the situation dramatically changed in the wake of the liberalization. Beginning with the revision of the Insurance Business Law (IBL) in April 1996, the following seven and a half years can be described as the hardest and most turbulent years in the history of the industry. The industry has since been forced to undergo a rapid and substantial change due to liberalization and deregulation within the framework of the Japanese *Big Bang*—a national project for structural reform of the entire financial market

In April 2001, three new non-life insurance companies (Sompo Japan, Millea and Mitsui Sumitomo Insurance Co.) were founded as a result of mergers. With the accelerated movement of mergers, consolidation, and affiliation, the non-life insurance industry had reached a peak in its reorganization.

Sompo Japan Insurance Inc. is the result of a merger between Yasuda Fire & Marine Insurance Co., Ltd. and Nissan Fire & Marine Insurance Co., Ltd. in 2002. The larger of the two companies, Yasuda Fire & Marine Insurance Company, was one of the oldest insurance companies in Japan, established in 1888. Sompo Japan is a non-life insurance company providing services to meet both corporate and individual needs. As of March 31, 2003, Sompo Japan - headquartered in Tokyo, Japan - has a consolidated net written premium of 1.34 trillion Yen, or approximately 11.17 billion US Dollars (using the 2003 exchange rate), along with total assets of 5.33 trillion Yen, or approximately 44.35 billion US Dollars.

The project was divided into several phases as follows.

(1) *Analysis phase*: In the first step of the project, members of the project made a master plan on IS integration from November, 2000 to December, 2000. For the next 3 months, they worked out the unification of terms, identifying switchover files in data and reviewing policy books.

(2) *Design phase*: In order to ensure quick decision-making, they decided on the main constituents of problem solving and shared their sense of values, through sufficient cooperation among different departments.

(3) *Implementation phase*: In progress management, they utilized a tool that enabled quantitative management for visualization through a graph showing completion ratio and time series. Next, they made use of Lotus Notes for information-sharing on quality control, problem management, and change control.

(4) *Test phase*: They set the environment conditions for testing, carried out tests, and completed a thorough rehearsal.

We selected three aspects by considering how much each aspect contributed to the success of the Sompo Japan project based on the interview with the project manager.

1) Speed

The system design was decided only one month after the solidification of the policy for the merger of Yasuda, Nissan, and Taisei in November 2000. The option to *select one company's systems*, in this case that of Yasuda fire, was pronounced and the data from the other companies were taken in.

Sompo Japan quickly carried out their plan from system design to final system integration through pilot testing and system development. This was done within a half-year period, although they suffered some large difficulties in the process.

2) Enterprise perspective

Based on the chosen integration option, a master plan was set up. This plan was immediately communicated to the other integration teams to help them assess its impact on their own plan. Along with the integration of systems, a plan was proposed to integrate all IT staff into one IT department that would be responsible for all IT issues in the merged entity as soon as possible.

3) Project governance

According to the overall project governance, each project reported to a specific Enterprise Integration Committee (EIC), which consisted of eight small groups. The role of EIC was to formally approve any changes in scope, schedule, budget, or delivery according to predefined rules. The EIC helped also to resolve potential conflicts between Business and IT project leaders. Further, EIC coordinated the integration program, maintained the master plan, monitored the progress of the project, and made certain that risks were identified and appropriately addressed.

## 4.3 Comparison of the Cases

We compared the two cases of MFG and Sompo Japan by using Figure 8. For example, 'adhering to each specification of previous system' (Project organizations part, Phase 2 in Figure 8) was a failure factor in the MFG project, because they spent too much time deciding which IS should be used after integration. Consequently, the required time to test and prepare for the integrated systems was shortened. On the contrary, in the case of the Sompo Japan project, the decision time was very short thanks to smooth communication especially between users and project organizations, and thanks to the strong leadership of supporters. Thus, they were able to prevent failure caused by 'adhering to each specification of previous system'.

