An Integrative Contingency Model of Software Project Risk Management

HENRI BARKI, SUZANNE RIVARD, AND JEAN TALBOT

HENRI BARKI is Professor in the Information Technologies department and Director of Research at the Ecole des Hautes Etudes Commerciales (HEC) in Montréal. His research interests focus on information technology implementation and software development project management. Journals where his publications have appeared include *Annals of Cases on Information Technology Applications and Management in Organizations, Canadian Journal of Administrative Sciences, IEEE Transactions on Professional Communication, Information Systems Research, Information & Management, INFOR, Journal of Management Information Systems, Management Science,* and *MIS Quarterly.*

SUZANNE RIVARD is Professor of Information Technology at the Ecole des Hautes Etudes Commerciales (HEC) in Montréal. She holds an M.B.A. from HEC and a Ph.D. from the Ivey School of Business, the University of Western Ontario. Her research interests focus on software project risk management, outsourcing of information systems services, and the use of IT in emerging organizational forms. She has published in *Communications of the ACM, Decision Support Systems, Information and Management, Interfaces, Journal of Management Information Systems, MIS Quarterly,* and *Omega.*

JEAN TALBOT is Associate Professor of Information Technology at the Ecole des Hautes Etudes Commerciales (HEC) in Montréal. He received his doctorate in information systems from the Université Montpellier II in France. His current research interests are in system development management, e-business, and e-commerce. Journals where his publications have appeared include *Canadian Journal of Administrative Sciences, Journal of Management Information Systems,* and *MIS Quarterly.*

ABSTRACT: Drawing both from the IS literature on software project risk management and the contingency research in Organization Theory literature, the present study develops an integrative contingency model of software project risk management. Adopting a profile deviation perspective of fit, the outcome of a software development project (*Performance*) is hypothesized to be influenced by the fit between the project's risk (*Risk Exposure*) and how project risk is managed (*Risk Management Profile*). The research model was tested with longitudinal data obtained from project leaders and key users of 75 software projects. The results support the contingency model proposed and suggest that in order to increase project performance a project's risk management profile needs to vary according to the project's risk exposure. Specifically, high-risk projects were found to call for high information processing capacity approaches in their management. However, the most appropriate management approach was found to depend on the performance criterion used. When meeting project budgets was the performance criterion, successful high-risk projects had high levels of internal integration, as well as high levels of formal planning. When system

quality was the performance criterion, successful high-risk projects had high levels of user participation.

KEY WORDS AND PHRASES: contingency models, software project management, software project risk.

IN A TECHNOLOGY-DRIVEN AREA OF STUDY such as Information Systems (IS), the half-life of research topics can be relatively short. Some topics, however, stay relevant over time, indicating the importance and complexity of the issue they are addressing. Software project risk management is such a topic. The well-documented difficulties early software projects experienced in meeting their objectives [10, 34]—be they related to budget, schedule, or product quality—have remained current throughout the eighties [31]. Even in this day and age of enterprise resource planning systems, such difficulties and project failures are still common [20, 22]. Not only has the issue of software risk management been researched for more than two decades now, but, as years go by, it appears to have attracted the attention of an increasing number of researchers.

Over the years, several aspects of software project risk management have been studied using a variety of approaches. Keil et al. [25] noted that "Since the 1970's, both academics and practitioners have written about risks associated with managing software projects. . . . Unfortunately, much of what has been written on risk is based either on anecdotal evidence or on studies limited to a narrow portion of the development process" (p. 77). In a recent literature survey, Ropponen [45] identified 34 empirical studies on software risk management published during the 1978–1999 period and analyzed their content, research purpose, research time frame, theoretical foundations, and research approach. This analysis revealed an area rich in terms of approaches, ranging from action research and case studies to survey and laboratory experiments, as well as in terms of its theoretical foundations, ranging from structural contingency theory and theories of organizational change to bounded rationality and prospect theory.

Despite these strengths of past research, certain weaknesses were also identified. One weakness stems from the fact that whereas software risk management studies have examined various aspects of software risk management (e.g., the concept of risk itself and its antecedents, risk analysis techniques, risk management heuristics, risk resolution techniques, management interventions, or achievement of aspiration levels), "the connections between the different research constructs have been weakly examined" ([45], p. 222). Another weakness is related to the research purpose (i.e., discovery versus testing) of the studies examined. Over two-thirds of the 34 surveyed articles had a discovery rather than a testing focus. Moreover, only five articles, drawn from two studies, tested a priori hypotheses that incorporated more than two variables related to software risk management [40, 41, 42, 46, 47]. Another important weakness pertains to the dominance of a single data collection period, with 25 of the 34 studies having collected data at a single point in time [45].

The diversity of software risk management studies and their associated weaknesses point to a relatively dispersed literature where empirical studies of a more integrative nature are needed. The present study is an effort in that direction. Adopting an information processing view of organizations and drawing both from research in software project risk management and from the contingency perspective research stream in Organization Theory, this paper develops an integrative contingency model of software project risk management. The central hypothesis of the model is that the performance of a software development project is influenced by the fit between the project's degree of risk exposure and its project management profile. Using a profile deviation perspective of fit, this hypothesis is tested for a specific project management profile, one that is described by the levels of internal integration, formal planning, and user participation employed in managing a project. To test the study hypothesis, data from 75 software development projects were collected from two respondents per project (i.e., the project manager and a user representative), and over two data collection periods (i.e., during development and a few months after implementation). Both by its objectives and research design, the present study attempts to address the weaknesses of past research. In particular, the constructs of the proposed contingency model of software project risk management represent a synthesis of past research. Moreover, the examination of relationships between more than two software project risk management concepts (i.e., project risk exposure, project risk management, and project performance), having testing rather than discovery as the study's research purpose, and the use of longitudinal data should also address some important limitations of past research.

The next section of the paper reviews prior work in IS research that adopted a contingency view of software project risk management and on which the present research model is based. This is followed by the presentation of the research model, the description of the study's methodology, and its results. The last section discusses the results obtained, and concludes with their implications for research and practice.

The Contingency Perspective in Software Project Risk Management

IS RESEARCHERS ADOPTING THE CONTINGENCY APPROACH to software project risk management have been strongly influenced by research in organizational contingency theory. Organizational contingency theorists, following Burns and Stalker [12], proposed that successful organizations establish a fit between the degree of uncertainty of their environment and their structural and process characteristics [9, 16, 29, 37, 38, 49]. In this work, environmental uncertainty (reflected by such factors as environmental complexity, the environmental rate of change, and the availability and clarity of information) is said to require more organic structures, more expert-based power, less centralization of authority [12, 38, 48], less formal planning [38], and more liaison devices [17, 39].

The relevance of this research stream to software project risk management has been recognized by a number of IS researchers, who adopted such a contingency

perspective. According to this view, software development projects managed with approaches that fit the demands imposed by the degree of risk or uncertainty of the project's environment will be more successful than projects that do not. For instance, the importance of having high levels of integration in uncertain environments but lower levels in more certain environments has been consistently stressed and empirically supported in Organization Theory literature [18, 29, 38, 50]. Extending this reasoning to software development projects, high–risk exposure projects can be thought to require high levels of integration, whereas low levels of integration would be appropriate for low–risk exposure projects [6, 34, 55]. Similarly, research in Organization Theory also recommends the use of low levels of formal planning in high-uncertainty environments [38]. This argument stems from the idea that rigidities inherent to high levels of formal planning decrease an organization's ability to adapt to external changes associated with uncertain environments. In the context of software development, too much emphasis on the use of formal planning tools is thought to be inappropriate for high risk exposure projects since the information needed for planning is often unavailable, and key elements of the project are not well understood. In contrast, employing formal planning tools is seen as useful in low–risk exposure projects because they can help structure the sequence of tasks in addition to providing realistic cost and time targets [2, 34].

Table 1 summarizes past IS research that adopted such a contingency approach to software project risk management. As shown in the table, different authors have suggested different terms for project management approaches considered more appropriate for a given level of project risk. However, upon closer examination, the proposed models are seen to share a common perspective of organizations—namely, the information processing view first advanced by March and Simon [33], and developed further by Galbraith [17, 19]. While the information processing perspective was first used to describe organizations rather than groups, it has been shown to be relevant to understanding and managing smaller units, such as teams [26, 30].

