Learning failure in information systems development

Kalle Lyytinen & Daniel Robey*

*University of Jyväskylä, Department of Computer Science & Information Systems, FIN-40 351 Jyväskylä, Finland, email: kalle@cs.jyu.fi, and *Georgia State University*

Abstract. *Information systems development is a high-risk undertaking, and failures remain common despite advances in development tools and technologies. In this paper, we argue that one reason for this is the collapse of organizational intelligence required to deal with the complexities of systems development. Organizations fail to learn from their experience in systems development because of limits of organizational intelligence, disincentives for learning, organizational designs and educational barriers. Not only have many organizations failed to learn, but they have also learned to fail. Over time they accept and expect poor performance while creating organizational myths that perpetuate short-term optimization. This paper illustrates learning failure in systems development and recommends tactics for overcoming it.*

Keywords: Information systems failures, information systems organization, learning, project management, system development methods

INTRODUCTION

The practice of information systems development (ISD) has undergone a radical transformation during the last decade. Advancing technologies have encouraged a migration away from the traditional, life cycle methods of development toward more flexible and dynamic approaches in which reusable components are assembled into working systems in a radically shorter time (Welke, 1994). Today, there are few technical reasons for companies to experience the delays and backlogs that plagued systems development 20 years ago. However, ISD remains a high-risk proposition. Information systems projects continue to fail at an alarming rate, and the problem of 'runaway' development projects has never been more serious. For example, a recent study estimated that American companies spent $59 billion in 1995 in cost overruns on runaway IS projects (Johnson, 1995). Costly failures continue to be reported in the popular press as well as the academic literature (Abdel-Hamid & Madnick, 1990; Myers, 1994; Drummond, 1996; Mitev, 1996; Rosenwein, 1997). Although some portion of troubled ISD projects are turned around successfully, research has generated little insight into means by which managers may avoid risks and prevent project failure (Keil & Robey, 1999). From the growing incidence of failed projects, we conclude that advances in development technologies are not sufficient to improve the rate of successful system implementation. Rather, ISD projects remain susceptible to failures because organizations fail to learn from their own experiences.

Consider the following example: On 11 March 1993 the world was shocked by the sudden cancellation of the Taurus project, which the London Stock Exchange had been developing for more than 6 years. Taurus was expected to be instrumental in the radical restructuring of the securities trade, widely known as the Big Bang, by forming a backbone system for the London Stock Exchange. The project cost the Stock Exchange $130 million, and securities companies invested $600 million more (Drummond, 1996). After years of alternating embarrassments and heroic efforts, Taurus was cancelled before a single module was implemented because the required functionality and performance could never be delivered.

Although Taurus was a very complex project, involving novel technologies and massive scale, ineffective project controls allowed requirements to change continuously throughout the project. Management ignored clear warning signs about technical risks, and powerful interests pushed for Taurus' development despite confusion over the system's purpose and design. In the end, advocates held an almost superstitious faith in the project, dismissing objections and proposals for modifications with comments such as '. . . we have had all those arguments. Great idea but no, we have been arguing about it for 12 years, forget it' (Drummond, 1996, p. 352). With the benefit of hindsight, the Taurus failure should have been averted by adjusting its course based on available information.

In this paper, we draw from the literature on organizational learning to explore reasons for the failure of projects like Taurus. We argue that ISD organizations have *failed to learn* effective means for solving problems such as project risk and project scope, so apparent in the Taurus case. Moreover, in many ISD organizations, ineffective practices have persisted so long that they have become impervious to change. Such organizations, it seems, have *learned to fail*. In the Taurus case, for example, serious problems came to be seen as normal, and participants accepted the self-destructive development trajectory. Using the phrase 'learning failure' in the title of this paper therefore signifies two related meanings: a failure to learn and learned failure.

Ironically, much of the information from which ISD organizations might learn is readily available from previous experience with ISD projects. To avoid learning failure, it is necessary for IS developers to learn from their own and from others' experiences and to use this knowledge to change their development practices. In the parlance of organizational learning, organizations must develop the capability to modify their *theories in use* about systems development. In effect, they must view themselves as *learning organizations*, able to learn from their experience and to effect changes in their own actions. Although learning is a simple concept, which all organizations actualize to some degree, sizeable obstacles may prevent effective organizational learning in ISD. Our purpose is to analyse these obstacles to learning and to recommend means of overcoming them.

We structure our arguments into five main sections. First, we discuss the importance of learning in ISD, identifying external and internal sources of knowledge and the importance of organizational theories in use. Second, we diagnose learning failures in two published case histories. These cases provide examples to illustrate our arguments more concretely; they cannot be considered as research data used to test the validity of our claims. Third, we examine four major reasons why organizations fail to learn: limits on organizational intelligence, disincentives for learning, organizational design and educational barriers to learning. Fourth, we examine the creation of myths whereby ISD organizations learn to fail. Fifth, we offer specific suggestions for overcoming learning failure so that ISD practices may become more effective: knowledge management, incentives for learning, organizational redesign, and reforming IS education. Our analysis is summarized in two models for learning: one that explains learning failure and one that guides positive learning.