After comparing the two cases in this way, we conclude that it is necessary to carefully consider three aspects - leadership, communication, and enterprise-wide perspective - in order to lead the project to success, because these aspects were clearly different in two cases. These aspects are listed and analyzed below.

### 4.3.1 Communication

In IS integration, communication was identified as one of the most important factors that can influence the success of a project. In the MFG project, effective communication among the components was not evident through all phases. It is critical to communicate what is taking place in such a project, including scope, objectives, and activities of the project.

Conversely, in the Sompo Japan case, they were eager to establish a clear communication system to avoid the failure factor 'Ineffective communications' in Phase 1 of Figure 8. Good communication among the components is essential to accomplish the overall goals of the IS integration project. Especially, communication between users and project organizations directly relates to the success of the project.

### 4.3.2 Leadership

In the MFG project, supporters failed to put someone in charge and thereby centralize the management structure of the project to avoid duplication of effort. As such, it was associated with the failure factor 'Insufficient leadership' in supporters part and Phase 1 of Figure 8. Without question, top management support is critical to the success of a project. It is important to achieve the support of senior management for accomplishing project goals and objectives, and for aligning these with strategic business goals. If not so, conflicts may arise in establishing common requirements.

In the Sompo Japan case, by recognizing the importance of the system integration from the initial phase, supporters quickly decided on the organization and integration option for the system integration, so that the IPC was able to prevent confusion in the implementation phase. They also did not forget to check how the integration process was proceeding by carrying out periodic briefing session in the implementation phase.

### 4.3.3 Enterprise-wide perspective

In the MFG project, there were insufficient enterprise-wide perspectives such as 'Failure to set out the vision of integration', 'Frequent changes of requirements', and 'Failure to gain user involvement' in phase 2. Supporters must propose a specific vision of integration to steer the other components into the right direction. Users and project organizations should make an effort to understand each other and give priority to enterprise's goals. The project must be based on an enterprise-wide design; must define what is needed at the enterprise-level; and then must apply it to the business unit level.

Conversely, in the Sompo Japan case, the integration plan with Tai-Sei F&G was announced when the basic system implementation started. It can also be said that the Yasuda fire project was interrupted at its beginning. But, on this occasion the 3 companies, including Yasuda fire, decided simultaneously to proceed with integration and development.

New system developments were, for example, (a) an accident response system, (b) a contract management system, (c) a next generation system, and (d) a network infrastructure restructuring project. These systems were developed without interruption in spite of the sudden system integration. Eventually, they led the IS integration project to a success by integrating IS centered on these systems.

Therefore, in general, we may conclude that communication, leadership, and enterprise-wide perspective are key aspects for analysis of IS integration, as shown in Figure 9.

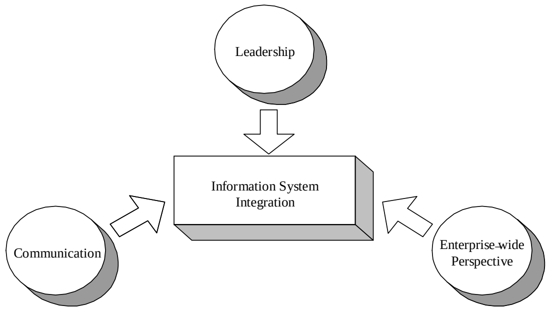


**Figure 9**. Key aspects in IS integration

## 5. CONCLUSION AND FURTHER RESEARCH

This paper has presented a prominent Japanese IS integration project deemed by many as a failure. Analysis of this case provides a valuable understanding of the complexity of IS integration.

Explanatory frameworks such as those of Lyytinnen and Hirschheim, and of Sauer, are particularly useful in broadening the notion of technical systems failure, and for highlighting the political, economic, and social nature of IS failure. Lyytinnen and Hirschheim's concept of expectation failure clearly places blame for such failure in the arena of human interpretation and expectations. Sauer's framework is useful in proposing that it is only when relationships between crucial elements of an information systems project break down irrevocably that the project can be seen as having failed.