The basic proposition of the information processing view is that, "If the task is well understood prior to its performance, much of the activity can be preplanned. . . . The greater the task uncertainty, the greater the amount of information that must be processed among decision makers during task execution in order to achieve a given level of performance" ([19], p. 36).

Rearranging the software project risk management practices reported in Table 1 in terms of their information processing capacity (see Table 2), low information processing capacity management approaches are seen to mostly rely on formal controls, policies, plans, milestones, rules, status reports, and standards. On the other hand, high information processing capacity management approaches tend to emphasize communications among the actors involved in a project—be they team members, managers, or users. As can be seen from Tables 1 and 2, there is much convergence in the recommendations made by different authors with respect to the level of information processing capacity appropriate for a given level of project risk or uncertainty. In essence, low information processing capacity approaches have been associated with low-risk projects, and vice versa. However, few published studies provide empirical

Table 1. Contingency Approaches to Software Project Risk Management

Project Management Construct Studied	Author	Recommended Course of Action for Low Project Risk or Uncertainty	Recommended Course of Action for High Project Risk or Uncertainty	Support Provided
Project Management Activities (16 activities grouped into 4 categories)	Alter [1], Alter and Ginzberg, [2]		Divide project, keep solution simple, develop good support base, meet user needs.	Cross-sectional data from 85 projects
Project Coordination Mode	Zmud [55]	Use impersonal mode of coordination.	Use group mode of coordination.	Conceptual
Internal Integration, Formal Planning, and Formal Control	McFarlan [34]	Depending on whether risk is due to project technology, structure, or size, use low or high levels of each.	Depending on whether risk is due to project technology, structure, or size, use low or high levels of each.	Conceptual
Relationship Between Parties Concerned	Beath [5, 6]	Use arm's length relationships.	Use clan relationship.	Case studies
Project Management Tools	Kydd [27]		Use prototyping, formal specifications and documentation, and structured walkthroughs.	Conceptual
Project Risk Management Steps	Boehm [7, 8]		Carry out risk planning, resolution, and monitoring.	Conceptual
User Participation	McKeen et al. [35, 36]	Use low levels of user participation.	Use high levels of user participation	Cross-sectional data from 151 projects
Coordination Structure	Nidumolu [40, 41]	Use vertical coordination	Use vertical coordination	Cross-sectional data from 64 projects

Table 2. Information-Processing Capacity of Software Project Risk Management Approaches

Author	Low Information-Processing Capacity	High Information-Processing Capacity
Alter [1], Alter and Ginzberg [2]	Formal planning	User participation, prototyping, evolutionary approach, training programs, ongoing assistance
Zmud [55]	Impersonal mode of coordination	Personal mode of coordination Group mode of coordination
McFarlan [34]	Formal planning Formal control	Internal integration
Beath [5, 6]	Arm's length strategy	Matrix strategy
Kydd [27]		Prototyping, formal specifications, documentation, structured walkthroughs
Boehm [7, 8]		Prototyping, user surveys, and user participation
McKeen et al. [35, 36]	Low levels of user participation	High levels of user participation
Nidumolu [40, 41]	Vertical coordination mode	Horizontal coordination mode

evidence for the proposed models. With the exception of [35, 36, 40, 41, 42], contingency studies of software project risk management are essentially conceptual articles or single case studies. Thus, even though the utility of applying the contingency approach to software project risk management appears to have gained some acceptance, it has done so without strong empirical verification.

An Integrative Contingency Model of Software Project Risk Management

FIGURE 1 PRESENTS A CONTINGENCY MODEL of software project risk management that summarizes and integrates the above points. The central hypothesis of Figure 1 can be summarized as:

> The better the fit between the level of risk exposure of a software project and its management profile, the higher the project's performance.

Performance

This construct refers to the efficiency and effectiveness with which a software development project was completed, and takes into account two key dimensions [14, 23,
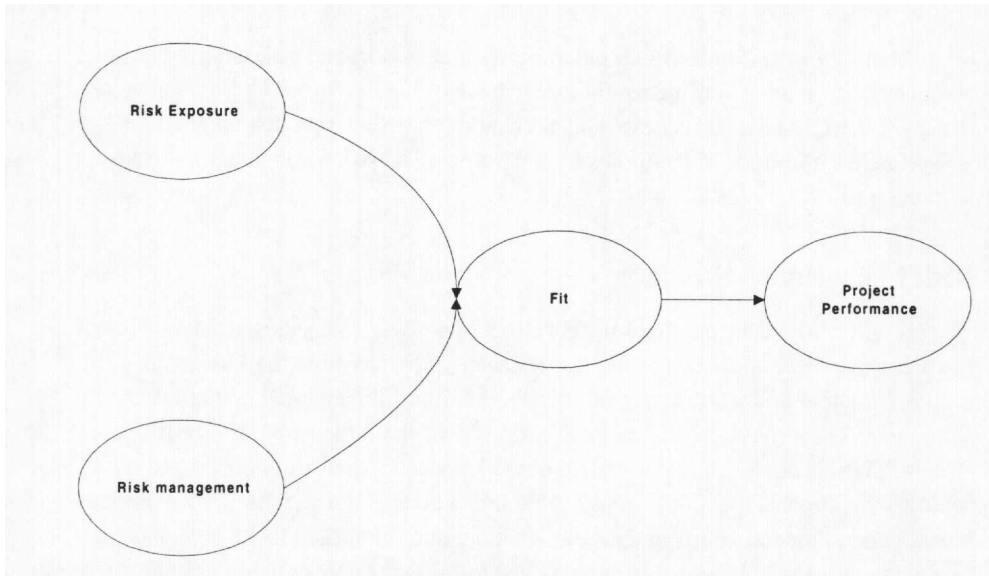
Figure 1. General Contingency Model of Software Project Risk Management

40]: process performance (how well the process of software development and the project went) and product performance (how good the developed system, that is, the product or output of that process, is). These two dimensions need to be assessed separately since they are not necessarily highly correlated. For example, it is quite possible for an over-budget or beyond-schedule project to deliver a high-quality product. Conversely, a within-budget and on-time project may deliver a product of poor quality [43].

Risk Exposure

Many IS researchers examining software development project management with a contingency perspective identified the concepts of project uncertainty and/or project risk as key constructs that need to be taken into account when managing a project. Strong parallels have been shown to exist in the meanings attributed to these terms (i.e., uncertainty and risk) in the IS literature [4], with both terms used to describe project characteristics that tend to increase the probability of project failure (e.g., project size, lack of user experience and support, and task complexity).

In the general risk literature, the probability of an unsatisfactory outcome is labeled "risk," while "risk exposure" is defined as this probability multiplied by the loss potential of the unsatisfactory outcome. To be consistent with this literature, the present paper adopted this latter definition. Accordingly, the term "risk exposure" is used here to refer to the notion of risk defined in [4]. It is also important to note that, in some contexts, the probability of occurrence of an undesirable outcome can be estimated on the basis of past performance of the object under study. However, in many contexts, including software project management, such an assessment is almost impossible to perform. In these contexts, many risk assessment approaches approximate

the probability of an undesirable outcome by identifying and assessing situational characteristics likely to influence the occurrence of the outcome. These characteristics are generally labeled risk factors, and characterize the approach adopted here—as well as in a multitude of IS studies on software project risk management—to assess software project risk [32].

Risk Management Profile

The present study conceptualizes the construct of software project Risk Management Profile as a *profile multidimensional construct* [28]. According to Law et al. [28], unlike the latent and aggregate models, where a multidimensional construct can be summarized as a single overall representation of all the dimensions, for profile constructs "there is not a single theoretical overall construct that summarizes and represents all the dimensions. For example, although a person can be identified as high or low in general mental ability and job satisfaction, one cannot say that a person is high or low in personality. Personality as a profile multidimensional construct can be interpreted only under various profiles. A person can score high or low in one personality profile but not high or low in personality as an overall construct" (p. 747).