ORGANIZATIONAL LEARNING AND INFORMATION SYSTEMS DEVELOPMENT

Most previous studies on ISD problems have looked at specific reasons for failure and generated corresponding prescriptions for ISD practice. Included are such remedies as risk management techniques, user participation, tools to represent system abstractions, and specific recommendations for the organization and management of ISD projects (Lyytinen, 1987; Lyytinen & Hirschheim, 1987; Sauer, 1993). Although these prescriptions may be worthwhile, organizations rarely modify their ISD practices to use such knowledge (Doherty & King, 1998). Many ISD organizations appear unable or unwilling to adjust their practices even when they fail to produce beneficial results. We conclude that advanced development tools and methodologies cannot, by themselves, produce that learning. To help us understand this problem, we examine some of the basic ideas in the field of organizational learning.

Sources of knowledge

An organizational learning perspective directs our attention to the sources of knowledge, both external and internal (Huber, 1991). The forms of external knowledge are diverse, including published reports, confidential documents, Internet sources, commercial databases, and the knowledge possessed by vendors and consultants. In early phases of technology diffusion, external knowledge is often the only means for most organizations to learn because internal experience is unavailable. External knowledge may be acquired by modelling other organizations (vicarious learning), by importing knowledge components directly (grafting), or by depending on intermediaries (Huber, 1991; Attewell, 1992; Fleck, 1994).

Because external learning originates in the experience of others, it may not be relevant or transferable to another organization. Moreover, the validity of external knowledge can be difficult to determine. Therefore, in many cases external knowledge is adopted based on its rhetorical power, symbolic value, or low cost. Given that so much external information is potentially available, organizations tend to seek knowledge that corroborates existing beliefs and values (Hedberg & Jönsson, 1978). The cost of external learning can range from small investments in published reports about new technologies to large investments in consulting expertise. Regardless of cost, externally acquired knowledge seems to bring little competitive advantage to a company because similar knowledge is readily available to any competitor. Yet, the importance of external knowledge is great in ISD contexts. Among the benefits of external knowledge are keeping pace with technological changes and managing the downside risks of systems development.

A strategically more important and relevant source of knowledge is internal, generated from a company's own experience. Internal learning can more often produce competitive advantages because such lessons may be concealed from other companies. Forms of internal learning can range from informal communications among individuals to formal analyses of experience and objective audits of systems development projects. The out-of-pocket cost for such knowledge is many times lower than for external sources, but the indirect cost of learning from experience may be greater due to internal tensions, power plays and time demands.

Theories in use

In addition to considering different sources of knowledge, organizational learning places direct attention upon an organization's 'theories in use' (Argyris & Schön, 1996). In the ISD context, an organization's theories in use are a set of assumed causal relationships between actions taken during ISD and desired outcomes. In one sense, all ISD methodologies are theories in use because they prescribe actions to be taken as instrumental steps toward achieving desired results, i.e. implemented systems. Theories in use may be broader, however, encompassing assumptions about roles played by consultants, users and in-house professionals and their impact on the development process. Each ISD organization holds such a set of core beliefs that becomes a model of cause–effect chains relating ISD activities to their consequences. The model, or theory in use, may differ from the organization's 'espoused' theories, which are more likely to reflect orthodox practices that are not actually used.

Theories in use are derived from absorbing both external and internal knowledge in a relatively uncontrolled and random fashion. The content of theories in use is affected, for example, by recruiting policies, vendor influence, professional press, varied interactions with consultants and educational agencies, organizational structure and management policies, all of which shape the way that past experience is interpreted. Theories in use are invoked as guides for action in the ISD process, especially when they rationalize demands to follow a specific methodology, tool or project management approach. Moreover, by defining the relevant variables in ISD, theories in use largely determine what count as 'facts' about ISD (Quine, 1961; Hedberg & Jönsson, 1978). Theories in use tend to be restricted to the domains covered by their postulated cause–effect relationships.

Theories in use have several characteristics: they are simple, weakly validated in the scientific sense and subject to arbitrary interpretation and modification. Their simplicity makes them easier to remember and implement, like many mnemonic devices, and one of their important functions is to filter out much of the information pertaining to actual experience (Miller, 1993). The validity of theories in use is rarely established in a scientific sense, and little documentation is provided to codify the knowledge gained from experience. Finally, theories in use may vary from one ISD project to another because of the differences in personal interpretation.

An organizational learning approach suggests that we regard every system development project as an experiment that generates evidence with which existing theories in use can be tested and verified. By doing so, the ISD organization can become a learning organization, able to adjust its actions based upon knowledge gained from internal experience and/or external information. But this requires that issues of theory content, internal validity and level of formality be addressed openly. Without a commitment to examining their experiences honestly, ISD organizations will operate with untested theories in use and fail to exploit alternative theories that may prove to be superior.

The concept of organizational learning draws particular attention to the two common problems mentioned in the introduction of this essay: failing to learn and learning to fail. Companies engaged in systems development may fail to learn by disregarding both external and internal sources of knowledge. Although potentially useful revisions to practice may be considered and espoused at a superficial level, ineffective theories in use may persist in practice. Many companies also learn to fail by becoming so attached to their ineffective practices that entertaining new options is not even considered. Members of these organizations may assume that their invalid theories in use are still applicable, or they may cynically acknowledge their inappropriateness while using them anyway.

DIAGNOSING LEARNING FAILURE

The consequences of learning failure in ISD include chronic delays, cost overruns, project cancellations, and the delivery of systems that do not work and are not used. To illustrate these effects and the learning failures that produce them, we briefly examine two published cases. We have chosen these cases because they endured for long periods of time, were acknowledged as major disasters during development, and exhibited a curious reluctance by the actors to use available information to avoid the catastrophic effects experienced. Both cases illustrate the effects of stubborn adherence to existing theories in use that were clearly in need of revision. Although these examples may seem extreme, the literature includes far too many similar examples.

