Analysis of Systems Development Project Risks: An Integrative Framework

Merrill Warkentin
Mississippi State University

Robert S. Moore
Mississippi State University

Ernst Bekkering
Northeastern State University

Allen C. Johnston
University of Alabama at Birmingham

Abstract

Information systems development projects are a significant expenditure of time, effort and money for many enterprises. Historically it has been estimated that 50-80% of projects fail to achieve their objectives for a variety of reasons. Researchers have identified numerous factors associated with system development failure. In this paper, we first synthesize the vast research regarding systems development risk factors and provide a framework that illustrates interactions between risk factors. The framework was used to develop an open-ended questionnaire that was answered by an inter-industry group of experienced systems development engineers and project managers. Analysis of their reports indicates that experienced professionals perceive that all risk factors (technical, resource, etc.) ultimately derive from organizationally-oriented factors, to be solved with organizational responses. This holistic viewpoint of risk assessment is counter to that of systems professionals more involved in day-to-day development decision making. For these developers, risks are more likely to be characterized as fitting into traditional discrete categories. This apparent dichotomy of risk importance was further investigated through an intra-organizational study which directly assessed how professionals recognize and treat risks in the development process. Results illustrate that a successful project environment may be characterised as one in which all systems professionals maintain a holistic view of organizational risk and that organizational culture, as opposed to experience, may predicate such an environment. Implications and future research directions are discussed.

Keywords: Systems development; project risk; risk factors; success; failure; organization; software

ACM Categories: K.6.1

Introduction

> "We have learned many valuable lessons over the past few years and, as a result, will be able to apply these lessons and avoid many of the pitfalls that befell this project in the past." (Congressional Testimony of Robert S. Mueller, III, Director, Federal Bureau of Investigation, February 3, 2005)

Director Mueller's statement regarding the failed half-billion dollar Trilogy project could have been said by

countless CEOs, CIOs and IT directors across the world and clearly illustrates that systems development processes can become "high profile" when they fail. Systems Development (SD) projects are undertaken by thousands of enterprises every year. Some are large endeavours requiring extensive resources (time, personnel, and money), with complex inter-organizational development processes and sophisticated technical requirements, such as the Trilogy project, while others seem relatively small and simple affecting a department or user group. Employees of the organization develop some systems "in-house" and other projects may be outsourced to firms specializing in SD activities. Regardless of size and scope, a 2004 global survey of IT executives found that, while 29% of all projects were considered successful (delivered on time, on budget and with stated deliverables), 53% experienced some type of problem and 18% of all projects were cancelled before completion or never implemented, (Standish Group International, 1995; see also Standish Group International, 1999; Standish Group International, 2001; Standish Group International, 2003). Only 34% of IT projects at Fortune 500 firms are completed successfully (Standish Group International, 2003).

Risk researchers such as Ewusi-Mensah (2003) corroborate the high failure rate of SD projects, estimating that one-third of SD projects fail or are abandoned. The unacceptably high rate of system failure across the board (Doherty & King, 2005), regardless of the size or complexity of the project, has commanded investigations of why these problems occur and what we can do to prevent them (Nelson, 2007). Furthermore, recent evidence suggests that reports from project managers of on-going projects may be optimistically biased (Snow et al., 2007), distorted or even non-reported (Keil et al., 2007), thereby further exasperating managers' efforts to detect potential project pitfalls.

Many researchers have sought to identify the key risks associated with SD projects (Lyytinen et al., 1998; Wallace, 1999; Keil et al., 2000; Schmidt et al., 2001; Wallace et al., 2004; Sherer & Alter 2004), leading to several innovative risk assessment methods (Al-Shehab et al., 2005; Gotterbarn & Rogerson, 2005; Jiang et al., 2001). In light of those findings, managers have sought to assess such risks in the systems planning process (Alter & Ginzberg, 1978; Alter, 1979; Segars & Grover, 1996; Armour 2005; Nelson 2007), systems evaluation process (McFarlan, 1981), systems development (Tiwana & Keil, 2006), and managerial assessments (Barki et al., 2001), usually considering each risk to be independent of the others. Research has established frameworks for risk assessment and normative models for risk reduction. Furthermore, research has established that IT project managers with more experience have risk perceptions that differ from those of more junior managers (Stephen et al., 2007). In this paper, we suggest that interactions between risk factors are often driven by organizational factors and successful amelioration of risks is most likely to occur when the organizational culture encourages the entire enterprise to adopt a "big picture" approach to SD.

To explore this proposition, we first develop a coherent list of the risks identified in the prior literature, which have traditionally been categorized as technical, resource constraint, organizational, and "other" risks, then organize the risk factors into a framework indicating their interactions. Using this framework, we next present two exploratory interpretive studies that expand on IT professionals' perceptions of risk in their project experiences. The first study provides a cross-industry perspective of the relative importance of risk factors, while the second provides an intra-organizational assessment of how one firm's SD professionals perceive risks in their organization. We present and discuss the findings of each study, and conclude with future research opportunities and practical applications for project managers.

Systems Development Risk Factors

Technical Risk Issues

The first of three primary categories of SD risk comprises "technical" risks that arise from hardware or software errors or incompatibilities, as well as other complex phenomena that are grounded in technological aspects. Technical risks have long been identified as potential impediments to successful project completion. Alter (1979) identified technology as one of eight SD risk factors, and McFarlan (1981) identified a lack of experience with technology as a significant threat to success. Since that time, technical risks have been included in nearly every SD risk assessment framework.

Technical risks may be encountered in nearly all facets of an SD project. Technical risks have been associated with the introduction of new, or "bleeding edge," technology (Schmidt et al., 2001), access control mechanisms (McGraw, 2002), the acquisition of new hardware and software (Barki & Talbot, 1993), and the involvement of multiple vendors in an SD project (Barki & Talbot, 1993). Lyytinen et al. (1998) suggest that certain technical risks, such as compatibility, manifest early in the SD lifecycle, while other technical risks such as extendibility,

maintainability, and reliability are associated with later phases of the lifecycle.

The impact of technical risks cannot be understated. A PriceWaterhouseCoopers survey of 120 SD project failures leading to litigation between 1975 and 2000 revealed that the majority of cases involved "systems failure," where the system did not perform acceptably or did not work at all (Webster, 2000). A survey administered by Cutter Consortium indicated that 67% of civil legal claims litigated by the respondents in connection with SD projects were brought because the system failed to live up to the claims of the developer, and 45% of such claims were brought because the system had defects that rendered the system unusable (Bednarz, 2002).

Resource Constraint Risk Issues

Another significant source of risk to the success of SD projects is the limitation of organizational resources available for the project. Haber (2003) asserts that companies must always focus on ROI and "the triangle, and balance, of cost, time, and quality." Simply throwing resources at a project may, in fact, decrease its quality and potential for successful completion (Hayes, 2005). Laudise and Nuara (2002, p. 299) suggest that "three critical resources – people, time, and money" must be scheduled and allocated in advance and monitored regularly throughout the SD process. Ewusi-Mensah (2003) cites cost overruns as a significant cause of project failure. Difficulties within the development process pose direct burdens on these resources, and if given the opportunity through project mismanagement, may compound over time (Tiwana & Keil, 2004). For example, issues regarding ownership of code resulted in a three- month delay in transferring a transportation program from Transdyne Inc. to Honeywell. The result was a cost increase of $7.2 Million for Boston's infamous "Big Dig" project (Songini, 2005).

Cost-benefit analysis has always been a cornerstone of systems feasibility studies, but the accurate assessment of total costs (development as well as operating costs) and benefits presents many challenges. For example, many of the benefits of a new system may be intangible and, therefore, difficult to quantify. Inaccurate cost-benefit analyses can lead to the development of a system that later proves to have a negative ROI.