These arguments also apply to the global construct of software project risk management: While a project cannot score high or low in risk management, it can score high or low in the extent to which different project risk management tools or approaches are employed in the project (e.g., high or low user participation, high or low utilization of project planning tools and approaches, etc.). Thus the construct of Risk Management Profile can only be interpreted under the various profiles of the approaches, tools, techniques, devices, or mechanisms utilized in managing a software project, and is therefore conceptualized as such in the present study.

We identified three key dimensions of project management practice along which the construct of Risk Management Profile can be assessed: formal planning, internal integration, and user participation. These three constructs reflect three project management approaches that capture key features of the various approaches suggested in the literature and listed in Table 2. For example, formal planning—defined as the reliance on plans, schedules, and budgets to ensure the efficient and timely execution of a project [2, 34]—has often been cited as a low information processing capacity project management approach [2, 55] and to a certain extent, to reduce the amount of information to be processed [40]. Formal planning is also related to the arm's length strategy [5, 6], where cost estimates are used to guide decisions, and formal specifications and a price tag act as a "buffer" between IS staff and users (i.e., direct interaction between users and developers is reduced, and is replaced by formal plans and formal specifications). Formal planning also reflects an impersonal mode of coordination [55], where policies, plans, and schedules are used in order to coordinate work. Moreover, policies, plans, and schedules reflect vertical means of coordination, and as such are related to aspects of vertical coordination [40], where coordination between users and IS staff is achieved through vertical means, such as authorized entities (e.g., strong project managers or steering committees). Similarly, defined as

management practices that increase communication and cohesion among team members [34], internal integration is a high information processing capacity approach akin to the notion of "group mode of coordination" identified by Van de Ven et al. [52], and cited by Zmud [55] as "the mutual interaction among the members of the task force" (p. 48). Finally, user participation comprises all those activities that increase communication and information exchange with users [2, 8, 35, 36] and shares similarities with the concept of personal coordination mode, which includes liaison devices [55]. User participation is also related to the concept of horizontal coordination, defined by Nidumolu [40] as the extent of coordination between users and project staff, and is similar to the matrix project management strategy, where the "most crucial ingredients . . . are those which bridge the gap between MIS and the user" [6]. Thus formal planning, internal integration, and user participation capture both high and low information processing capacity approaches to managing software project risk, as well as providing an integrated view of a multitude of approaches that have been suggested in the literature.

Given the appropriateness of viewing the concept of project management as a profile multidimensional construct, and having identified formal planning, internal integration, and user participation as three project management approaches representing and capturing many of the key project management notions conceptualized in past research, the present study defined Risk Management Profile as a profile multidimensional construct whereby a software project is characterized along these three dimensions.

Fit

This construct reflects the extent to which a project's Risk Management Profile matches its Risk Exposure. When adopting a contingency approach to studying a phenomenon, researchers must carefully define their conceptualization of fit. Definitional rigor is critical, since different conceptual definitions of fit imply different meanings of a contingency theory and different expected empirical results [15, 53]. Venkatraman [53] proposed a framework that defined six different perspectives from which fit could be studied. These six perspectives can be classified into two broad categories, according to the number of variables being simultaneously examined: the "reductionist" conceptualization of fit and the "holistic" conceptualization [54]. The reductionist conceptualization "is based on a central assumption that the coalignment between two constructs [and] . . . can be best understood in terms of pairwise coalignment among individual dimensions that represent two constructs" (p. 2). Three of the six perspectives of fit defined by Venkatraman [53] belong to this conceptualization. They are the moderation, the mediation, and the matching perspectives. The holistic conceptualization of fit emphasizes the systemic nature of coalignment, wherein several variables are examined simultaneously. Fit as covariation, fit as profile deviation, and fit as gestalt all pertain to the holistic perspective. Several researchers have argued that the latter conceptualization was richer, because of its ability to retain the complex and interrelated nature of the relationships between variables [15, 37, 54].

Given that it considers several variables simultaneously, the profile multidimensional construct conceptualization of Risk Management Profile adopted in the present study suggests a holistic conceptualization of fit. In addition to classifying fit perspectives according to the number of variables examined, Venkatraman's framework further characterizes each perspective along two other dimensions: the presence—or absence—of a criterion variable, and the degree of specificity of the functional form of fit. In this study, there exists a criterion variable—project performance—but the degree of specificity of the relationship studied is low. For example, although the literature provides specific recommendations for internal integration and user participation in the case of high-risk projects (recommending high levels of both), it says little about these two constructs for low-risk projects. Similarly, although the literature recommends high levels of formal planning for low-risk projects, it is less clear about the levels of formal planning recommended for high-risk projects. Consequently, the relationship between the constructs of project risk management and performance are characterized by a low degree of specificity. A low degree of specificity suggest that—among the three perspectives which belong to the holistic conceptualization—fit as profile deviation best reflects the characteristics of the contingency phenomenon studied here. Consequently, fit is defined here as the degree of proximity to an externally specified profile or pattern, whereby "patterns of consistency among dimensions of organizational context, structure, and performance" ([15], p. 520) are simultaneously examined. Adapting Venkatraman's definition of fit as profile deviation to the context of software development projects, this approach implies that if an ideal pattern for Risk Management Profile is specified for a particular level of Risk Exposure, a software project's degree of adherence to such a multidimensional profile will be positively related to Performance if it has a high level of risk exposure–project management practices Fit. Conversely, as shown in Figure 1, deviation from this profile would imply a low degree of fit, resulting in a negative effect on Performance. According to Venkatraman, "this perspective allows a researcher to specify an ideal profile and to demonstrate that adherence to such a profile has systematic implications for effectiveness" ([53], p. 434).

An important issue when adopting this conceptualization of fit is the development of an ideal pattern or profile. According to Venkatraman, there are two obvious alternatives: (a) developing them theoretically, or (b) developing them empirically. Venkatraman argues that while the first alternative is "intuitively appealing, the operational task of developing such a profile with numerical scores along a set of dimensions is difficult" ([53], p. 434). Moreover, past research does not provide sufficiently precise levels for the different project management approaches it recommends for different levels of project risk. Consequently, developing theoretically ideal profiles based on past research is not currently feasible.

The second alternative is to develop an ideal profile using a calibration sample—generally defined as the data points that are closer to the top of the performance scale. Then a measure of the closeness between the two patterns (the Euclidean distance between the ideal pattern for a project and the project's actual pattern) is correlated with indicators of project performance. Significant—but negative—correlations pro-

vide evidence for the presence of contingent relationships, as defined by the profile deviation perspective of fit [15].[1] Consistent with Venkatraman's recommendation, the present study adopted the second approach, as described in the following section.

Method

Sample and Data Collection

INITIALLY, A LETTER DESCRIBING THE STUDY was sent to the IS managers of the largest 100 companies in Quebec, and to all the ministries, government agencies, and public corporations of the province. A few weeks following the mailing, the recipients were contacted by phone to solicit their participation and to inquire about the availability of software projects in progress, but with system conversion not yet completed. The purpose of this requirement was twofold. The fact that the projects were still ongoing meant that in answering questions about project characteristics and management activities respondents would report on current events, thereby eliminating retrospective bias. This requirement also meant that the sample would contain projects at varying degrees of advancement, ranging from the requirements determination stage to system conversion, increasing the representativeness of the sample.

The sampling process resulted in an initial sample of 120 ongoing software development projects in 75 organizations. For each project two respondents were identified. First, the project leaders (i.e., the project managers) were interviewed, and were asked to respond to a questionnaire containing the items for Internal Integration, Formal Planning, the project's estimated cost, as well as the Project Risk Exposure items pertaining to the project leader (as per [4]).

Each project leader also assisted in identifying a user representative for the project. This individual was typically a member of the project team, was knowledgeable about the user community, the project's objectives, and the organizational goals, and could reliably reflect the views of the system's future users. For each project, a key user thus identified was also interviewed, and given a questionnaire containing the Project Risk Exposure items pertaining to the users [4]—that is, project characteristics items about which users would typically know more than the project leader (e.g., questions pertaining to user tasks for which the system was being developed and user characteristics). To ensure a high response rate, questionnaires were left with the respondents and usually hand-collected a few weeks later.