CompuSys

CompuSys is the pseudonym for a large computer company that developed an expert system (given the pseudonym CONFIG) for over 10 years to help its sales representatives configure products for customers (Markus & Keil, 1994; Keil, 1995). CONFIG was designed to help the company's sales representatives produce error-free configurations before quoting a price. In 1981, the CONFIG project was regarded as a real winner with a short payback time, and it was expected to give remarkable savings by reducing costly configuration errors that disrupted manufacturing and damaged customer relations. A similar system used in the production function had worked well, demonstrating the potential value of configuration systems for CompuSys.

Despite the support of top management and the allocation of adequate resources to develop the system, the CONFIG project faced serious challenges from its inception. These included the lack of support by the sales organization, software errors and an unrealistic project schedule. Later, the system experienced implementation difficulties because very few sales people used it. Despite the obvious failure of CONFIG to deliver expected results, the project obtained lavish support for nearly 10 years and continued until a financial crisis and the death of one of the project's managers eroded CONFIG's political support. Despite all efforts to enhance the system, the level of usage remained disappointingly low until CONFIG was 'unplugged' at the end of 1992 and all future development terminated.

The primary reason for CONFIG's failure was that its design concept reflected an inaccurate understanding of the sales function. The designers conceived of CONFIG as a stand-alone support tool to be used by sales representatives to reduce their errors in the product configuration process. But designers completely misunderstood the incentives that motivated sales representatives. Sales representatives were not rewarded for configuring systems correctly or punished for configuration errors. They were rewarded for closing sales, and they had little incentive to use a system that improved configuration accuracy. Although CONFIG was regarded as an exemplary use of expert-system technology, and valued highly in CompuSys' engineering culture, it was not a good business investment. In fact, CONFIG was treated more like a research and development project than a project to increase sales and support marketing.

CompuSys failed to learn from its experience for numerous reasons (Keil, 1995). Essentially, the theory in use was that technological innovation led to success, so CONFIG's perceived value was high despite its low value to the prospective users in sales. At every decision point, financial calculations dismissed contradictory evidence and showed that continued support could produce large pay-offs.

Centco

Centco is the pseudonym for a large corporation that sought for almost 15 years to implement a materials management system in its supply division. Four distinct periods of development were identified (Robey & Newman, 1996). The first four years of the project were marked by an ambitious commitment to an outside consulting firm to develop and implement a system called MMS3. Unfortunately, the consultants had never developed a complete system for any of its clients, and ultimately Centco cancelled the contract after spending \$2.3 million on defining requirements that could not be built. This initial period also was marked by conflicts between MIS professionals and managers from the supply division, who disagreed on who should control the project. The second period, which lasted 5 years, began with a proposal from supply managers to build a new materials management system (MDS) in-house. Conflicts again surfaced between MIS and the supply division, specifically over MIS's insistence on a complete business case and 11 development phases with an estimated cost of \$10 million. To counter these requirements, the supply division contacted an outside vendor with a packaged software product called MMIS II. The third period, which lasted another 4 years, proceeded with the supply division in charge of developing the MMIS II package. Again, difficulties were exacerbated by the ongoing conflict between MIS and the supply division. MIS belittled MMIS II on technical grounds and insisted that an unproven IMS version be acquired instead of the CICS version. Unfortunately, the MIS version suffered from serious flaws and was performing poorly at the vendor's other installation sites. The third period ended amid a corporate financial crisis in which work on the MMIS II project came to a halt. Finally, the last 4 years of the project saw new leadership come to Centco, the supply division and the MMIS II project. A project manager from marketing took over the project, rejustified it, and resolved the conflicts between MIS and the supply division. The vendor was also relieved of MMIS II's maintenance contract, which was astutely awarded to Centco's MIS group. In the end, modules of the new system were successfully implemented, at astronomical cost, roughly 15 years after the initiation of the project.

Centco's problems were analysed from multiple theoretical perspectives, including that of organizational learning (Robey & Newman, 1996). The researchers concluded that Centco learned only belatedly from its previous failures. For the first three periods, spanning 11 years, Centco persisted in the belief that such a system could be built, although no external vendor seemed capable of supplying a workable prototype and internal resources were insufficient for in-house development. Adherence to this repeating pattern of learned failure persisted until major changes in personnel led to the discarding of the theory in use. Notably, the project was removed from the control of both historical combatants, MIS and the supply division, and given over to people with fresh ideas and new approaches to managing projects.

Why do organizations, like the two examples given above, fail to learn from their own, readily available experience? Figure 1 offers a tentative model for addressing this question and for

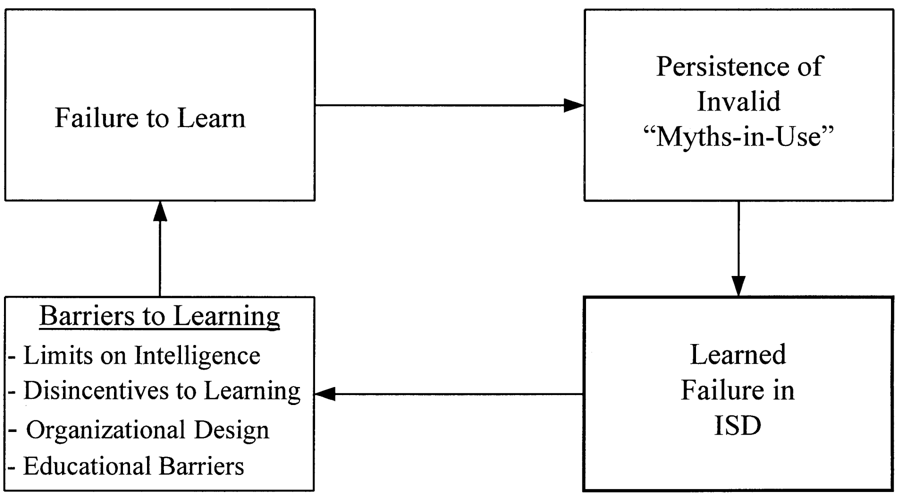

Figure 1. Model of learning failure in information system development.