Organizational Risk Issues

Organizational factors became recognized as significant risks to the success of SD projects in the mid 1970's (Lucas, 1975), and subsequent research suggests that organizational risks eclipse technical risks as the primary determinants of success (Hornby *et al.*, 1992; Ewusi-Mensah & Przasnyski, 1994; Clegg *et al.*, 1997; Doherty & King, 1998; Doherty & King, 2001a; Standish Group International, 2001; Doherty et al., 2003). A survey of senior IS executives found that half of the factors contributing to decisions to cancel SD projects were organizationally-oriented, ranging from ambivalent senior management support to a lack of end-user involvement (Ewusi-Mensah & Przasnyski, 1994). A recent global survey of IT executives (Standish Group International, 2001) found that the reason for most project failures "…was not for lack of money or technology; most failed for lack of skilled project management and executive support" (p. 1). Successful projects, on the other hand, were linked to executive support, a high level of user involvement, the assignment of an experienced project manager, and clear business objectives (Standish Group International, 2001). Faraj and Sambamurthy (2006) found leadership to be a significant antecedent of project success when there was a high degree of task uncertainty or team expertise. Trust and communication issues have also been cited as significant organizational risks, particularly when SD projects are outsourced to an external firm (Morgan & Hunt, 1994). For some firms, "80-90% of the important issues today are organizational" (Clegg et al., 1997, p. 856). Despite the magnitude of organizational risks, in practice they are often not explicitly acknowledged or managed (Doherty & King, 1998), and when they *are* addressed in the development process, they are often treated by the wrong people at the wrong time (Doherty & King, 2003). In some instances, improper treatment of organizational risks leads to a culture of failure derived from a commitment to ineffective methods and unsuccessful knowledge management practices (Lyytinen & Robey, 1999). However, King and Doherty (2001b) argue that attention to organizational issues in the SD process is positively correlated with the maturity of an organization's use of the technology.

Organizational risks stemming from organizational culture, structure, and business processes have impacts on technical SD issues as well as human-centred issues such as task redesign (Doherty et al., 2003), thus creating a wide range of potential trouble points. While the majority of systems developers "treat most organizational issues at some point in the systems development process" (p. 57), those who do not are much more likely to encounter project failure. Further, the timing of the treatment of organizational issues is important. While treating an organizational issue in one phase of the development process (feasibility, analysis and design, or implementation)

appears to be no more effective than treating it in a different phase, there is evidence to suggest that repetitive treatment is valuable (Doherty et al., 2003). In other words, "those organizations that treat an organizational issue in more than one phase of a systems development project are likely to have higher levels of systems' success than those that do not" (p. 52).

Another study (Al-Mushayt et al., 2001) sought to determine the extent to which the 14 organizational issues adopted from Doherty and King (1998) are addressed in SD projects and whether the use of best practices aids in the treatment of these issues. This study also explored the link between SD methodology and the treatment of organizational issues. Findings indicate that there is a strong tendency to address organizational issues, with organizational contribution issues most frequently treated. Organizations utilizing best practice methods were able to treat a wider range of organizational issues than those that did not. A well-balanced project development team is more adept in addressing organizational issues. This capability translates into a greater number of issues that can be addressed throughout the project development lifecycle. Finally, findings indicate that there is no significant relationship between the type of SD methodology and the degree to which organizational issues are addressed. Not surprisingly, structured methods such as SSADM do not assist in treatment of organizational issues; however, it is notable that neither did socio-technical or prototyping methods.

Other Risk Factors

Additional risks arise from environmental, cultural, user acceptance, ethical, and legal factors, which are usually categorized together in the prior literature as "other" risks. Legal risks, for example, become particularly acute when SD projects are outsourced. A mismatch of expectations and poorly drafted contracts are often the source of disputes, partly because "[t]he vendor goes by defined contractual obligations, whereas the client looks to solve business issues…" (Bednarz, 2002, p. 2). Legal disputes also commonly arise in connection with software warranty issues, breaches of confidentiality, and the re-use of code by outsourcers in systems built for competitors of the client. While such legal and other risks may be significant, further discussion of them is beyond the scope of this paper.

Synthesis and Framework

Based on the aforementioned literature streams, we conducted an extensive synthesis of micro- and macro-level SD issues previously employed in theoretical models and empirical studies. Our synthesis of the extant research resulted in a comprehensive list of technical, resource constraints, organizational, and "other" risk factors. These risk items and their assigned categories are distilled into the collection of risks shown in Table 1. This succinct list of risks captures the essence of the majority of SD risk research, integrating the pre-eminent research of Barki and Talbot (1993), Schmidt et al. (2001), and Wallace (1999) with organizational and emerging risk research. The specific categories to which the risks are assigned in Table 1 is dependent upon the context of the study in which the risks were identified. In many cases, a risk may be designated within multiple categories, thereby suggesting that perspective and context do, indeed, have much to do with the manner in which a risk factor is addressed.

Our next step was to integrate how each of the noted risks have been conceptualized and linked to project risk factors. This integration allows one to easily see the relative scope and impact a single risk item has on the development process. As can be seen from the Table 1, SD risk factors have been considered as different types of risk, depending on the context. For instance, the item "Project Leadership Problems" can result from insufficient funds to hire a better project manager (resource constraints), internal politics and power struggles (organizational), and even contractual obligations (other). Additionally, the likelihood of a single risk item to be classified into multiple contexts by its nature suggests an interactive element—that is, risks that appear in multiple contexts might have an increased opportunity to interact.

Methodology

In order to investigate the relative interactive effects and influences of risks, we proceeded in two research stages. In Research Stage 1, experienced informants from several industries were contacted and requested to participate and complete a questionnaire composed of open-ended questions. In the second stage, we pursued an in-depth investigation of SD risks by creating a new questionnaire (based on the findings from Stage 1) designed to assess risk perceptions and risk resolutions from within a single organizational context. The process of developing each of these stages and analysis of the results is discussed in turn. To enable a more comprehensive understanding of the assessment of the true nature of each risk evidenced by our informant's comments, we directly tie the results of each stage of our investigation back to the underlying theoretical foundation presented in Table 1.

#	SD Risk Factor	Technical	Resource Constraint	Organizational	Other
1	Inability to acquire necessary hardware	[5]			
2	Inability to acquire necessary software	[1, 5]			
3	Inadequate hardware vendor support	[2, 5]	[2, 5]		[11]
4	Inadequate software vendor support	[5]	[2, 5]		[11]
5	Project technical complexity	[2, 3, 5]	[12]	[12]	
6	Technical incompatibility with existing systems	[1, 2, 3]	[12]	[12]	
7	Technical incompatibility between new system components	[2, 5]	[2, 5, 12]	[12]	
8	System requires connectivity between multiple firms	[3]	[12]	[12]	
9	Large size of project (large number of departments or users)	[3, 5]		[2, 4]	
10	Large size of project team (large number of developers)	[2, 3, 5]		[4]	
11	Insufficient or inappropriate staffing	[1, 5]	[2, 5, 12]	[4, 5, 12]	
12	Team's lack of skills or expertise	[2, 3, 5, 6]	[2, 5, 12]	[4, 5, 12]	
13	Team member communication or compatibility problems	[2, 3]	[2, 12]	[2, 10, 12]	
14	Team instability		[2, 12]	[1, 4, 12]	
15	Project leadership problems	[6]	[7]	[2, 5, 9]	[11]
16	Lack of effective development process or methodology	[2]	[7, 12]	[4, 12]	
17	Inadequate planning	[2]	[2, 12]	[2, 12]	
18	Unclear or misunderstood scope or objectives		[2, 12]	[2, 9, 12]	
19	Changing scope or objectives during project		[12	[2, 12]	
20	Inaccurate or vague user requirements		[12]	[2, 9. 12]	
21	Organizational transition difficulties	[3]	[12]	[2, 12]	
22	Lack of user involvement		[3, 5, 12]	[2, 5, 8, 12]	
23	User expectations don't match project objectives	[3]		[2, 5]	[11]
24	Conflict between user departments	[2, 3, 5]		[2, 5]	
25	Budgetary or financial constraints		[1, 2]	[2]	
26	Lack of top management commitment to project		[2, 12]	[2, 8, 9, 12]	[5]
27	Organizational politics		[12]	[2, 12]	[5]

1. Ewusi-Mensah, 2003
2. Schmidt et al., 2001
3. Barki and Talbot, 2001
4. Al-Mushayt et al., 2001
5. Barki & Talbot, 1993
6. McFarlan, 1981
7. Tiwana & Keil, 2004
8. Ewusi-Mensah & Przasnyski, 1994
9. Faraj & Sambamurty, 2006
10. Morgan & Hunt, 1994
11. Bednarz, 2002
12. Wallace et al., 2004

Table 1. Synthesis and Interaction of SD Risk Factors with Examples of Classifications from the Literature

Research Stage 1 – Cross-Industry Perspectives

Questionnaire Development. The synthesized risk items shown in Table 1 served as the foundation for the development of an open-ended, probing questionnaire, reproduced in Appendix A. The individual items contained in the questionnaire were derived from the previously published risk research literature, then iterated incrementally by the team of five authors through a process of discussion and negotiation, and then presented to four experienced developers for assessment and further refinement. It incorporates the issues presented by previous published manuscripts as well as those identified by practicing developers. The purpose of the questionnaire design was to evoke detailed experiential knowledge by stimulating responses related to self-selected categories of SD risk. It was also designed to elicit descriptions of ameliorative actions which contributed to project success as well as causes of failure. The questionnaire was delivered via email due to disperse geographic locations of our experienced informants; however, each informant was also given the choice of conducting a telephone interview.