Once the initial data for a project were collected, the project was followed while awaiting its completion. After the completion of system implementation plus a three-month usage period, a second questionnaire was sent to the project leaders and the key users. The second project leader questionnaire contained items assessing System Quality and the actual cost of the project, whereas the second key user questionnaire contained items assessing User Participation. Following a two-year data collection period, a sample of 75 completed projects was obtained. Of the remaining 45 projects in the initial sample, 15 were still in development, 19 had been abandoned, and complete performance data from 11 projects could not be obtained for various reasons

Table 3. Project and Organization Characteristics—Initial Sample (N=120)

	Mean	Standard Deviation	Minimum	Maximum
Number of employees (organization)	3963	6664	5	23000
Number of employees (IS department)	4	148	0	1000
Estimated project cost (thousand $)	1840	5886	10	52000
Estimated project size (person-days)	2035	2995	20	20000
Estimated project duration (months)	20	16	3	84
Project team size (number of people)	11	12	1	100

(e.g., because the project leader had left the organization). The characteristics of the 75 projects and the organizations that form the final study sample are presented in Tables 3 and 4.

As can be seen from Tables 3 and 4, both the initial sample (N=120) and the final sample (N=75) contain projects that exhibit considerable variation in terms of project cost, duration, and size, suggesting that the final sample is representative of software development projects encountered in practice. Since performance data could not be obtained from 45 of the original 120 projects, it is possible that the final sample of 75 projects may consist of projects that were "more successful" than average, and could constitute a limitation of the present study. However, the 75 projects were analyzed in terms of estimated versus actual cost and duration. This analysis showed that more than 50 percent of the 75 projects were over budget and 42 percent were over their estimated duration. As these results are quite similar to industry averages typically cited [20, 22, 24], they suggest an adequate representativeness for the study sample. The development phase during which data were collected from the initial sample was relatively evenly distributed along five life cycle phases, with 20.3 percent of the projects being in the preliminary study phase, 14.1 percent in the system analysis phase, 10.9 percent in the system design phase, 17.2 percent in the physical design phase, and 37.5 percent in the implementation phase. Although this ensures the representativeness of the study sample, it may raise a question regarding the measurement of some of the 23 Risk Exposure variables (e.g., respondents may have had differing views and perceptions depending on project stage or they may not have been able to answer some of the questions). However, each project leader was interviewed after they responded to the first questionnaire containing the Risk Exposure items. At no time during these interviews did the issue of them not being able to respond to these items arise, nor did any of them express any difficulties in this matter. To verify this we examined the missing items for the Risk Exposure variables and found that they

Table 4. Project and Organization Characteristics—Final Sample (N=75)

	Mean	Standard Deviation	Minimum	Maximum
Number of employees (organization)	4829	7426	70	23000
Number of employees (IS department)	113	176	0	1000
Estimated project cost (thousand $)	1686	6176	2	51000
Estimated project size (person-days)	2186	3544	30	21300
Estimated project duration (months)	18	12	1	60
Project team size (number of people)	12	14	1	100

did not exceed 12 percent of the sample for any single item, supporting our conclusion. Also, with the large number of items used to measure risk (93 items for 24 risk variables), the impact of missing items on the reliability and validity of the measurement of risk (because project leaders could not respond to some questions) is likely to be negligible.

On the average, the project leaders in the sample had previously managed 3.93 similar projects (standard deviation = 4.46, minimum = 0, maximum = 20) and had 53.5 months experience as project leaders (standard deviation = 40, minimum = 1, maximum = 180). The users in the sample were staff (36.5 percent), senior executive (8.1 percent), department head (20.3 percent), middle manager (25.7 percent), and other (9.5 percent), and had an average of 46 months experience in their current positions (standard deviation = 41.3, minimum = 3, maximum = 180) .

Measures

Risk Exposure was assessed with the measure proposed by [4], and is reproduced in the Appendix for the reader's convenience. Accordingly, project characteristics that increase the probability of project failure were assessed with 23 risk variables along five dimensions: technological newness, application size, lack of expertise, application complexity, and organizational environment. Of the 23 risk variables, 11 were measured with multiple-item scales, 11 were ratio variables, and 1 was measured with a single-item Likert-type scale. The magnitude of potential loss in case of project failure was assessed by averaging the eleven items comprising this variable. For all multiple-item scales, convergent validity was assessed by examining factor analyses of scale items, inter-item correlations of each scale, and scale reliabilities. (However, no factor analysis was performed when the sample size/number of items ratio of a variable was less than 10.) Following the elimination of items with poor psychometric

properties, the items shown in the Appendix remained. The internal consistency reliabilities of all multiple-item scales are shown in the Appendix, following each scale's description, and indicate an adequate level of measurement (eight of the twelve scales had reliabilities of 0.80 or above, two of 0.75 and 0.76, one of 0.65, and one of 0.61). Scores for each of the 12 multiple-item risk variables were then calculated by averaging their respective items. Finally, a Risk Exposure score for each project was calculated by averaging that project's scores on the 23 risk variables[2] and by multiplying this average with the project's magnitude of potential loss score.

A project's *Risk Management Profile* was assessed with *Internal Integration, User Participation,* and *Formal Planning.* The items used in measuring the three constructs are shown in the Appendix. Internal Integration was operationalized with four items assessing the extent to which liaison and communication devices were employed to achieve collaboration and coordination between the members of the project team [6, 34, 55]. The construct of user participation—defined as the behaviors and activities that the target users perform in the system development process [3]—and its associated measure capture key aspects of this relationship. Specifically, the measure proposed by [3] assesses user participation along three dimensions: Responsibility, User–IS Relationship, and Hands-On Activity. Of these, the Responsibility and User–IS Relationship items capture the extent to which liaison and communication devices between the project team and the users were employed. Since it is through such participation activities and assignments that collaboration and coordination between the project team and the user community is achieved, the items of these two dimensions were not modified. The Hands-On Activity items of [3] were modified to assess users' role in system analysis, design, and implementation (User Participation items 6, 7, and 8 in the Appendix) since it was felt that such an overall assessment would provide a closer evaluation of the coordination and collaboration between the project team and the users compared to the assessment of specific analysis, design, or implementation activities found in the original instrument. Formal Planning items were based on [2, 6, 34, 55], and assessed the extent to which formal planning tools and planning practices were used in the project.

To assess convergent and discriminant validity for the three constructs of Risk Management Profile, the following analyses were conducted. First, correlations were calculated among all 22 items constituting the three measures. Second, the resulting correlation matrix was examined using multitrait-multimethod analysis [13]. Convergent validity was assessed by determining the extent to which items measuring the same trait indeed behaved as if they were measuring the same construct—that is, that they were positively and significantly correlated. In the case of Formal Planning, all inter-item correlations were positive and significant at $p < 0.001$ (Cronbach alpha = 0.82). For Internal Integration, all six inter-item correlations were positive, with four significant at $p < 0.001$, one significant at $p < 0.09$, and one not significant (Cronbach alpha = 0.67). Finally, for User Participation, all 105 inter-item correlations were positive, with 52 significant at $p < 0.01$, 23 significant at $p < 0.05$, and 15 significant at $p < 0.10$ (Cronbach alpha = 0.88). These results indicate an adequate level of convergent validity. Discriminant validity was assessed by examining the correlations

between items designed to measure different traits—here, Internal Integration, User Participation, and Formal Planning. Low and not significant correlations between items measuring different traits is an indication of discriminant validity. Of 222 correlations between items measuring different traits, 212 (95.5 percent) were not significant. That is, only 10 correlations (i.e., 4.5 percent)—a number that would be expected by chance—were significant at $p < 0.05$. These results suggest adequate discriminant validity of the Risk Management Profile measures.

Performance was assessed both from a product and a process perspective. Product performance (i.e., the performance of the project in terms of the quality of the system it delivered) was assessed by the project leader using an 18-item *System Quality* measure developed by [44]. As such, System Quality was assessed from the perspective of the project leader. This measure takes into account various system characteristics, including system reliability and performance, system costs and benefits, and the relevance of the information provided by the system. The internal consistency reliability of the items of this scale, shown in the Appendix, was adequate (Cronbach alpha = 0.88).[3]

The project's performance in terms of process was assessed with *Cost Gap,* reflecting the estimated versus the actual cost of the project, and was calculated as:

$$Cost\ Gap = 1 - (actual\ \$\ cost\ of\ project\ /\ estimated\ \$\ cost\ of\ project)$$

For Cost Gap, positive values indicate under-budget projects, while negative values indicate over-budget projects. Table 5 shows the means, standard deviations, ranges, and correlations of study variables.