linking the failure to learn and learned failure into a single vicious circle. Failure to learn is shown to lead to recurrent failure, which in turn becomes regarded as a normal situation when sustained over a long period of time. Thus, adherence to invalid theories in use leads to continued failures to learn because relevant information from experiences is filtered out or reinterpreted. The learning failures represented in Figure 1 may be traced to specific causes, which are analysed in the two following sections.

FAILING TO LEARN: FOUR BARRIERS TO EFFECTIVE LEARNING IN ISD

We doubt that massive ISD failures could be attributed solely to the personal weaknesses or stupidity of individuals. Both CompuSys and Centco were affluent organizations and drew upon the skills of many talented people. The projects were innovative and strategic, attracting considerable interest from the most influential people in their respective organizations. We believe the reasons for learning failure are more deeply rooted in organizational structures and processes that cause smart people to do stupid things. Specifically, we identify four barriers to organizational learning in ISD: limits on organizational intelligence, disincentives for learning, organizational design and educational barriers.

Limits on organizational intelligence

An organization's ability to learn is limited for four specific reasons. First, even the best intentions to learn are affected by *bounds on the capacity to process information* and to make sense of experience (March & Simon, 1958). Information overload is a barrier to building organizational knowledge because it is difficult to learn anything when there is so much to know and so much information to process. Typically, ISD organizations are occupied with too many ongoing concerns to step back and reflect on available empirical information. It is difficult to assess the extent to which this cause affected the failure of the CompuSys and Centco systems, but the pressures to get the systems operational clearly made critical self-examination more difficult. Both CompuSys and Centco tried to expand their intelligence by hiring consultants, but they never questioned their more basic assumptions about ISD.

Second, the *high turnover in most ISD organizations saps them of relevant experience and knowledge*. It is even difficult to transfer experience from one project to another because of time pressures and the specialized requirements and cultures of different users. For example, marketing systems can be very different from financial or production systems, as evidenced by CompuSys' mistaken assumption that CONFIG would be accepted and used in the same way that a production support system had been.

Third, organizational actors become blinded by pre-existing institutional arrangements and related thinking patterns, what Ciborra & Lanzara (1994) called *'formative context'*. This context of organizational and cognitive elements has far-reaching influences: it constitutes a background condition that gives action direction and meaning, and it establishes the range of opportunities. The naturalness of the formative context also shapes actors' experiences by blocking attempts to build new cognitive strategies. It embodies a pervasive texture of relations, which accounts for the inertia of learning and the unawareness of actual practices. Accordingly, organizations often exhibit a curious habit of blinding idividuals to their own mistakes. For example, at Centco the established relationship between MIS and supply prevented serious thought about alternatives and reinforced the futile search for external technical assistance.

Fourth, *theories in use lack adequate scientific validity* because experimental controls are absent, measures are weak, and analyses lack rigor. Frequently, people must try to draw valid conclusions from 'samples of one or fewer' (March *et al*., 1991) without knowing how. Thus, participants may draw erroneous conclusions from their experiences in ISD projects. Organizational learning may be undertaken more rigorously by exploiting existing data, conducting post-mortem analyses and designing natural field experiments. However, most organizations lack people with adequate training in research to increase the scientific validity of the learning process.

Disincentives for learning

The second barrier to organizational learning in ISD is the absence of proper incentives for learning. Opportunities to learn from failure are usually missed because of an obsession with success. Organizations provide many incentives to succeed, but few incentives to fail. Failures are often wiped from an organization's memory out of fear that they will recur, and consequently valuable material for internal learning and theory testing is ignored. Failures precipitate disciplinary actions in which guilty parties are punished. For example, at Centco, the project leader responsible for the MMS3 fiasco was subjected to a formal hearing, banished to a small office and given nothing important to do. He ultimately left Centco to begin a new career writing children's books (Robey & Newman, 1996, p. 44).

When learning does result from failures, the lesson is often limited to knowing how to avoid future failure. Unfortunately, the defensiveness of avoiding failure is unlikely to change action in positive directions. Defensiveness may work well in the short run but may emasculate organizational learning in the long run. For example, the Centco development team depended upon trusted methodologies designed to minimize internal technical risks, but they never regarded the development effort as an organizational experiment until new management changed the whole direction of the development initiative.

Organizational design

Third, organizational design can be a barrier to organizational learning. Departmental boundaries can limit access to relevant information, diminish testing and development of alternative theories in use, and retard the introduction of better ISD methods. Although general knowledge about ISD has progressed astonishingly, the organizational structures within which new methods are used normally reinforce old patterns of action. For example, the traditional structure of IS departments fosters organizational politics and competition for resources. As a result, information technology becomes isolated from the rest of the business, and ISD must be undertaken as a revenue-producing activity for client departments. Compartmentalizing IS narrows the domain for theories in use to technological issues instead of incorporating larger business issues. This was clearly the case with Centco, which preserved the unproductive traditional boundaries between MIS and supply. Under such structural conditions and the incentives they support, sharing information needed to test theories in use is very difficult.