Cross Industry Experienced Informants. A diverse group of eight experts with national and international experience in various types of systems project management was recruited from both large and small organizations. This exploratory investigation required considerable time commitment from the participants; thus the authors requested the involvement of SD professionals in industry and government with whom we had personal acquaintance. Each informant was presented with a standard research consent document and was given the opportunity to choose the manner in which he or she would be identified. Each was asked to provide as much or as little detail in their written responses as they would like. While some participants skipped some questions, most were very forthcoming in providing complete answers to most or nearly all questions. (In fact, their personal choice to answer or skip questions was fundamental to incorporating their experientially-derived assessments of *important* SD risk and its causes.) Table 2 provides a summary of the experienced informants, using each participant's choice of self-description. Pseudonyms were assigned to mask their identities.

The group of participants comprised a diverse set of developers and SD project managers with experience ranging from four to 25+ years. All members of the group reported having been involved in numerous large projects in capacities ranging from senior project managers to junior systems developers. Some of their projects were highly structured, as in the case of a turnkey system for tracking maintenance and repair of medical systems, while others were less structured, as in the case of a system controlling all live audio and video of a large broadcasting company that was one of the first to include active standby systems in case of failure. Some were new projects, such as the development of a new eGovernment system, while others involved adaptation of existing systems.

Analysis of Stage One Responses

Responses from the respondents resulted in over 7500 typed words representing over 20 pages of verbatim responses. The time spent completing the questionnaire ranged from thirty to ninety minutes. The typed responses of the respondent panel were read several times and analyzed by each of the study's authors. The iterative analysis revealed several themes, or consistent patterns of responses, across individuals that illustrate the interactive nature and effect of risk factors. After comparison of identified themes between the authors, differences were resolved through discussion until consensus was achieved on each theme.

Informant	Title	Type of Organization	Yrs*	Primary Role
Greg	Enterprise Architect	State Government	15	Project Manager
Harry	Senior Engineering Mgr	Global Medical Technology	8	Technical manager
Steve	Senior Programmer	International Transportation	4	Lead Developer
David	Software Engineer	Distribution/Retail	22	Tech Team Leader
Ellen	Programmer	Large U.S.-based Company	17	Team Leader
Roger	Operations Manager	Boston-based Financial Firm	4	Project Manager
Bill	Company Owner	Independent Database Developer	19	Project Manager
Mac	CIO and CTO	Global Technology Firms	25+	Project Manager

* Number of years' experience

Table 2. Experienced System Developers (Respondents to Questionnaire for Stage 1)

Success Factors	Failure Factors
• High level of client participation • Strong executive management support • Good team member technical skills • Thorough initial requirements definition • Good project management	• Lack of requirements definition • Poor planning • Insufficient project management

Table 3. Contributing Factors Leading to Project Success or Failure

Theme 1 – Factors Contributing to SD Success or Failure. Despite the disparity in development projects and diversity of industries and roles, the participants in this study identified specific factors which contributed to either the success or the failure of projects. Based on the experts' responses, factors that were mentioned by a simple majority were considered sufficiently significant to be identified as a contributing factor. Our reasoning for a simple majority stems from the cross industry and cross project experience nature of the panel. We theorize that our holistic analysis identifies which factors can be recalled and noted as a contributor across experts and projects; therefore it is likely the factor should be considered in most projects. The contributing factors that were identified in conjunction with a project's success or failure are noted in Table 3. Analysis of these factors supports an assertion that organizational risks are of paramount importance.

Interestingly, to avoid failure, in the experiences of our respondents, most projects were not abandoned. However, factors that were identified as ones which would contribute to a failure were ameliorated through effort spent to avoid them and frame them as a contributing factor to success. For example, Steve, a senior programmer with an international transportation company, explains, "Fortunately, I haven't been on any projects that were just a bust. I have been on a number of projects that required a lot of last-minute scrambling. Primarily this has been due to lapses in the analysis and design phases." He adds, "…we've learned to dedicate more of the schedule to analysis and design. Taking the time on the front-end has helped make coding more efficient and improved our testing cycles. We have a lot less *gotchas* during the testing cycle." In this instance, identification of learned failure factors regarding planning and problem definition has been used as an impetus to avoid these contributing factors. This informant's comment is evidence of support for Risk Factors 5, 17, 18, 20, and 23 in Table 1, and suggests that developers identify those factors that contribute to project success or failure.

Theme 2 – Communication – Management Process. Our experts noted that communication between elements of the project team when it involves vendors and clients is especially critical and allows the management of the project to proceed more smoothly. In general we found that good management improves communication, and communication leads to better management. This was especially evident when problems occur, good communication is essential. For example, Roger, the operations manager of a Boston-based financial firm, explains, "Overall our projects have been very successful, with the exception of timing. The end results have always been good, but too often the time it takes to complete a project is longer than anticipated and longer than desired. … The only way to resolve a time issue is to manage your relationships and understand your leverage. Once you understand your leverage, you take control of the situation. Ultimately you need effective communication channels with your vendors and technology partners. Mutual respect and understanding play a large role in the relationship, but at the end of the day, whoever is footing the bill needs to get a return on their investment." (ROI) In Roger's case, as he puts it, to "control" the project he must maintain effective communication between parties in order to achieve project success. This finding supports the theorized risk factors found in Table 1 – Risk Factors 8 and 13, which address inter-organizational communication risks.

In contrast, the lack of the ability to effectively manage the project and communicate between the vendor and the client can have disastrous repercussions especially when communication is dependent on individual relationships. For example, Bill, an independent database developer, relates this experience – "A major national insurance company needed a sales reporting application. After full specification … and substantial progress in development, the client changed the contact person to someone totally ignorant of the types of information needed for the developer to complete the project. The project was terminated half way to completion." In this instance, the client couldn't manage the project due to the inability to communicate needed information. This evidence also confirms foundation Risk Factors #8.

Theme 3 – Project Complexity – Technical Skills – Human Resources Interaction. Complex projects require higher technical skill levels and better trained professionals. Overall, as these elements interact – resulting in projects of larger scope, numerous technical skill requirements, and flexible capabilities – the likelihood of a project failure will increase. Sometimes the project itself just proves to be too ambitious and complex. David, a software engineer in the distribution/retail industry, explained that an application solution intended to "meet cross sector (e.g., grocery, mass merchandise, specialty), cross geography (North America, South America, Europe, Asia), and even cross industry (retail/distribution, banking, insurance)" failed despite solid executive support and ample time and resources. He continued, "we simply couldn't meet the expectations." This challenge has dimensions consistent with technical risks (#5 – project technical complexity and #7 – compatibility issues between system components), but also has elements of project and team size risks, such as #9, 10, and 11 in Table 1. Ultimately, the informant did not say that the project was technically infeasible, but that they couldn't meet the expectations, supporting the notion that the problem was derived primarily from the lack of ability to respond as an organization (insufficient project team resources, etc.).

Past problems have led others to reject projects deemed too complex. Harry, the senior engineering manager at a global medical technology company, stated "I declined a project that I viewed as too complex. At the beginning of all projects, I try to map out the ending I want to have and then work backwards from there to figure out how to achieve it. If the project is so complex that I am not clear on the approach, then I won't take it on."

Others try a phased approach, allowing personnel decisions to drive solutions to other risks. Mac, the former CIO and CTO of several global technology firms with over 25 years of experience adds, "There are projects that are too complex in total, and have to be broken down into doable chunks." But when problems can be foreseen and the risks are accepted, prioritization is still important, according to Harry: "Identify and focus on the main risks early and find the necessary talent to mitigate the risks. I usually try and find multiple mitigation strategies for high risk areas and I sometimes go after these strategies in parallel until I am comfortable in a final solution."