While the measures used in the present study exhibit acceptable psychometric properties, the size of the final sample (N = 75), the large number of variables assessed, and the fact that many variables were measured with a large number of items precluded a full-fledged and simultaneous construct validity analysis of all the multiple-item measures employed, and is acknowledged as a limitation.

Testing the Contingency Model

AS DESCRIBED EARLIER, THE PRESENT PAPER DEFINED FIT as the closeness between the Risk Management Profile *or pattern* of a software project and an empirically determined "ideal" or "best practices" profile. The contingency model hypothesizes that as the distance between a project's profile and the ideal profile increases the project's performance will decrease. To operationalize deviations from an ideal profile, the calculation of a Euclidean distance score is suggested [15], which in effect represents the degree of fit. To the extent that the distance scores correlate significantly and negatively with performance measures, evidence for the presence of contingent relationships is obtained.

To test the contingency model, the procedure recommended by Drazin and Van de Ven [15]—also used by Gresov [21], and Nidumolu [41]—was employed. First, the sample of 75 projects was classified into five quintiles based on their Risk Exposure scores. Observations from the middle quintile were then dropped, so as to clearly delineate between low- and high-risk exposure projects. This left 58 projects in the

Table 5. Descriptive Statistics of Model Constructs and Their Correlations

	N	Mean	Min	(Max)	S.D.	Cronbach alpha	1	2	3	4	5
Risk Exposure	72	0.21	0.08	(0.47)	0.08	NA					
Internal Integration	74	5.18	2.33	(7.0)	0.98	0.67	0.05				
Formal Planning	74	4.30	1.0	(6.75)	1.40	0.82	-0.03	0.16			
User Participation	69	4.29	1.13	(6.67)	1.01	0.88	-0.02	-0.13	0.20		
Cost Gap	66	-0.17	-2.47	(0.91)	0.75	NA	0.16	-0.04	0.07	0.10	
System Quality	71	5.28	3.67	(6.53)	0.71	0.89	-0.19	0.26*	0.16	-0.01	-0.09

* $p < 0.05$

sample. Second, using the reduced sample, and in each of the two risk exposure categories, high-performance projects were selected (i.e., projects with Cost Gap or System Quality scores higher than half a standard deviation above the mean of the entire sample), resulting in four calibration subsamples: low Risk Exposure/large Cost Gap, high Risk Exposure/large Cost Gap, low Risk Exposure/high System Quality, and high Risk Exposure/high System Quality. Third, the ideal profile for each calibration sample was empirically derived by calculating the mean value of each of the three Risk Management Profile variables (Internal Integration, User Participation, and Formal Planning). The four empirically derived, ideal Risk Management Profiles are shown in Figure 2. The ideal profile means of Internal Integration, User Participation, and Formal Planning for the four calibration subsamples are shown in Table 6. To ensure that statistically different patterns had emerged, the three means for the low Risk Exposure/large Cost Gap subsample were compared with the three means for the high Risk Exposure/large Cost Gap subsample, using the t-test. This procedure was then repeated to compare the low Risk Exposure/high System Quality subsample means with those of the high Risk Exposure/high System Quality subsample. As shown in Table 6, two of these mean differences were significant when Cost Gap was the performance measure, and one was significant when System Quality was the performance measure. These results indicate that the empirically derived ideal profiles differ.

Fourth, deviations from the ideal profiles were derived by calculating the Euclidean distance between a given project's Risk Management Profile and the empirically derived ideal profile for the corresponding category of Risk Exposure. Following Drazin and Van de Ven [15], the Euclidean distance was computed as follows:

$$DIST_i = SQRT [SUM(X_{is} - X_{is})^2],$$

where DISTj is the distance score, for a given performance measure, between the ideal profile and project j's profile. X_{is} is the score of the ideal profile on the sth dimension (e.g., Formal Planning) and X_{js} is the score of the jth project on the sth dimension. For example, project j's distance score, with System Quality as the performance measure, was calculated as follows:

$$DIST_j = SQR [(5.71 - \text{Internal Integration score}_j)^2 +$$
$$(3.89 - \text{User Participation score}_j)^2 + (4.95 - \text{Formal Planning score}_j)^2],$$

if project j was low-risk (i.e., its Risk Exposure score was in the last two quintiles), and

$$DIST_j = SQR [(5.46 - \text{Internal Integration score}_j)^2 +$$
$$(4.90 - \text{User Participation score}_j)^2 + (4.52 - \text{Formal Planning score}_j)^2]$$

if project j was high-risk (i.e., its Risk Exposure score was in the top two quintiles).[4] A similar distance score was calculated for Cost Gap as the performance measure, for each project from the low Risk Exposure and the high Risk Exposure holdout samples.[5] Finally, the correlations between the projects' distance scores and their Performance scores (System Quality and Cost Gap) were calculated. The resulting correlations are presented in Table 7.
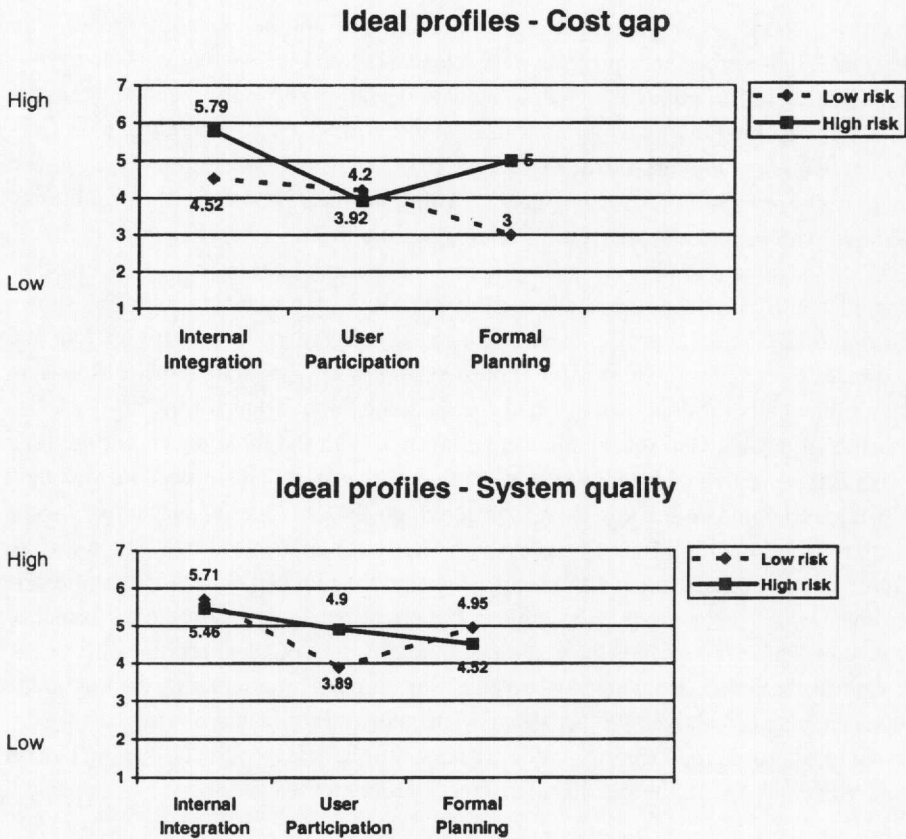
Ideal profiles - Cost gap

Ideal profiles - System quality

Figure 2. Empirically Derived Ideal Profiles

The results indicate that deviations from an ideal Risk Management Profile were negatively correlated with Performance, supporting the contingency model of Figure 1 and the research hypothesis. The closer a project's Risk Management Profile was to the empirically derived ideal profile, the higher its Performance, both in terms of Cost Gap and System Quality. Moreover, as can be seen from Table 5, the correlation between the two Performance measures, Cost Gap and System Quality, was –0.01, indicating their independence. The fact that the contingency model was supported for two uncorrelated, dependent variables also provides replication evidence for the study results.