Even in situations in which organizational learning is valued, organizations often minimize the effect of learning by establishing special units that are 'walled off' from core processes. Dedicated units are often created to oversee organizational development, quality assurance, methodology design, risk assessment and other valuable learning activities. These isolated departments act like repositories of knowledge that are rarely accessed. Unlike the monastery library in Eco's *Name of the Rose*, which bore existential value whether anybody read the books or not, repositories of organizational knowledge must be used to be valuable. By separating learning from doing, organizational designs make the results of organizational experiments seem distant and easily ignored by user departments.

Educational barriers

The profession of ISD is characterized by specialized technical training and circumscribed theorizing. Since the dawn of business computing, training in IS has meant 'computer' training, and IS professionals remain technologists at heart. Unfortunately, a technologist's perspective does not encourage an accurate diagnosis of the role of computing in business strategy and operations. For example, Centco's MIS group had ostensibly sound technical reasons for insisting on the IMS version of the MMIS II package, and they viewed IMS as a key aspect of the solution to their client's problem. However, this narrow view rendered the project unmanageable because it overextended the capabilities of the contractor. Whether politically motivated or not, insistence on this technical feature is symptomatic of the narrow mindset induced by specialized professional training.

The profession's technical focus is manifest in the predominant root metaphor for IS professionals: engineering. Accordingly, traditional ISD proceeds on the assumption that a technically valid design that meets functional specifications must have its requirements stated in advance of system building. Unfortunately, learning is discouraged by this presumption; the deeply embedded value of functionality essentially blocks out other metaphors such as coping and learning that promote a more reflective analysis.

LEARNING TO FAIL: THE CREATION AND PERPETUATION OF MYTHS IN USE

Over time, failures to learn blind a company from seeing reasonable alternatives and investigating them. Consequently, the assumptions and routines that comprise standard approaches to system development are sustained as 'myths in-use', despite their inappropriateness. Organizations learn to live with inadequate performance and attribute negative outcomes to external causes rather than their own processes. Challenges to standard development practices are received defensively rather than viewed as opportunities for organizational inquiry or theory testing. Consequently, ISD organizations may refuse to test or refine their myths in use. Essentially, they have learned to fail. Clearly, both Centco and CompuSys provide examples of learned failure in which failing to learn became institutionalized.

Myths are evidence of superstitious beliefs embedded into organizational routines, and ISD practices that have assumed mythical status are insulated from blame for IS failures (Hirschheim & Newman, 1991). Myths prevent reflective discourse about alternative approaches to ISD and perpetuate learning failure. CompuSys' relentless pursuit of expert-system technology to reduce configuration errors and Centco's fetish adherence to external consultants with unproven products are powerful examples of ISD myths. These irrational beliefs were not seriously questioned as causes of failed ISD, and they exercised powerful influence over the participants despite the presence of contrary evidence.

We can identify three generic myths in use that disable organizational learning and reinforce learned failure. The most prominent is the myth of the *technological fix.* Indeed, the entire IS profession perpetuates the myth that better technology, and more of it, are the remedies for practical problems. For instance, applications generators, expert systems, CASE tools, prototyping methodologies, relational technologies, object-oriented programming and other techniques have all been introduced at different times to solve ISD problems. The myth of a technological fix also includes the superstition that a particular development practice or methodology is correct, even when experience demonstrates otherwise (Ciborra & Lanzara, 1994). Moreover, a narrow focus on technique blinds participants to the organizational context of a technique's application, which is exactly where many projects encounter their most serious difficulties (Lyytinen, 1987).

Organizational myths are another common type of unjustified belief. Organizational myths place faith in the power of organizational designs to repair what is wrong. Accordingly, fixing problems is accomplished by reshuffling the organizational chart and redrawing organizational responsibilities and roles. Popular examples of this myth in practice include process reengineering, outsourcing, strategic alliances, and virtual teams. At Centco, for example, failure to learn from experience was perpetuated by the belief that a better solution could be obtained from external software vendors. Although this is often the case, Centco's own experience with outside vendors was disappointing. However, rather than abandoning the myth, Centco searched persistently to find the right vendor.

A third common IS superstition is the myth of a *silver bullet*, an unjustified belief that a rapid and forceful solution will reverse ISD failures (Brooks, 1987; Markus & Benjamin, 1997). Indeed, common reference to silver bullets by ISD professionals reveals a fascination with magical weapons possessed by heroes fighting the forces of evil. Although such metaphors may be considered simple embellishments to language, the symbolism commonly used in professional discourse is apt to reveal important assumptions about the definition and solution of problems (Hirschheim & Newman, 1991). Swanson (1988), for example, argues that IS professionals are frequently cast in heroic roles and armed with weapons to win battles. Unfortunately, the myth of the silver bullet is an inaccurate reflection of ISD, and the search for silver bullets may prevent deeper introspection into the reasons for development failures.

OVERCOMING LEARNING FAILURE: LEARNING TO LEARN

We have identified reasons why ISD organizations fail to learn and shown how learning failures can become institutionalized as myths resulting in learned failure. We have argued that ISD's superstitious adherence to such myths can only make matters worse. There are no real silver bullets that can cure ISD failures, so it is time to 'bite the bullet' and find ways to prevent or overcome ISD problems.