Theme 4 – Management – End User Interaction. Whether it is an in-house development project or an outsourced project, the management team must maintain an appropriate level of interaction. With too little interaction, the end users may receive a system they can't use or refuse to learn. But with too much interaction, the users may introduce significant requests for changes not specified in the specifications ("want to haves") and the scope creep will threaten the schedule and budget. Failure of management to make decisions can influence schedule delays which can also significantly affect projects "mainly when we have a set launch date," revealed Steve. He adds, "Delays directly impact all of the phases of the development cycle, so something is going to get missed. Delays tend to lead to a 'launch and fix' approach. These decision delays also lend to 'scope creep' because requirements are still being finalized throughout the coding phase." This informant's problems are a common concern by developers and relate to the following Risk Factors in Table 1 – 13, 18, 19, 20, 22, 24, and 27. Failures in requirements analysis, accompanied by subsequent scope crepe, are endemic in large projects, and often doom such projects. Systems developers try to cope with delays by emphasizing the importance of well-understood requirements, increasing staffing and using overtime, and even continuing to work on the project without official approval. Again, the solution is often an organizational one.

Similarly, systems developers often try to be as flexible as possible with respect to client requests for changes. David accommodates changes, albeit at a price: "Though you should work hard to avoid it, you should also plan on dealing with some degree of change. We are most successful when we 'train' the customer early on to understand the impact (cost, time, etc.) of requested changes. We rarely reject a customer request for change since we are a custom services organization. It is even more rare that we accept those changes without compensation." Ellen, an experienced programmer at a large U.S.-based company, does not even charge extra – "It is part of the job. You are paid for a job, not hours." This seems to be a general trend. Ellen acknowledges that cost overruns occur, "but more and more companies are trying to accommodate the customer. It depends on the importance of the project and the customer. An important customer has a company jumping through hoops!" A minority of systems developers are less flexible, exemplified by Harry, who has "refused to incorporate changes if I felt they were unreasonable or not needed. Unless the change is absolutely necessary or easy to incorporate, I almost always reject it in order to keep the project moving along. Changing the scope of the project midstream is very damaging to the team and is something I strongly discourage."

Theme 5 – Communication – Expectations Interaction. Good communication is repeatedly mentioned in relation to managing client expectations. Harry notes, "Explicit instruction along with continued communication is very important to having a successful project … [but it is not perfect]. Providing explicit written expectations is a good start but this still leaves some room for interpretation." Communication is especially important if complexity of the project increases due to the participation of multiple parties. Greg, an enterprise architect for a state Department of Information Technology Services, remembers a project with "numerous sub-contractor development firms participating, as well as numerous client agencies participating as customers. My role was to communicate among the client agencies, which turned out to be a very demanding task. Each agency had very unique ideas on how the project should proceed, thus creating a very challenging scenario. Expectations had to be adjusted as the project progressed. This was accomplished through face-to-face negotiation with the customer agencies, as well as the development vendors." Mac encountered this in a project where "everyone had a different agenda," and this "bogged down progress to the point of the project being stopped." These two examples illustrate the expectations gap that often blocks the success of large projects, and is directly related to Risk Factors #8 (connectivity), 9 and 10 (size), 13 (communication), 17 (planning), and 23 (mismatched user expectations).

Theme 6 – Technical – Human Interaction. Technical human interaction refers to the process in which individual organizations adapt (or do not adapt) to the technologies that must be employed in the project. Ultimately, people either have to make the technology work or adopt a different technology solution. Interestingly, for our panel, technology itself is seldom seen as a problem, except when a new technology is still unstable. Additionally, hardware is never seen as a problem, and software is problematic mainly when multiple vendors are involved. The human element appears to be a primary concern. For example, Mac recommends that if you "focus on behaviour and people issues, the technology takes care of itself…" For Mac, technology interacts with the human element in a positive manner as a problem to be solved through group dynamics. This focus on human factors over technical skills is also evident in the hiring process, where personal attributes like "a cooperative attitude" (according to David) and "problem solving abilities" (according to Harry) are mentioned as frequently as technical skills. Therefore, the pairing of technical skills with critical thinking skills allows our panel members to have successful projects with minimal technology adaptation issues. This theme suggests that staffing issues may be the real source of risk, rather than the underlying technological factors that are initially apparent. Risk Factors #11 and 12 (staffing) are especially evident here, but perhaps improved leadership (#15) and planning (#17) would provide solutions to such problems. Ultimately, as with other themes, the senior manager's perspective tends to identify problems as organizational problems with organizational solutions.

Theme 7 – Impact of Day-to-Day Involvement. Becoming mired in the details of the project can be a serious risk to team motivation. The inability to see where pieces fit together can cause problems. Sometimes, some distance promotes a better general view of project progress. Though many of the anecdotes indicate a contingency approach to risk amelioration, we also noted that responses regarding what our informants deemed major causes and strategies for risk reduction differed significantly with our experts that were involved very closely with projects and senior project managers. In contrast with our previous organizationally-oriented interactions, a myopic discrete vision was revealed when asked about the biggest risks with projects. For our respondents who are more involved with day-to-day development issues, the focus is on discrete risks:

> "Not having the required skills or talent on the team" (Harry)
>
> "A lack of thorough analysis and design is probably the biggest risk with projects I've been involved with." (Steve)
>
> "One major issue: poorly defined requirements." (David)
>
> "Development cycles are shortened in order to make dates. Additional line items are necessary. Adequate machines to test on. Several times a product runs on several platforms. It is necessary to be able to verify the code works on each of these. This can be very time consuming as well as difficult to acquire the equipment." (Ellen)

These responses are in strong contrast with the organizational focus of those respondents who were overseeing multiple projects or had experience with overseeing many projects:

> "Poor planning and requirements gathering on the front end and poor communications skills during the project" (Greg)
>
> "Management commitment and people motivation" (Mac)

Similarly, risk minimization strategies differ between these two groups. Those involved with day-to-day oversight appeared to focus on specific measures:

> "Identify and focus on the main risks early and find the necessary talent to mitigate the risks. I usually try and find multiple mitigation strategies for high risk areas and I sometimes go after these strategies in parallel until I am comfortable in a final solution" (Harry)

> "Requiring finalized requirements to minimize scope creep." (Steve)

> "Invest significant time and effort working with the client to define requirements before proceeding to the implementation phase." (David)

> "Clearly understand what needs to be done prior to providing sizings." (Ellen)

Those with broader project responsibilities plan to minimize risk in more general organizational terms:

> "Planning, up-front" (Greg)

> "We don't like to be the first player on the street to sign on to any new product. We typically wait for some of our partners of industry contacts to run with the product, before we get on board." (Roger)

> "Focus on behavior and people issues; the technology takes care of itself, assuming you used an iterative development process" (Mac)

The participants' comments acknowledge that risks interact and, further, that SD managers are aware of such interactions. Their comments also revealed that risk reduction strategies tend to focus on traditional risk factors, unless the manager has broader overall project oversight responsibilities. Illustrated in Figure 1, the difference of perspectives between senior and junior SD managers may in itself present a risk factor for successful project completion. Without a common view of the problems, dealing with them becomes more difficult.

Research Stage 2 – Intra-Organizational Perspectives. Based on the results of the first stage, our informants' responses (discussed below) suggested that professionals with more experience in project leadership were more likely to view projects, and their associated risks, more holistically and assign and resolve risk as if they were organizational in nature.

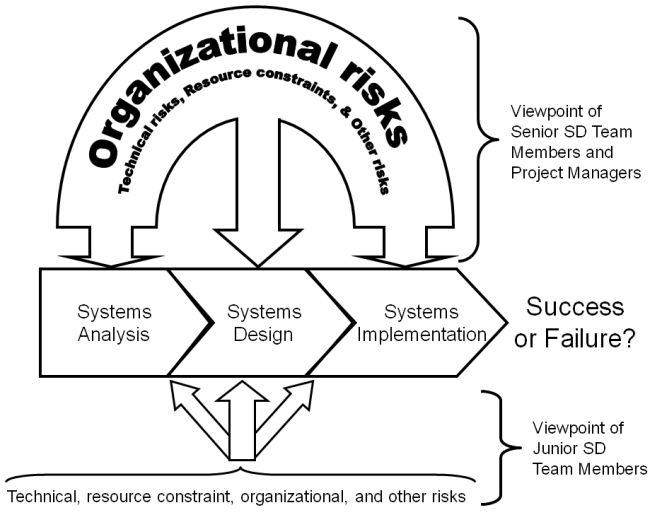

Figure 1. Comparative Perspectives on System Design Risk

Responses also suggest that culture impacted how the informants reported risk amelioration. The second stage of our research, therefore, was designed to conduct an in-depth examination of a group of SD professional aimed at exploring risk factors within a single organizational culture environment, while also exploring more deeply the impact of the various levels of project experience.