When testing profile deviation fit models of contingency, Venkatraman [53] recommended that researchers "specify a baseline model to demonstrate that the predictive power of the measure of coalignment . . . is significantly better than a measure calculated as deviation from a random profile" (p. 435). Consistent with the procedure used by [54], a Baseline Project Management Profile construct was defined using three other project management variables that were similar to a number of project management variables assessed in past research [45], but that did not significantly correlate with the variables of the study. These were: *Prototyping* (a single item assessing, on a 1–7 scale, the extent to which prototyping described the development

Table 6. Differences Between Ideal Profiles, for Low and High Levels of Risk Exposure

	Low Risk Exposure—Large Cost Gap			High Risk Exposure—Large Cost Gap			t-test	
	μ	s	N	μ	s	N	t	p
Internal Integration	4.52	0.89	4	5.79	0.53	7	-3.01	0.02
User Participation	4.2	0.68	4	3.92	1.79	6	0.29	n.s.
Formal Planning	3	1.61	4	5	1.35	7	-2.22	0.05

	Low Risk Exposure—High System Quality			High Risk Exposure—High System Quality			t-test	
	μ	s	N	μ	s	N	t	p
Internal Integration	5.71	0.95	7	5.46	0.55	7	0.6	n.s.
User Participation	3.89	0.7	7	4.90	0.72	6	-2.59	0.03
Formal Planning	4.95	2.22	7	4.52	1	7	0.47	n.s.

Table 7. Correlations Between Performance Measures and Profile Distance Scores

Performance Measure	Correlation with Respective Profile Distance Score
Cost Gap	−0.53 (p < 0.01)
System Quality	−0.32 (p < 0.05)

approach utilized); *External Evaluations* (a single item assessing, on a 1–7 scale, the extent to which external hardware and/or software performance evaluations were obtained); and *Project Leader Discretionary Power* (a single item assessing, on a 1–7 scale, the extent to which the project leader had discretionary power in his decisions concerning the management of the project). All three items were responded to by the project leader during the first-wave questionnaires. Using the same model testing procedure as above, ideal profiles were empirically derived for Baseline Project Management Profile, distance scores were computed for the projects remaining in the sample after the removal of high-performance projects, and their correlations with the respective performance measures were calculated. Neither correlation obtained with the baseline model was significant, providing additional evidence in support of the significance of the negative correlations reported in Table 7.

An additional analysis was conducted to confirm that the Risk Management Profiles of Cost Gap and System Quality were different. If the two ideal profile patterns were the same, regardless of how Performance was measured, a significant negative correlation would be expected between one measure of Performance and the distance scores calculated using the other Performance measure (i.e., the distance scores calculated between projects' Risk Management Profiles and the ideal profile empirically derived from the high-performance projects, but using the other Performance measure). That is, in addition to obtaining a significant negative correlation between Cost Gap and the distance scores from the ideal profile derived using Cost Gap as the Performance measure, we would also expect to observe a significant negative correlation between Cost Gap and the distance scores from the ideal profile derived using System Quality as the Performance measure, and vice versa. The correlation between Cost Gap and the distance scores from the ideal profile derived using System Quality as the Performance measure was $r = -0.14$, ns, whereas the correlation between System Quality and the deviation from the ideal profile derived using Cost Gap as the Performance measure was $r = -0.01$, ns. The lack of significance of these cross-correlations provides additional support indicating that the empirically derived ideal profiles for Cost Gap and System Quality were indeed different.

Discussion and Conclusion

THE OBJECTIVE OF THE PRESENT STUDY was to test the general hypothesis that software project *Performance* is influenced by *Fit*, defined as the extent to which a project's *Risk Management Profile* matches its *Risk Exposure*. Drawing from the IS project

management and Organization Theory contingency research, three constructs reflecting software project risk management were identified to reflect the construct of Risk Management: Internal Integration, User Participation, and Formal Planning. Performance was assessed along two dimensions: Cost Gap (an objective measure of budget compliance) and System Quality (the quality of the developed system as perceived by the project leader). Adopting a profile deviation perspective of fit, the research model was tested and supported. That is, it was found that the farther a project's risk management was from an ideal profile described by its levels of Internal Integration, User Participation, and Formal Planning, the lower its performance (both in terms of budget compliance, and system quality).

These results suggest that a software project's Risk Management Profile needs to be adapted to its degree of Risk Exposure. In other words, high Risk Exposure projects seem to require a different Risk Management Profile than do low Risk Exposure projects. In particular, when Performance was measured with Cost Gap, the ideal profiles of high Risk Exposure projects were found to have higher levels of Internal Integration than low Risk Exposure projects. This supports the general hypothesis that higher levels of risk call for higher levels of information processing capacity in managing a project. On the other hand, for Cost Gap, the ideal profiles of high Risk Exposure projects were found to have significantly *higher* levels of Formal Planning than did low Risk Exposure projects. As such, the results support the opposite of a hypothesis, often cited in Organization Theory, whereby higher levels of uncertainty call for *lower* levels of formal planning.

When Project Performance was measured with System Quality, ideal profiles of high Risk Exposure projects had significantly higher levels of User Participation than low Risk Exposure projects. Again, this supports the general hypothesis that high-risk projects call for high information processing capacity management approaches. Since, for System Quality, no significant differences were found between Formal Planning levels of the ideal profiles of high versus low Risk Exposure projects, the Organization Theory hypothesis regarding Formal Planning was not supported.

The fact that a significant difference was found between the Formal Planning levels of ideal profiles of high and low Risk Exposure projects (for Cost Gap) may be explained by the temporary nature of software projects, and by the emphasis placed in these projects on process performance outcomes, such as budget compliance (i.e., Cost Gap). As noted by [41], "risk-based researchers differ from traditional structural contingency researchers in the degree to which they emphasize the importance of ensuring that the project stays under control and converges within time and budget" (p. 84). In temporary organizations, such as software project teams, deviations from planned budgets and schedules—which might be perceived as being minor problems in permanent organizations—may have critical adverse effects [11]. As such, formal planning and control are particularly important in software development projects. While in a permanent organization, such as a firm, highly uncertain environments call for much flexibility in planning, the results obtained in the present study (for Cost Gap) suggest that temporary organizations such as software development projects require high levels of Formal Planning for high Risk Exposure projects and low levels

of Formal Planning for low Risk Exposure projects. These results also suggest that IS researchers need to consider the temporal nature of software project contexts when adapting or applying Organization Theory concepts and hypotheses, as some of these may be operating differently in software projects than they do in more permanent organizations.

The overall pattern emerging from the results of the present study indicates that the most appropriate approach to managing software development projects depends on the performance criterion used. For practitioners, this finding has important implications. On one hand, the results show that when meeting project budgets is the key performance criterion, high-risk projects call for high levels of internal integration and high levels of formal planning. This suggests that, in such cases, project leaders should emphasize frequent team meetings, informing team members of important project decisions, keeping team turnover at a minimum, obtaining team member input when setting project goals and schedules, and using formal planning tools such as PERT or CPM, as well as formal cost and schedule estimation tools. On the other hand, when system quality is the key performance criterion, high-risk projects call for high levels of user participation. This suggests that, in such cases, project leaders should encourage users to assume greater responsibilities and to play a more active role in the project. The present study's findings also imply that in order to apply the most appropriate project management approach the key performance criterion to be used in evaluating the success of a project needs to be clearly identified from the outset.

Although the results of the present study provide interesting insights for software project management, more research is needed to overcome some of its limitations, as well as to further explore its findings. First, some of the measures used here, although possessing adequate psychometric properties, would gain from additional refinements. One Risk Exposure variable (Team Diversity) was measured with a single-item Likert-type scale, and two multiple-item Risk Exposure scales—as well as the Internal Integration scale—had adequate but relatively low reliability coefficients (i.e., in the 0.60 to 0.67 range). Efforts are needed to improve the measurement of these variables. Second, obtaining a larger sample size than the one used here would enable the derivation of ideal profiles from a larger number of projects, hence improving the significance of the findings. Third, obtaining multiple observations of each project throughout the life cycle, combined with a larger number of observations, would enable the examination of the contingency model throughout the life cycle stages. Finally, this study assessed project success with two performance measures—Cost Gap and System Quality from the project leader perspective. Finding that two contingencies are at work depending on which performance measure is used contributes to our understanding of software project management. However, it must be recognized that assessing System Quality from the project leader perspective provides a one-sided picture. More research is needed to examine whether the contingency relationships found here also apply when System Quality is measured from users' perspective, as well as for other measures of project performance.