Figure 2 offers a more constructive representation of learning for ISD. Rather than perpetuating learning failures as myths, as shown in Figure 1, Figure 2 shows how learning fosters the development of valid theories in use, which in turn lead to what we call a 'smart' ISD organization. Conventional practice would apply the label 'learning organization' to describe what we mean by a smart organization. However, this jargon does not distinguish between organizations that learn properly and those that learn superstitiously. According to Figure 2, the smart ISD organization engages in four learning activities: knowledge management, learning incentives, organizational redesign and IS education. Each of these addresses one of the primary reasons for learning failure identified earlier. When undertaken on a recurring basis, these practices produce new knowledge and dismantle old myths.

INCREASING ORGANIZATIONAL INTELLIGENCE THROUGH KNOWLEDGE MANAGEMENT

Knowledge management has become a popular buzzword in recent years. In the context of ISD, knowledge management addresses deficiencies in organizational intelligence by reducing the

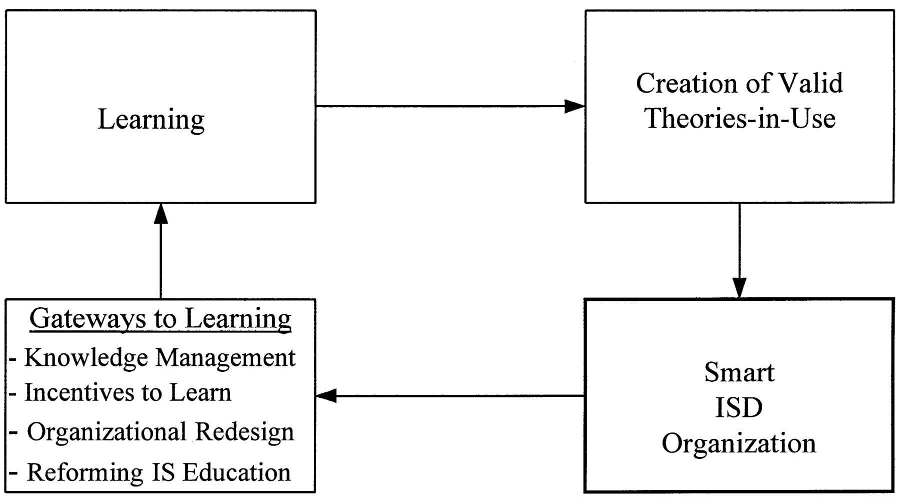

Figure 2. Model of organizational learning in information system development.

boundaries that isolate expert knowledge from its domains of application, by treating ISD projects as experiments from which the organization can learn, and by overcoming the inherent technical biases established among ISD professionals. Knowledge management in ISD requires ISD groups to take stock of the knowledge they already possess and to ascertain what additional knowledge they need to acquire. Formally, the stocks of ISD knowledge can be located in a knowledge repository, created by moving performance data out of production and into a usable format for research and reflection (Purao & Fedchuk, 1997). Many currently popular approaches to process improvement in ISD provide formats for gathering process data and performing such assessments (e.g. Paulk, Curtis & Chrissis, 1995). Unfortunately, most of these focus on internal aspects of software production, such as cost, budget and quality, thereby favouring the identification of symptoms rather than causes.

A more valuable knowledge repository would include both performance metrics and ‘softer’ data. The former could identify symptoms, whereas the latter could enable reflection about the theories in use within ISD. Here the concept of a *system design rationale* can be useful (Conklin & Begeman, 1988). A design rationale captures the history of design decisions and reveals their logic within the domain of the ISD process. A project’s design rationale can support critical evaluation by surfacing deeply rooted assumptions underlying design decisions (thus revealing theories in use) and by guiding refinements and modifications in ISD methods. When applied in post-mortem analyses, the design rationale can provoke more radical discoveries than the usual discussions of project performance, in which participants may simply reinforce past myths by using project history data selectively.

PROVIDING INCENTIVES TO LEARN

To combat the disincentives for learning, the ISD organization must create positive incentives for learning. The worst mistake is to regard the creation of a learning organization as another silver bullet for solving ISD problems. The unrecognized irony in catchy slogans demanding that organizations become ‘learning organizations’ is that organizations have always learned. The problem is with what they have learned. To become smarter, organizations must provide incentives for members to treat experience as a source of knowledge. This includes the counter-intuitive practice of rewarding project members for acknowledging errors and admitting faults. It is impossible for organizations to test their theories in use without determining the causes of negative results. However, in most organizations, bearing responsibility for negative results is associated with punishment. By removing customary penalties for mistakes, ISD organizations can create a climate for learning in which lessons learned the hard way are shared with others who need to know. For example, if the participants in Centco and CompuSys had been given the slightest incentives for bringing bad news up for discussion, the troubled projects in those companies may have been turned around much earlier. As it was, both cases show how members avoided the punishments that some of their colleagues received.

ORGANIZATIONAL REDESIGN

Because much learned failure can be traced to the structure of an organization, overcoming learning failure requires redesign of the organization. Extensive treatments of organizational redesign are found today, complete with radical prescriptions that learning and adaptation become key priorities in making organizations more responsive to customers. However, with rare exceptions, customers do not directly consume the efforts of the ISD organization. ISD is more accurately viewed as a process that adds value to the organization by enhancing other core processes (Rockart & Hofman, 1992). Therefore, prescriptions for organizational transformation should not be applied directly to the design of ISD organizations unless ISD is integrated with core business activities. It makes little sense to optimize the production of software without meshing the software development process with the larger organization **(Paulk** *et al*., 1995**)**.