Target Organization and Respondents. Through consultation with personal contacts within the IT industry, a single organization was identified in which professionals would be most likely to participate in the second stage of the study. The organization is a technology information department at a large U.S. university. The department has a broad range of responsibilities including the development and maintenance of large and small internal projects and maintenance of legacy systems. The department employs more than one hundred full time IT professionals organized into multiple units, each dedicated to different IT functions. The organizational climate is perceived as a positive one overall (as depicted by respondents), where the participants report an 80% success rate with 20% of projects being "challenged" and very few project failures. Given recent reports concerning project failures (Standish Group International, 1995; 1999; 2001; Ewusi-Mensah, 2003), their positive assessment is justifiable. Eight key informants were either personally contacted by the authors, or contacted via referrals from contacts within the organization. The respondents represent a cross section of the department and multiple experience levels (junior, mid-level and senior mid-level). Table 4 provides a summary of the informants involved in the second research stage.

Informant	Title	Level within Organization	Current Responsibilities
#1	Systems Programmer (security related issues)	Junior	Base design and implementation
#2	Systems Programmer	Senior	Decides system configurations and which systems to develop and support
#3	Database Administrator	Junior	Learn procedures and protocols
#4	Database Administrator	Senior	Analysis, design and management of database and applications projects
#5	Programmer	Middle	Meet with end users, write code and programming
#6	Support Consultant	Middle	Meet with end users
#7	Support Consultant	Middle	Meet with end users
#8	Programmer	Junior	Implementation and maintenance

Table 4. System Developers (Respondents to Questionnaire for Stage 2)

The time for a respondent to complete the open ended questionnaires ranged from 15-45 minutes. The questionnaires, once completed, were emailed directly to the authors and identifying information was removed. The completed surveys resulted in sixteen single-spaced pages of individual anecdotes and comments comprised of more than 3600 words, which were then analyzed and decomposed by multiple authors for important clues into the perceptions of SD risk.

Questionnaire Development. To follow up on the specific findings of the first stage, a modified instrument was developed. The modified questionnaire consisted of thirteen open-ended-questions relating to project experiences, risk assessment, risk amelioration and intra-organizational communication. The instrument was first pre-tested with two external expert judges. The judges, each an academic professional with over 20 years of information systems research experience, were informed of the first stage's results and of the goals of the second stage. The instrument, which appears in Appendix B, was modified based on the comments received from the judges. Following Institutional Review Board (IRB) approval, it was emailed to Stage 2 informants. As with the Stage 1 instrument, subjects were allowed to focus on questions of their choosing, based on their relative knowledge and experience with the topics explored by various questions. (IRB requirements also mandated that all study participants are empowered to skip any part of any survey that they choose not to answer.)

Analysis of Stage Two Responses

Of the eight respondents, numbers 1, 3, and 8 were classified as junior-level professionals; respondents 5, 6, and 7 were mid-level; and respondents 2 and 4 were senior mid-level. Classification was conducted by self descriptions of responsibilities, experience, and knowledge of the organization. Each of the authors read the verbatim typed responses several times in order to get a sense of general themes within and between professionals. Then the authors compared observations and agreed that regardless of classification, the culture of the organization 1) instilled in our informants the perception that the responsibility of more senior developers is to expose junior members of the team to the big picture; and, to somewhat lesser extent, 2) encouraged the empowerment of junior-level developers to ask questions and present ideas. These perceptions directly address the SD risk factors regarding communication and leadership noted in Table 1, items 13 and 15. The basis for this organizational culture theme was formed from respondents' views of how risks are perceived and resolved as well as the communication underlying the organization. We discuss this theme in depth.

Culturally Driven Assessments. Roles and responsibilities are important factors in the development of any system. As our opening quote from FBI Director Mueller illustrates, even the highest individual is accountable when a system fails. For our target organization, the culture is one in which all developers are encouraged to understand how their work fits into the SD process. Junior and senior developers alike, when asked about the biggest risks to a project, agreed that organizational risks were paramount. For example, a programmer with eight months experience has already been acculturated to think that timing issues are paramount organizational risks:

> "The inability to meet deadlines would greatly hinder the success of our development projects. There are certain "windows of opportunity" when we may upgrade our software packages, for example. If things are not ready for production at that time, it could be six months or more before the next opportunity." (Informant # 3, consistent with Risk Factor #17 – Inadequate Planning – in Table 1)

Similarly a mid-level professional notes that:

> "Communication between clients and [the department] … can breakdown for a number of reasons. Most commonly, clients do not know what they want, or they change what they want midstream. … This prevents everyone from being on the same page." (Informant #5, consistent with Risk Factor #23 – User expectations don't match project objectives – in Table 1.)

This point is echoed by a senior mid-level developer assessing the biggest risk to project success:

> "…full comprehension of system being developed by all [areas of department]. If all groups do not have a full understanding of what is being developed throughout the life cycle of a project, it will jeopardize a successful system." (Informant #2)

Not only does this informant's comment map closely to Risk Factors #13, 15, and 24, it also closely mirrors this study's primary finding regarding big picture perspectives. This recognition that organizationally-oriented goals are paramount at all levels is likely a result of the aforementioned organizational culture of system success within this department. As Informant #8 notes, junior developers should try to "see the broad picture, know … the culture … in the organization." The emerging importance of the "big picture" orientation was assessed by our asking informants to provide us with three things senior developers should do to ensure project success. All of our informants invoked some notion that senior managers need to show the big picture and involve all member s of the team. This finding directly relates to risk factor #26 in Table 1 – lack of top management commitment to the project. From Informant #1, a junior developer:

> "The three things I have noticed most often with senior developers are a wide ranging view of the problem, expressed confidence to junior developers as to the possibility of a solution, and willingness to provide assistance (ideas not just actual fixes)."

The feeling that senior developers provide guidance as opposed to technical solutions was also articulated through the observations that a failure to clearly communicate expectations is a clear risk as noted by informant #5 as he notes that "(senior developers need to) understand what is desired by the clients and clearly communicate that to less senior employees." This supports the earlier theoretical assertion that team communications problems (#13 in Table 1) are a major SD risk factor. And there are consequences to clear communication as informant #4, a senior mid-level developer, and #7, a mid-level developer note:

> "I have seen communication problems between users and developers. Often times, the users perceive the finished product as being something different than what they get. Of course… this happens within the team as well; where management and developers think they are on the same page when actually they are not." (Informant #4)

> "One division does something and expects another to follow, but the other division is not made aware of this…It … slows down the deployment of projects. It also adds confusion to an already confusing situation." (Informant #7)

The former comment confirms risk factor # 23 (user expectations don't match project objectives) and the latter comment confirms risk factor #21 (organizational transition difficulties). These comments are especially interesting because these miscommunications are occurring in an organization that has had great success in project development, yet even here the systems development process is noted as being confusing. Other informants noted that junior developers should proactively "Ask questions when unsure" (Informants #6 & 7), and inquire about what is going on to ensure project success as noted by responses to the question "what should junior developers do:"

> "Open communication with senior staff, don't be afraid to ask questions….Get to know the end-user, how they do their jobs with your product." (Informant #3, consistent with the findings of Risk Factors 13 and 22 in Table 1.)

> "Follow organizational guidelines….Do not be afraid to express ideas that you feel are beneficial to a project. A good manager will listen to any ideas that might help a project. Not all ideas will be used; however, any idea can lead to a solution." (Informant #1, consistent with Risk Factors #12, 13, and 15.)

In summary, Stage 2 provides an exploratory assessment of risk perceptions, amelioration, and communication from the perspective of developers within an organization that has a relatively successful history of systems development projects. Our findings reveal that the type of culture found in this successful organization can empower its developers to seek "big-picture" solutions by communicating effectively with other developers and senior managers. This open environment encourages solutions that cross organizational boundaries, and can even alter perceptions of SD risk.