The present paper, as well as other reviews of the literature, clearly indicates the need for an integrated approach to the study of project risk management that would

provide a synthesis of the varied and diverse approaches examined in the past. We believe that defining the notion of project risk management as a profile multidimensional construct, analyzing different risk management approaches in terms of their information processing capacity, and representing project risk management profiles through the constructs of internal integration, user participation, and formal planning, provide useful steps toward such a synthesis.

NOTES

1. This correlation should be negative, since the smaller the distance between the two profiles (i.e., the closer a project's management profile is to the ideal or best practices profile), the higher the project's performance will be.

2. Each of the 23 variables, when present in a project, increases the chances that the project will fail. When more variables are present in a project, or as the level of each variable increases, the project's probability of failure increases. As such, the 23 risk variables constitute an additive scale but they are not necessarily correlated with each other (e.g., the number of hardware suppliers and the lack of general expertise in the project team are not necessarily correlated since the latter depends largely on management decisions, which would tend to be independent of the former). Because many of the 23 risk variables are similarly independent, and because classical methods of assessing construct reliability and validity are based on the premise of correlated indicators, the traditional approaches to assessing reliability and validity are not appropriate for the construct of Risk Exposure defined here.

3. The small sample size to number of items ratio (75/18) prohibited a factor analysis of this scale.

4. Consistent with [15, 41], this approach assigns equal weights to the three Risk Management constructs. Van de Ven and Drazin [51] noted that the equal weight "assumption can be relaxed by introducing the possibility of differentially weighting the importance of deviation in each structural element in determining performance" (p. 351). The assumption of equal weights is a more restrictive condition, and provides a more conservative test of the contingency model. In addition, it is considered a more appropriate approach when there is no prior theory that suggests differential weights [41].

5. Three projects with Cost Gap scores more than two standard deviations below the mean were considered outliers and removed from the sample.

REFERENCES

1. Alter, S. Implementation risk analysis. *TIMS Studies in the Management Sciences, 13* (1979), 103–119.

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

3. Barki, H., and Hartwick, J. Measuring user participation, user involvement, and user attitude. *MIS Quarterly, 18,* 1 (1994), 59–82.

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

5. Beath, C.M. Strategies for managing MIS projects: a transaction cost approach. *Proceedings of the Fourth International Conference on Information Systems,* Houston, TX, ACM/SIGBDP, New York, 1983, pp. 133–147.

6. Beath, C.M. Managing the user relationship in information systems development projects: a transaction governance approach. *Proceedings of the Eighth International Conference on Information Systems,* Pittsburgh, PA, 1987, pp. 415–427.

7. Boehm, B.W. *Software Risk Management.* Los Alamitos, CA: IEEE Computer Society Press, 1989.

8. Boehm, B.W. Software risk management: principles and practices. *IEEE Software* (1991), 32–41.

9. Bourgeois, L.J. Strategic goals, perceived uncertainty, and economic performance in volatile environments. *Academy of Management Journal, 28,* 3 (1985), 458–573.

10. Brooks, F.P. *The Mythical Man-Month.* Reading, MA: Addison Wesley, 1975.

11. Bryman, A.; Bresnen, M.: Beardsworth, A.D.; Ford, E.T.; and Keil, E.T. The concept of the temporary system: the case of the construction project. *Research in the Sociology of Organizations, 5* (1987), 253–283.

12. Burns, T., and Stalker, G.M. *The Management of Innovation.* London: Tavistock, 1961.

13. Campbell, D.T., and Fiske, D.W. Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin, 56,* 9 (1959), 81–105.

14. Deephouse, C.; Mukhopadhyay, T.; Goldenson, D.R.; and Kellner, M.I. Software processes and project performance. *Journal of Management Information Systems, 12,* 3 (Winter 1995–96), 187–205.

15. Drazin, R., and Van de Ven, A.H. Alternative forms of fit in contingency theory. *Administrative Science Quarterly, 30* (1985), 514–539.

16. Duncan, R.B. Characteristics of organizational environments and perceived environmental uncertainty. *Administrative Science Quarterly, 17* (1972), 313–327.

17. Galbraith, J.R. *Designing Complex Organizations.* Reading, MA: Addison Wesley, 1973.

18. Galbraith, J.R. Organizational design: an information processing view. *Interfaces, 4,* 3 (1974), 28–36.

19. Galbraith, J.R. *Organization Design.* Reading, MA: Addison Wesley, 1977.

20. Gibbs, W.W. Software's chronic crisis. *Scientific American* (September 1994), 86–95.

21. Gresov, C. Exploring fit and misfit with multiple contingencies. *Administrative Science Quarterly, 34* (1989), 431–453.

22. Hoffman, T. Software snafu triggers order delays, loss. *Computerworld, 32,* 27 (July 6, 1998), 6.

23. Kappelman, L.A., and McLean, E.R. User engagement in the development, implementation, and use of information technologies. *Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences.* Maui, HI: IEEE Computer Society Press, 4 (1994), 512–521.

24. Keil, M., and Robey, D. Turning around troubled software projects: an exploratory study of the deescalation of commitment to failing courses of action. *Journal of Management Information Systems, 15,* 4 (Spring 1999), 63–88.

25. Keil, M.; Cule, P.E.; Lyytinen, K.; and Schmidt, R.C. A framework for identifying software project risks. *Communications of the ACM, 41,* 11 (1998), 76–83.

26. Kunz, J.C.; Christiansen, T.R.; Cohen, G.P.; Jin, Y.; and Levit, R.E. The virtual design team: a computational simulation model of project organizations. *Communications of the ACM, 41,* 11 (1998), 84–92.

27. Kydd, C.T. Understanding the information content in MIS management tools. *MIS Quarterly, 13,* 3 (1989), 277–290.

28. Law, K.S.; Wong, C.S.; and Mobley, W.H. Toward a taxonomy of multidimensional constructs. *Academy of Management Review, 23,* 4 (1998), 741–755.

29. Lawrence, P., and Lorsch, P. *Organization and Environment.* Homewood, IL: Irwin, 1967.

30. Levit, R.E.; Thomsen, J.M.; Christiansen, T.R.; Kunz, J.C.; Jin, Y.; and Nass, C. Simulating project work processes and organizations: toward a micro-contingency theory of organizational design. *Management Science, 45,* 11 (1999), 1479–1495.

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

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

33. March, J.G., and Simon, H.A. *Organizations*. New York: John Wiley, 1958.

34. McFarlan, F.W. Portfolio approach to information systems. *Harvard Business Review* (September–October 1981), 142–150.

35. McKeen, J.D., and Guimaraes, T. Successful strategies for user participation in systems development. *Journal of Management Information Systems, 14,* 2 (Fall 1997), 133–150.

36. McKeen, J.D.; Guimaraes, T.; and Wetherbe, J.C. The relationship between user participation and user satisfaction: an investigation of four contingency factors. *MIS Quarterly, 18,* 3 (1994), 427–451.

37. Miller, D. Toward a new contingency approach: the search for organizational gestalts. *Journal of Management Studies, 18* (1981), 1–26.

38. Miller, D. Environmental fit versus internal fit. *Organization Science, 3,* 2 (1992), 159–178.

39. Miller, D., and Friesen, P.H. *Organizations: A Quantum View.* Englewood Cliffs: Prentice Hall, 1984.

40. Nidumolu, S. The effect of coordination and uncertainty on software project performance: residual performance risk as an intervening variable. *Information Systems Research, 6,* 3 (1995), 191–219.

41. Nidumolu, S.R. A comparison of the structural contingency and risk-based perspectives on coordination in software-development projects. *Journal of Management Information Systems, 13,* 2 (Fall 1996a), 77–113.

42. Nidumolu, S.R. Standardization, requirements uncertainty and software project performance. *Information and Management, 31,* 3 (1996b), 135–150.