With these considerations in mind, the ISD organization should be designed around the development process, and clear connections need to be established with ISD's customers, the user groups. Rather than housing the various tasks of the ISD process into different departments, learning in ISD can be facilitated by removing boundaries among the tasks, thereby allowing a smooth flow from beginning to end. This may require a radical rethinking of the roles and responsibilities, and it will also set new performance targets for ISD (not just time and budget). In redesigning the organization, ISD should be understood as a value-adding business process, whose performance can be measured (Hammer, 1996). Modifications to the process must be made when performance targets are not achieved. This means also reconfiguring the process and its governance, and analysing the critical technologies, skills and knowledge to execute that process.

REFORMING IS EDUCATION

As noted earlier, the IS profession is typically identified as a technical craft. Making the learning process effective requires that values and expectations of the profession change, and this can best be done by reforming IS education. Currently, people recruited into the ISD function in most organizations assume that their biggest challenge is to acquire new technical knowledge. This is a limited view of learning because ISD professionals also need to learn how to exercise judgements in situations where technical and business issues are inseparable. No amount of formal systems training can remove the value of learning from experience. IS education needs to place a stronger emphasis on *organizational* problem solving as opposed to the solution of technical problems. Judgement is traditionally developed through simulations and case studies, in which students confront real problems and defend their proposed solutions. This approach to education contrasts sharply with the study of blueprint methodologies and technological components that currently characterize IS education.

In summary, ISD organizations can *learn to learn* by following the specific strategies discussed above and summarized in Table 1. Although these recommendations may seem

Table 1. Strategies for learning to learn

Reason for learning failures	Means to overcome learning failures
Limits on organizational intelligence	Increasing organizational intelligence through knowledge management
Disincentives for learning	Providing incentives to learn
Organizational design	Organizational redesign
Educational barriers	Reforming IS education

obvious, they are not simple to implement. At first, our recommendations bear similarity to the many calls for awareness of organizational issues in ISD. However, widespread awareness of these issues has apparently not influenced practice (Doherty & King, 1998). Indeed, the difficulty of implementing obvious recommendations, such as adjusting practice based on feedback from experience, is an apt illustration of the need for improved approaches to organizational learning in ISD. The challenge is to move beyond the habit of *espousing* appropriate theories to the critical step of *changing action* by revising theories in use. Accordingly, our suggestions for overcoming barriers to learning require concrete actions that are, admittedly, difficult to implement. Nonetheless, organizations seeking improvements in ISD will need to confront those difficulties and make a commitment to practising ISD differently.

CONCLUSION

Information systems development is a high-risk undertaking where the risks often realize dramatic consequences, both positive and negative. In this paper we have argued that one reason for ISD failures is an absence of organizational intelligence applied to the complexities of ISD. ISD organizations tend not to view development projects as opportunities for modifying their theories in use; rather, much of ISD is fashioned after dominant ideological myths. ISD organizations do not learn from their failures for a variety of reasons: limits of organizational intelligence, disincentives for learning, organizational designs, and educational barriers. Over time, adherence to unexamined practices produces learned failure in ISD organizations, and poor performance becomes an accepted standard. Such mediocre performance is accounted for by myths and fuelled by a focus on short-term optimization.

The major conclusion from our analysis is that organizations often misplace their attention on what and how they should learn. Organizations are keen to learn from external sources and are generally eager to accommodate new technologies (Attewell, 1992). They encapsulate knowledge into methodologies and their supporting tools, making them less open to critical review and revision. But although ISD organizations acquire knowledge in this way, they often fail to make sense of their own experiences. Learning from experience is not valued, and there are few processes and routines that help organizations to promote learning.

Learning from experience can be a long and tortuous adventure because it challenges established myths and their supporting structures in radical ways. But this seems to be the only way to overcome the vicious circle of learning failure. ISD organizations must recognize the value of obtaining more valid knowledge about their core activities and devise new mechanisms for acquiring, maintaining and transferring that knowledge. They must also redesign their incentives and organizational designs accordingly. Finally, they must view systems development processes as value-adding processes that serves the needs of multiple stakeholder groups.

The problems encountered by CompuSys, Centco and countless other companies could potentially have been avoided if these organizations had paid more attention to their own experience and acted on that knowledge. The metaphor of learning suggests new ways to understand and analyse recurrent problems in ISD, and it offers hope for future improvements. In this essay, we have used these metaphors to focus on learning failure, admittedly a negative focus. In fact, we believe that much of the practice of ISD has failed to live up to its promise. For all the great accomplishments of information technology, the process of application development continues to be plagued by ineffective practices. By focusing on their own experiences in developing IS applications, developers can learn to move in more positive directions.

ACKNOWLEDGEMENTS

The authors thank Mark Keil, Lars Mathiassen and Suzanne Pawlowski for their contributions to this paper. This research was funded in part by the Academy of Finland.

REFERENCES

Abdel-Hamid, T.K. & Madnick, S.E. (1990). The elusive silver lining: how we fail to learn from software development failures. *Sloan Management Review*, **Fall**, pp. 39–48.

Argyris, C. & Schön, D.A. (1996). *Organizational Learning II*, Addison-Wesley, Reading, MA.