Conclusions

Our synthesis of the prior literature revealed interactive themes which were reinforced by the experiences and opinions of our experienced informants. The first theme is that organizational risks appear to overshadow many other risks, and that all risks might be ultimately construed as organizational risks. Stated another way, there appears to be an interactive effect such that risk factors cannot be appropriately managed without maintaining a "bird's-eye" view of the project, which inevitably requires an organizational perspective. For example, resource constraints may be alleviated by sufficient organizational commitment to the project, such that funds are properly allocated to the ailing project, or by partnering with other organizations, such as vendors or consultants.

A related theme that emerged is that past research, which has examined a single layer of IT professionals within an organization, may be masking the effects of organizational climate or culture in SD. For example, in their exploration of organizational issues, Doherty & King (1998; 2001a) implicitly recognized this phenomenon when they sampled only senior IT professionals with high levels of managerial responsibility to comment on the importance of managerial issues (see also Ropponen & Lyytinen, 2000). The importance of the cultural dimension was evident, for example, when Steve (our least experienced respondent from Stage 1, who was the most involved in day-to-day project development) noted specific traditional risks when he commented that "Our first foray into pure XHTML and CSS for the presentation layer …" was especially problematic because "…some of our initial code would completely crash some browsers." This represents a technically oriented response that suggests that Risk Factors #1, 6, and 7 (in Table 1) are also evident. By contrast, other more senior respondents recognized that such problems may be addressed by moving design team members to the appropriate projects. This seasoned approach suggests that the solutions to technological problems may actually lie in organizational responses. This is consistent with Risk Factor #12 in Table 1 – a team's lack of skill or experience.

Rather than viewing the many risks as separate or discrete categories of risks, senior project managers, with their more comprehensive organizational perspective, are more likely than junior managers to view all risks as essentially organizational in nature. Having this "big picture" perspective is critical to the success of all systems development projects, especially those that are large and complex. They may be related to personnel issues (such as staffing), organizational structure issues (reporting relationships), political issues (top management support), inter-personal issues (design team compatibility), sociological issues (corporate culture clashes between outsourcer and client), or psychological issues (motivational factors for designers). These themes also emerged through the experiences of our survey respondents. However, in Stage 2 we attempted to control for the cross-industry effect and focused on a single organization that has experienced a successful history of system development projects. The ensuing comments from our informants potentially place a huge contingency on this finding.

While there is an intuitive appeal of saying that more experienced project managers are more likely than lower level developers to be concerned with the big picture, we saw that for a successful firm this was not the case. Rather, we found that the organizational culture drives what is considered important, which for the organization in Stage 2 was communication and interpersonal relationships.

The present study illuminates the existing theoretical foundations regarding SD risk by exploring the detailed and voluminous responses provided by two subsequent groups of experienced system developers and IT professionals. While the respondents provided rich and fertile data for the examination of SD risks, future studies should seek to increase the sample size to enable quantitative assessments in addition to the qualitative assessments provided here. In addition to the limitation posed by the number of study participants, future research should pursue the involvement of a greater number of organizations. Both constraints serve as limitations on the overall generalizability of the present study. Further research should be undertaken to discover ways to integrate the perspectives of less experienced, lower-level personnel with those of senior, upper-level IT professionals, and to study how these different views are managed to foster a more cohesive approach to risk management. Finally, the examination of firms

along the organizational culture continuum is needed to more fully understand SD risk and amelioration.

The findings of our two-staged exploratory studies provide normative guidance to both junior and senior developers alike. These findings provide rich qualitative, experiential evidence in support of the theories advanced by other recent studies – namely, that the organizational aspects of SD risk are the greatest source of failure of SD projects, and that organizational risks effectively subsume the other categories of risk. For example, the participants in the Stage 1 study confirm that most technical challenges can be overcome with proper managerial processes and sufficient commitment of resources, both of which are organizational responses, as we illustrate in Figure 1. As another example, strong organizational support for a project, coupled with compromises and creative solutions, can usually resolve resource constraint issues. Further, our findings indicate that SD risk can be ameliorated by an organizational culture which maintains clear and open communication channels and empowers its developers to ask questions and seek "big-picture" solutions. Such a culture instills a mindset that enables its developers to recognize and successfully solve SD problems when they arise.

References

Al-Mushayt, O., Doherty, N. and King, M. (2001) An investigation into the relative success of alternative approaches to the treatment of organizational issues in systems development projects. *Organization Development Journal* 19(1), 31-48.

Al-Shehab, A.J., Hughesm R.T. and Winstanley, G. (2005) Modelling risks in IS/IT projects through causal and cognitive mapping. *The Electronic Journal of Information Systems Evaluation* 8(1), 1-10.

Alter, S. (1979) Implementation risk analysis. *TIMS Studies in Management Sciences* 13(2), 103-119.

Alter, S., and Ginzberg, M. (1978) Managing uncertainty in MIS implementation. *Sloan Management Review* 20(1), 23-31.

Armour, P.G. (2005) To plan, two plans: A planning approach to managing risk. *Communications of the ACM* 48(9), 15-19.

Barki, H., Rivard, S., and Talbot, J. (1993) Toward an assessment of software development risk. *Journal of Management Information Systems* 10(2), 203-225.

Barki, H., Rivard, S., and Talbot, J. (2001) An integrative contingency model of software project risk management. *Journal of Management Information Systems* 17(4), 37-69.

Bednarz, A. (2002) *IT Malpractice*, available at http://www.nwfusion.com/careers/2002/0408man.html, accessed October 3, 2002, Network World Fusion.

Clegg, C., Axtell, C., Damodaran, L., Farbey, B., Hull, R., Lloyd-Jones, R., Nicholls, J., Sell, R, and Tomlinson, C. (1997) Information technology: A study of performance and the role of human and organizational factors. *Ergonomics* 40(9), 851-871.

Doherty, N.F. and King, M. (1998) The consideration of organisational issues during the systems development process: An empirical analysis. *Behaviour & Information Technology* 17(1), 41-51.

Doherty, N.F. and King, M. (2001a) An investigation of the factors affecting the successful treatment of organisational issues in systems development projects. *European Journal of Information Systems* 10(4), 147-160.

Doherty, N.F. and King, M. (2001b) The treatment of organisational issues in systems development projects: The implications for the evaluation of information technology investments. *The Electronic Journal of Information Systems Evaluation* 4(1), 1-14.

Doherty, N.F. and King, M. (2003) From technical change to socio-technical change: Towards a proactive approach to the treatment of organisational issues. In *Socio-technical and human cognition elements of information systems* (CLARKE S, COAKES E, HUNTER G and WENN A, Eds), pp 22 - 40, Idea Group Publishing, Hershey, PA.

Doherty, N.F. and King, M. (2005) From technical to socio-technical change: Tackling the human and organizational aspects of systems development projects. *European Journal of Information Systems* 14, 1-5.

Doherty, N.F. and King, M., and Al-Mushayt, O. (2003) The impact of inadequacies in the treatment of organizational issues on information systems development projects. *Information & Management* 41(1), 49-62.

Ewusi-Mensah, K. (2003) *Software development failures.* The MIT Press, Cambridge, MA.

Ewusi-Mensah, K. and Przasnyski, Z.H. (1994) Factors contributing to the abandonment of information systems development projects. *Journal of Information Technology* 9(3), 185-201.

Faraj, S. and Sambamurthy, V. (2006) Leadership of information systems development projects. *IEEE Transactions of Engineering Management* 53(2), 238-249.

Gotterbarn, D. and Rogerson, S. (2005) Responsible risk analysis for software development: Creating the software development impact statement. *Communications of the AIS* 15, Article 40, 730-750.

Haber, L. (2003) *How do you measure ITSM success?*, available at http://itmanagement.earthweb.com/service/print.php/3291421, accessed April 2, 2004, Jupitermedia Corporation.

Hayes, L. (2005) *Development study: Haste makes waste*, available at http://www.computerworld.com/developmenttopics/development/story/0,10801,104846,00.html, accessed October 6, 2005, Computerworld.

Hornby, C., Clegg, C., Robson, C., McClaren, J., Richardson, C., and O'Brien, P. (1992) Human & organisational issues in information systems development. *Behaviour & Information Technology* 11(3), 160-174.

Jiang, J.J., Klein, G., and DIscenza, R. (2001) Information systems success as impacted by risks and development strategies. *IEEE Transactions on Engineering Management* 48(1), 46-55.

Keil, M., Tan, B.C.Y., Wei, K., Saarinen, T., Tuunainen, V., and Wassenaar, V. (2000) A cross-cultural study on escalation of commitment behavior in software projects. *MIS Quarterly* 24(2), 299-325.