The outcome of this study is the proposal of a framework for analysis of the IS integration failure. This framework uses components of the IS integration as horizontal axis and the phases of IS integration as vertical axis, specifying the factors that negatively influence the failure in accordance with the phases of IS integration.

FTA was used to explicitly identify the relationship among events, including root failure events. We regard FTA as a technique to be applied both in analysis of systems and in identification of failure factors in IS integration.

But, there are some limitations in the proposed framework. This framework doesn't provide substantial and actual methods to prevent the failure, and it is only applicable to IS integration followed by business system integration.

There is no foolproof recipe for IS integration success. However, there are many factors that can greatly enhance the chances of success.

In recent years, Enterprise Architecture (EA) is coming into focus as an organized way of optimizing systems as a whole. The main goal of EA is the overall optimization of IT investment, but a well decoupled architecture also makes it much easier to allocate work to IT development teams. There is a strong connection between the architecture and the process of IS integration. Future work may attempt to identify the influence of architecture to the failure of IS integration.

## ACKNOWLEDGMENTS

The authors are grateful for the insightful comments of the referees and the Regional Editor on an earlier version of this paper. They wish to thank all the people who have participated in the evaluation of the system as well.

## REFERENCES

Benbasat, I. I., Goldstein, D. K., and Mead, M. (1987), The case research strategy in studies of information systems, *MIS Quarterly*.

Bignel, V., Fortune, J. (1984), *Understanding systems failures*, Manchester University Press, Manchester.

Boehm, B., Ross, R. (1989), Theory-W software project management principles and examples, *IEEE Transactions on Software Engineering*, **36**, 902-916.

Cole, A. (1995), Runaway projects-Cause and effects, *Software World*, **26**, 3-5.

Ewusi-Mensah, K., Przasnyski, Z. H. (1994), Factors contributing to the abandonment of information systems development projects, *Journal of Information Technology*, **9**, 185–201.

Ewusi-Mensah, K., Przasnyski, Z. H. (1995), Learning from abandoned information system development projects, *Journal of Information Technology*, **10**, 3–14.

GIAJ (The General Insurance Association of Japan) (2002), *General Review of Fiscal 2001*, Non-Life Insurance in Japan Fact Book 2001-2002,

Ives, B., Olson, M. H. (1984), User Involvement and MIS success: a review of research, *Management Science*, **30**, 586-603.

Keider, S. P. (1984), Why Information Systems Development Projects Fail, *Journal of Information Systems Management*, **1**, 33-38.

Klein, H. K., Hirschheim, R. A. (1987), A comparative framework of data modelling paradigms and approaches, *The Computer Journal*, **30**, 8–14.

Lucas, H. C. (1975), *Why Information Systems Fail*, Columbia University Press, New York,.

Lyytinen, K., Hirschheim, R. (1987), Information systems failures: a survey and classification of the empirical literature, *Oxford Surveys in Information Technology*, **4**, 257–309.

Lyytinen, K. (1988), The expectation failure concept and systems analysts view of information systems failures: results of an exploratory study, *Information and Management*, **14**, 45–55.

Mark A, Guidi. (2002), Merger misery, *http://www.accj.or.jp/document_library/1053149845.pdf*.

May, L. J. (1998), Major causes of software project failures, *http://stsc.hill.af.mil /crosstalk/1998/jul/causes.asp*.

Michael, S. D., Ronald, R. W. (1988), *Software Quality Engineering*, Prentice Hall.

Nanjo, A., Hosaka, I. (1989), *Development and operation of Information system* (in Japanese), Oomusha.

Powell, P. L., and Klein, J. H. (1996), Risk management for information systems development, *Journal of Information Technology*, **11**, 309-319.

Sauer, C. (1993), *Why Information Systems Fail: A case study approach*, Alfred Waller, Henley-On-Thames.

US nuclear regulatory commission (1981), Fault Tree Handbook, NUREG-0492.

Yamamoto, S. (2002), M&A to IS integration (in Japanese), *business communication*, **39**, 62-71.