Attewell, P. (1992). Technology diffusion and organizational learning: the case of business computing. *Organization Science*, **3** (1), 1–19.

Brooks, F. (1987) No silver bullet: essence and accidents in software engineering. *Computer*, **April**, 10–19.

Ciborra, C. & Lanzara, G. (1994). Formative contexts and information technology: understanding the dynamics of innovation in organizations. *Accounting, Management and Information Technologies*, **4** (2), 61–86.

Conklin, J.E. & Begeman, M.L. (1988). gIBIS: a hypertext tool for exploratory policy discussion. *Proceedings Of CSCW '88*, 140–152.

Doherty, N.F. & King, M. (1998) The importance of organisational issues in systems development. *Information Technology and People*, **11** (2), 104–123.

Drummond, H. (1996). The politics of risk: trials and tribulations of the Taurus project. *Journal of Information Technology*, **11**, 347–357.

Fleck, J. (1994). Learning by trying: the implementation of configurational technology. *Research Policy*, **23**, 637–652.

Hammer, M. (1996). *Beyond Reengineering: How the Process-Centered Organization is Changing Our Work and Our Lives*, Harper Business, New York.

Hedberg, B. & Jönsson, S. (1978). Designing semi-confusing information systems for organizations in changing environments. *Accounting, Organizations and Society*, **3** (1), 47–64.

Hirschheim, R. & Newman, M. (1991). Symbolism and information systems development: myth, metaphor, magic. *Information Systems Research*, **2** (1), 29–62.

Huber, G.P. (1991). Organizational learning: the contributing processes and the literatures. *Organization Science*, **2** (1), 88–115.

Johnson, J. (1995). Chaos: the dollar drain of IT project failures. *Application Development Trends*, **2** (1), 41–47.

Keil, M. (1995). Pulling the plug: software project management and the problem of project escalation. *MIS Quarterly*, **19**, 421–447.

Keil, M. & Robey, D. (1999) Turning runaway software projects around: the de-escalation of commitment to failing courses of action. *Journal of Management Information Systems*, (in press).

Lyytinen, K. (1987). Different perspectives on information systems: problems and solutions. *ACM Computing Surveys*, **19**, 5–46.

Lyytinen, K. & Hirschheim, R. (1987). Information system failures: a survey and classification of the empirical literature. In: *Oxford Surveys in Information Technology, 4,* Zorkoczy P. (ed.), pp. 257–309. Oxford University Press, Oxford.

March, J.G. & Simon, H.A. (1958). *Organizations*. Wiley, New York.

March, J.G., Sproull, L.S. & Tamuz, M. (1991). learning from samples of one or fewer. *Organization Science*, **2** (1), 1–13.

Markus, M.L. & Benjamin, R.I. (1997). The magic bullet theory in IT-enabled transformation. *Sloan Management Review*, **38**, 55–68.

Markus, M.L. & Keil, M. (1994). If we build it, they will come: designing information systems that users want to use. *Sloan Management Review*, **35** (4), 11–25.

Miller, D. (1993). The Architecture of Simplicity. *Academy of Management Review*, **18** (1), 116–138.

Mitev, N.N. (1996) More than a failure? The computerized reservation systems at French railways. *Information Technology and People*, **9** (4), 8–19.

Myers, M.D. (1994) A Disaster for everyone to see: an interpretative analysis of a failed IS project. *Accounting, Management and Information Technologies*, **4** (4), 185–201.

Paulk, M., Curtis, B. & Chrissis, M. (1995). *The Capability Maturity Model: Guidelines for Improving the Software Process.* Addison-Wesley, New York.

Purao, S. & Fedchuk, N. (1997) Creating a knowledge repository to assist the IS developer: the nationsbanc Experience. *ISACC '97Proceedings*, Monterrey, Mexico.

Quine, O. (1961). *From a Logical Point of View.* Harvard University Press, Boston, MA.

Robey, D. & Newman, M. (1996). Sequential patterns in information systems development: an application of a social process model. *ACM Transactions on Information Systems*, **14** (1), 30–63.

Rockart, J. & Hofman, J.D. (1992). systems delivery: evolving new strategies. *Sloan Management Review*, **Summer**, 21–31.

Rosenwein, M. (1997). The optimization engine that couldn't. *OR/MS Today*, **24** (4), 26–29.

Sauer, C. (1993). *Why Information Systems Fail: A Case Study Approach*. Alfred Waller, Henley-on-Thames.

Swanson, E.B. (1988). *Information System Implementation: Bridging the Gap between Design and Utilization*. Irwin, Homewood, IL.

Welke, R.W. (1994). The shifting software development paradigm. *Data Base*, **25** (4), 9–16.

Biographies

Kalle Lyytinen is a professor in Information Systems at the University of Jyväskylä, Finland. He serves on the editorial boards of several leading IS journals, including *Information Systems Research*, *MIS Quarterly* and *Information Systems Journal*. He has published over 70 articles and edited or written six books. His research interests include information system theories, system design, system failures and risk assessment, computer-supported cooperative work, and diffusion of complex technologies.

Daniel Robey holds a joint appointment as Professor in the Departments of Computer Information Systems and Management at Georgia State University. His current research deals with the consequences of information systems in organizations and the processes of system development and implementation. He serves on editorial boards for *Accounting*, *Management and Information Technologies*; *Information System Research*; *Organization Science*; *Canadian Journal of Administrative Sciences*; *Information Technology & People*; and *Journal of Information Technology Management*.