Keil, M., Im, G.P., and Mahring, M. (2007) Reporting bad news on software projects: The effects of culturally constituted views of face-saving. *Information Systems Journal* 17(1), 59-87.

Laudise, T.M. and Nuara, L.T. (2002) How to contract for a successful e-commerce development project: Beating the odds. *The Business Lawyer* 58, 299-316.

Lucas, H.C. (1975) *Why information systems fail.* Columbia University Press, New York.

Lyytinen, K., Mathiassen, L. and Ropponen, J. (1998) Attention shaping and software risk - a categorical analysis of four classical risk management approaches. *Information Systems Research* 9(3), 233-255.

Lyytinen, K. and Robey, D. (1999) Learning failure in information systems development. *Information Systems Journal* 9(2), 85-101.

McFarlan, F.W. (1981) Portfolio approach to information systems. *Harvard Business Review* 59(5), 142-150.

McGraw, G. (2002) Managing software security risks. *IEEE Computer* 35(4), 99-101.

Morgan, R.M. and Hunt, S.D. (1994) The commitment-trust theory of relationship marketing. *Journal of Marketing* 58(3), 20-38.

Nelson, R.R. (2007) IT project management: Infamous failures, classic mistakes, and best practices. *MIS Quarterly Executive* 6(2), 67-78.

Ropponen, J. and Lyytinen, K. (2000) Components of software development risk: How to address them? A project mananger study. *IEEE Transactions on Software Engineering* 26(2), 98-112.

Schmidt, R., Lyytinen, K., Keil, M., and Cule, P. (2001) Identifying software project risks: An international Delphi study. *Journal of Management Information Systems* 17(4), 37-69.

Segars, A.H. and Grover, V. (1996) Designing company-wide information systems: Risk factors and coping strategies. *Long Range Planning* 29(3), 381-392.

Sherer, S.A. and Alter, S. (2004) Information system risk and risk factors: Are they mostly about information systems? *Communications of the AIS* 14, 29-64.

Snow, A.P., Keil, M., and Wallace, L. (2007) The effects of optimistic and pessimistic biasing on software project status reporting. *Information and Management* 44(2), 130-141.

Songini, M. (2005) Software ownership battle adds $10m to cost of 'Big Dig'. *Computerworld* 39(9), 16.

Standish Group International. (1995) *The chaos report (1994)*, available at http://www.standishgroup.com/sample_research/PDFpages/chaos1994.pdf, accessed March 23, 2004, The Standish Group International.

Standish Group International. (1999) *Chaos: A recipe for success (1999)*, available at http://www.standishgroup.com/sample_research/PDFpages/chaos1999.pdf, accessed March 23, 2004, The Standish Group International.

Standish Group International. (2001) *Extreme chaos (2001)*, available at http://www.standishgroup.com/sample_research/PDFpages/extreme_chaos.pdf, accessed March 23, 2004, The Standish Group International.

Standish Group International. (2003) *Chaos Chronicles, Version 3.0*, available at http://www.standishgroup.com/chaos_resources/chronicles.php, accessed November 15, 2007, The Standish Group International.

Stephen, D., Keil, M., Mathiassen, M., Shen, Y., and Tiwana, A. (2007) Attention-shaping tools, expertise, and perceived control in IT project risk assessment. *Decision Support Systems* 43(1), 269-283.

Tiwana, A. and Keil, M (2004) The one-minute risk assessment tool. *Communications of the ACM* 47(11), 73-77.

Tiwana, A. and Keil, M. (2006) Functionality risk in information systems development: An empirical investigation. *IEEE Transactions on Engineering Management* 53(3), 412-425.

Wallace, L.G. (1999) *The development of an instrument to measure software project risk*. Ph.D Thesis, College of Business Administration, Georgia State University.

Wallace, L.G., Keil, M., and Rai, A. (2004) Understanding software project risk: A cluster

analysis. *Information and Management* 42(1), 115-125.
Webster, B.F. (2000) Patterns in IT litigation: Systems failure (1976-2000). A Study by Pricewaterhouse Coopers LLP.

About the Authors

Merrill Warkentin is a Professor of MIS at Mississippi State University. His research, primarily in computer security management, eCommerce, and virtual collaborative teams, has been published in numerous books and in journals such as *MIS Quarterly, Decision Sciences, Decision Support Systems, Communications of the ACM, Communications of the AIS, Information Systems Journal, Information Resources Management Journal, Journal of Organizational and End User Computing, Journal of Global Information Management*, and others. He has served as Guest Editor or Associate Editor for a number of journals, including *MIS Quarterly, European Journal of Information Systems, Journal of Organizational and End User Computing*, and *Information Resources Management Journal*. He has recently chaired several global conferences on computer security, including WISP and the IFIP Workshop on Information Security. His most recent book is *Enterprise Information Systems Assurance and System Security: Managerial and Technical Issues* (2006). He earned his PhD from the University of Nebraska-Lincoln.

Robert S. Moore is an Associate Professor of Marketing at Mississippi State University. His research interests focus on the adoption and use of technology and consumer behavior. He has had his research published in outlets such as *The Journal of Business Research*, *Journal of Advertising*, *Journal of Interactive Marketing*, *Journal of End User Computing*, *Journal of Public Policy and Marketing*, and the *Marketing Management Journal*. He earned his PhD at the University of Connecticut.

Ernst Bekkering is an Assistant Professor in Information Systems at Northeastern State University, where his research interests include Human-Computer Interaction, Information Assurance and Security, Telecommunications, and Electronic Commerce. He earned his MSIS and PhD degrees in Information Systems at Mississippi State University. His dissertation "Visual Angle in Videoconferencing: The issue of Trust" showed that camera position in electronic communications can negatively influence trust perceptions. Dr. Bekkering has co-authored articles in *Communications of the ACM*, *Journal of Organizational and End User Computing*, the *Journal of Digital Forensics, Security and Law*, and *Information Resources Management Journal*, and has presented papers at AMCIS, IRMA, and DSI conferences.

Allen C. Johnston is an Assistant Professor in the School of Business at the University of Alabama at Birmingham. He holds a BS from Louisiana State University in Electrical Engineering as well as an MSIS and PhD in Information Systems from Mississippi State University. His works can be found in such outlets as *Communications of the ACM*, *Journal of Global Information Management*, *Journal of Organizational and End User Computing*, and *Journal of Information Privacy and Security*. The primary focus of his research has been in the area of information assurance and computer security, with a specific concentration on the behavioral aspects of information security and privacy. He has also served as guest speaker and provided consultation services to numerous entities including Regions Financial Corporation, the Birmingham Chapter of the Institute of Management Accountants, and the National Decision Sciences Institute.

Appendix A : Instrument for Research Stage 1 – Cross-Industry Perspective (Open-ended questions, with respondent instructions omitted for brevity)

Section 1 – How shall we refer to you?

Section 2 – Your Professional Experience

1. How many years have you been involved in system development or IT management?
2. What has been your role in most of the system development projects you've worked on?
3. Do you have a particular expertise – do you work mostly on certain kinds of projects?
4. Have your projects been mostly large projects with large budgets and many members of the team, or smaller projects (single contractor or small team)?

Section 3 – Past Projects

1. We understand that what you tell us is limited by your obligations of confidentiality, but can you briefly describe a couple of the most successful projects you've worked on, focusing on the factors that made them successful?
2. Now can you describe a couple of projects that were not very successful, focusing on the major problems with those projects?
    - How were those problems resolved?
    - What is the current status of the project?
3. In cases where you have had problems during a project, how have delays (i.e., in information, specifications, approvals, etc.) been a source of problems?
    - Were those delays caused by the developer or by the client?
    - How were those issues resolved?
4. Do you recall instances in which there had been a mismatch of expectations? In other words, after a project gets under way, did it ever become clear that the client was expecting more from the developer than the developer realized?
    - Was this the result of a failure to communicate?
    - Do you wish the legal contract had been more explicit regarding the client's expectations?
5. In your opinion, do problems with a project tend to stem from one major issue (whatever that issue may be) or from an accumulation of minor issues?

Section 4 – Project Risks

1. In your experience, what would you say are the biggest risks with projects?
2. Overall, how do you try to minimize risks in connection with system development projects?
3. Do you have insurance for those risks (such as an "Errors and Omissions" policy)?
4. What percentage of your firm's development projects are performed under a written contract?
    - How adequately are responsibilities & expectations (of you & the other party) described?
    - How extensively are the terms of those contracts negotiated? Which terms are negotiated the most vigorously?

Section 5 – The Changing Project

1. Can you recall any projects in which requested changes during development significantly expanded the scope of the project?
    - Have you (or has a developer) ever refused to incorporate changes because they were either infeasible or unreasonable?
    - If the scope was expanded, were you (or was the developer) appropriately compensated for the additional work?
    - Have changes to projects been rejected because both sides could not reach agreement about additional fees for the additional work?
2. How frequently have cost overruns been the reason for canceling a project?
    - What was the nature of the cost overruns?

Section 6 – The Cancelled Project

1. Has a project you've been involved with been cancelled after work began on it?

- What were the stated reasons for canceling the project?
- Do you think those were the real reasons? If not, what do you think the reasons were?

Section 7 – Communication During Projects

1. Could you provide an example of when you felt that there wasn't enough communication concerning new developments or expectations for a project (e.g., between team members, between the team and managers, or between the firm and client)?
   - How did this lack of communication affect the project?
   - Did that lack of communication affect future projects?
2. Could you provide an example of when you felt that you did not have enough information to make decisions regarding a project?
   - How did this lack of information affect the project?
   - Did that lack of information affect future projects?
3. Could you provide an example of when you perceived a lack of trust regarding a project (e.g., between team members, between the team and managers, or between the firm and client)?
   - How did this lack of trust affect the project?
   - Did that lack of trust affect future projects?
4. How would you describe the cooperation atmosphere for most projects (e.g., between team members, between the team and managers, or between the firm and client)?
   - Has a lack of cooperation affected any projects you may recall?
   - Did that lack of cooperation affect future projects?
5. Could you provide an example of when you felt conflicts between departments or individuals have affected a project's success or implementation?
   - How did this conflict affect the project?
   - Did that conflict affect future projects?

Section 8 – For Systems Developers Only

1. Has a client ever claimed that the system you delivered failed to perform properly?
   - If so, what do you think was the source of their dissatisfaction? Was that reasonable?
   - Did the client express its dissatisfaction *during development* or *after* the final delivery?
   - Do you offer a warranty for your systems? How long does it last?
   - If the client expressed dissatisfaction after final delivery, was it before the warranty expired?
   - How was the issue resolved? Was the contract invoked in order to resolve the issue?
2. Has your firm been sued for breach of contract?
   - If so, what was the alleged breach?
   - Did you retain a lawyer?
   - How was the matter resolved?
3. Have you ever sued a *client* for breach of contract?
   - If so, what was the alleged breach?
   - Did you retain a lawyer?
   - How was the matter resolved?

Section 9 – Risk Factors

1. In your experience, do projects developed for different industries have different probabilities of project failure?
2. In your experience, do projects developed for different organizations have different probabilities of project failure? Does the size or type of the organization matter?
3. In your experience, what constitutes the size of the project? Dollar amount? Number of people on the team? Number of software components? Other?
4. Does the probability of project failure change with the size of the project?
   - Is the probability of failure higher in small or in large projects?
5. In your experience, does the probability of project failure depend on the complexity of the project?
   - Can you give some examples of factors that have contributed to project complexity?
   - If you anticipated difficulties due to project complexity, how did you try to limit those problems?
   - Have you ever declined to accept a project because it was too complex?

    - What contributed the most to the complexity?
    - Would you happen to know how the team that did accept the project fared, and how they dealt with the complexity of the project?
6. In your experience, does team familiarity with the technology influence the probability of project failure?
    - Can you give some examples of technologies that have proven especially problematic?
    - If you anticipated problems due to lack of familiarity with technology, have you ever sought to add expertise to the team (contractors, temporary staff, new staff, consultants)?
    - If you anticipated problems due to lack of familiarity with technology, have you ever chosen to use a different (more familiar) technology?
7. In your experience, are there factors that necessitate extra attention to end user involvement in the project?
    - Can you give an example?
8. In your experience, are there factors that necessitate extra attention to definition of system requirements?
    - Can you give an example?
9. When you hire new staff, what qualifications do you look for?
10. Have you ever had to remove a team member from a project? What happened?
11. When you look for new projects, which type of project do you prefer?
12. How would former clients describe your expertise, your strengths, and your weaknesses?
13. Who is your strongest business competitor? What is their strength?
14. Before you start a project, how do you decide on the level of communication (frequency, form) with the client during the project?
15. Have you seen projects where the systems development methodology did not seem to "fit" the project?
16. Have you experienced obstacles to project success related to the introduction of new or "bleeding edge" technology?
    - Were the problems associated with a major technological shift during the project?
17. Have you experienced obstacles to project success related to the acquisition and installation of new hardware?
    - Were the problems associated with hardware compatibility issues?
    - Were the problems associated with a lack of familiarity with the hardware?
    - Were the problems associated with the complexity of the hardware?
    - Were the problems associated with the number of hardware vendors involved in the project?
18. Have you experienced obstacles to project success related to the acquisition and installation of new software?
    - Were the problems associated with software compatibility issues?
        - Were the compatibility issues OS related?
        - Were the compatibility issues application related?
    - Were the problems associated with a lack of familiarity with the software?
    - Were the problems associated with the complexity of the software?
    - Were the problems associated with the number of software vendors involved in the project?
19. Have you experienced obstacles to project success related to the obsolescence of hardware or software?
20. Have you experienced obstacles to project success related to communication protocol incompatibility?
21. Have you experienced obstacles to project success related to component reliability?
22. Have you experienced obstacles to project success related to performance shortfalls?
23. Have you experienced obstacles to project success related to technical interfaces?
24. Have you experienced obstacles to project success related to defects in quality?
25. Have you experienced obstacles to project success related to new and untried methods or technologies?
26. Have you experienced obstacles to project success related to expense (financial burden)?
    - Were the expenses related to initial costs of hardware or software?
    - Were the expenses related to maintenance costs of hardware or software?
27. Have you experienced obstacles to project success related to technical specification problems?
28. Have you experienced obstacles to project success related to technical security problems?
    - Were the security problems related to buffer overflows?
    - Were the security problems related to access control?
    - Were the security problems related to cryptography?
    - Were the security problems related to input validation?
    - Were the security problems related to passwords?

Appendix B – Instrument for Research Stage 2 – Intra-Organizational Perspectives

Section 1 – How shall we refer to you?

As we write up our findings for publication, we may use your views, perspectives, or anecdotes in our analysis, but your identity will be masked. Please indicate the manner in which you would like to be described in our manuscript. For example, individuals may want to be known as "a senior software development manager with twenty years experience," or "a third-year programmer/analyst at a large state university in the US." How would you prefer we refer to you?

Section 2 - Project Experiences

1. Please think about development projects that you would consider **successful**. Could you please describe what your role was within these successful projects?

2. Please think about development projects that you would consider **not** so successful. Could you please describe your role within these **not** so successful projects?

3. Consider developers in your organization who have **less** experience than you, or who are more junior. Can you briefly describe a few differences in your development responsibilities versus those less experienced developers?

4. Now consider developers in your organization who have **more** experience than you, or who are more senior. Can you briefly describe a few differences in your development responsibilities versus those more experienced developers?

Section 3 – Risks

1. What are the biggest risks to the success of systems development projects in your organization?

2. What do more junior developers in your organization view as the biggest risks?

3. What do more senior developers in your organization view as the biggest risks?

Section 4 – Risk Amelioration

1. Do you believe that most junior developers apply short term "bandaids" to solve problems, or do they use a long term problem solving approach? Can you give us an example?

2. Do you think that most senior developers take a longer-term, broader approach to target underlying causes of problems rather than simply treat symptoms? Can you give us an example?

3. Based on your experiences, can you provide us with 3 things that junior developers should do to ensure project success.

4. Based on your experiences, can you provide us with 3 things that senior developers should do to ensure project success.

Section 5 – Communication During Projects

1. Could you provide an example of when you felt that there wasn't enough communication concerning new developments or expectations for a project (e.g., between team members, between the team and managers, or between ITS and users)?
    - How did this lack of communication affect the project?
3. Could you provide an example of when you perceived a lack of trust regarding a project (e.g., between team members, between the team and managers, or between the firm and client)?

    How did this lack of trust affect the project?

Thank you for your time. Responses to this survey are strictly confidential. Please save this file and email it back to us. Again, Thank You.