43. Reimus, B. The IT system that couldn't deliver. *Harvard Business Review, 75,* 3 (1997), 22–26.

44. Rivard, S.; Poirier, G.; Raymond, L.; and Bergeron, F. Development of a measure to assess the quality of user-developed applications. *Data Base, 28,* 3 (1997), 44–58.

45. Ropponen, J. *Software Risk Management: Foundation, Principles and Empirical Findings.* Jyväskylä: Jyväskylä University Printing House, 1999.

46. Ropponen, J., and Lyytinen, K. Can software risk management improve system development: an exploratory study. *European Journal of Information Systems, 6,* 1 (1997), 41–50.

47. Ropponen, J., and Lyytinen, K. Components of software development risk: how to address them? a project manager survey. *IEEE Transactions on Software Engineering* (1999).

48. Thompson, J.D. *Organizations in Action.* New York: McGraw-Hill, 1967.

49. Tung, R.L. Dimensions of organizational environments: an exploratory study of their impact on organization structure. *Academy of Management Journal, 33,* 4 (1979), 672–693.

50. Van de Ven, A.H. A framework for organizational assessment. *Academy of Management Review, 1,* 1 (1976), 64–78.

51. Van de Ven, A.H., and Drazin, R. The concept of fit in contingency theory. In L.L. Cummings and B.M. Staw (eds.), *Research in Organizational Behavior.* Stamford, CT: JAI Press, 1985, pp. 333–365.

52. Van de Ven, A.H.; Delbecq, A.L.; and Koenig, R., Jr. Determinants of coordination modes in organizational design. *American Sociological Review, 41,* 2 (1976), 322–338.

53. Venkatraman, N. The concept of fit in strategy research: toward verbal and statistical correspondence. *Academy of Management Review, 14,* 3 (1989), 423–444.

54. Venkatraman, N., and Prescott, J.E. Environment-strategy coalignment: an empirical test of its performance implications. *Strategic Management Journal, 11,* 1 (January 1990), 1–23.

55. Zmud, R.W. Management of large software development efforts. *MIS Quarterly, 4,* 2 (1980), 45–55.

APPENDIX

Risk Management

Internal Integration. 7-point Likert scale. Respondent: *project leader.* Time of measurement: during development. For each item the respondent indicated the extent to which the contents of the statement corresponds/does not correspond to what transpired in the project. (Standardized Cronbach alpha = 0.67)

1. The project team meets frequently.
2. Project team members are kept informed about major decisions concerning the project.
3. Every effort is made to keep project team turnover at a minimum.
4. Project team members actively participate in the definition of project goals and schedules.

User Participation. 7-point Likert scale. Respondent: *key user.* Time of measurement: three months after project implementation completed. For each item the respondents indicated the extent to which the contents of the statement completely/not at all describes what transpired in the project. (Standardized Cronbach alpha = 0.88)

1. Users took on the leadership role in the development of the system.
2. Estimating development costs was users' responsibility.
3. Evaluating system benefits was users' responsibility.
4. Covering unforeseen budget increases in the project was users' responsibility.
5. Selecting the hardware/software was users' responsibility.
6. Users played a major role in the system analysis phase of the project.
7. Users played a major role in the system design phase of the project.
8. Users played a major role in the implementation phase of the project.
9. One or more users acted as liaison between the users and the project team.
10. Ensuring project success was users' responsibility.
11. The project team drew up a formalized agreement of the work to be done.
12. Users were able to make changes to the formal agreements of the work to be done.
13. The project team kept users informed concerning project progress and problems.
14. Users formally evaluated the work done by the project team.
15. Users formally approved the work done by the project team.

Formal Planning. 7-point Likert scale. Respondent: *project leader.* Time of measurement: during development. For each item the respondent indicated the extent to which the contents of the statement corresponds/does not correspond to what transpired in the project. (Standardized Cronbach alpha = 0.82)

1. Tools such as PERT or CPM are used to closely follow the project's status.
2. Special attention is being paid to project planning.
3. Significant resources were allocated to estimate project times and budgets.
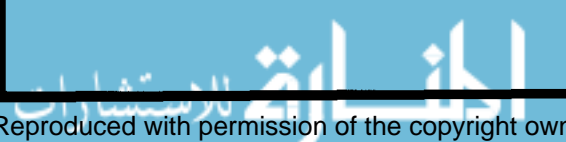
System Quality. 7-point Likert scale. Respondent: *project leader.* Time of measurement: three months after project implementation completed. For each item the respondents indicated the extent to which the contents of the statement completely/not at all describes the system that was developed. (Standardized Cronbach alpha = 0.88)

1. Reliable—(The system runs without errors, does what it is supposed to do, and the information it produces is error-free and accurate.)
2. Ease of use—(The system is easy to use.)
3. Secure—(The system enables recovery from errors, accidents, and intrusions while maintaining data security and integrity.)
4. Easy to maintain—(Programming errors can be easily corrected.)
5. Flexible—(The system can easily be modified to meet changing requirements.)
6. Technically simple—(The programs, the database structure, and the technical documentation are easy to understand.)
7. Portable—(The system can easily be adapted to a new technical or organizational environment.)
8. Efficient in its usage of resources—(The system performs its different functions without wasting technical resources.)
9. Testable–(It is easy to test whether the system is functioning correctly.)
10. Meets initial objectives—(The system conforms to the specifications established at the start of the project.)
11. Advantageous from a cost/benefit point of view—(The benefits that will be derived from the system exceed its cost.)
12. Understandable—(The system is easy to understand.)
13. Documented—(Documentation exists describing how the system functions and its structure.)
14. Quick—(The system performs its functions within acceptable delays.)
15. Precise—(The information produced by the system is precise.)
16. Complete—(The range of functions offered by the system is adequate.)
17. Relevant—(The information produced by the system is useful for the users.)
18. Recent—(The information produced by the system is up to date.)

Risk Exposure

From [4]. Respondent: both *project leader* and *key user* (as indicated below). Time of measurement: during development. Items indicated with an asterisk are reverse coded. Project Risk Exposure score was calculated as Overall Uncertainty (an average of 23 variables, after conversion to the same 0 to 1 scale) multiplied by Magnitude of Potential Loss (see [4], p. 215–216)

Technological Newness

1. *Need for New Hardware* (1-item binary scale; respondent: project leader): The new system will require the acquisition and installation of new hardware

2. *Need for New Software* (1-item binary scale; respondent: project leader): The new system will require the acquisition and installation of new software

3. *Number of Hardware Suppliers* (1-item ratio scale; respondent: project leader): How many hardware suppliers are involved in the development of this system?

4. *Number of Software Suppliers* (1-item ratio scale; respondent: project leader): How many software suppliers are involved in the development of this system?

5. *Number of Users Outside the Organization* (1-item ratio scale; respondent: project leader): Approximately how many people external to the organization will be using this system (examples of external users would be customers using an automated bank teller machine or an airline reservation system)?

Application Size

6. *Number of People on Team* (1-item ratio scale; respondent: project leader): How many people are there on the project team?

7. *Relative Project Size* (3-item Much Lower Than Average/Much Higher Than Average 7-point semantic-differential scale; respondent: project leader, Standardized Cronbach alpha = 0.82):

7a. Compared to other information system projects developed in your organization, the scheduled number of person-days for completing this project is:

7b. Compared to other information system projects developed in your organization, the scheduled number of months for completing this project is:

7c. Compared to other information system projects developed in your organization, the dollar budget allocated to this project is:

8. *Team Diversity* (1-item 4-point interval scale, one point added for each category checked; respondent: project leader): The project team members fall into which of the following groups (you can check more than one)

(1) Information system or data processing staff, (2) Outside consultants, (3) Users, (4) Other

9. *Number of Users in the Organization* (1-item ratio scale; respondent: project leader): Once it is implemented, how many employees of this organization will be using this system?

10. *Number of Hierarchical Levels Occupied by Users* (1-item ratio scale; respondent: project leader): What is the total number of different hierarchical levels occupied by the employees who will be using this system (for example, office clerks, supervisors, and managers each occupy different hierarchical levels in an organization)

Expertise

11. *Lack of Development Expertise in Team* (4-item 7-point No Expertise/Outstanding Expertise Likert scale; respondent: project leader, Standardized

Cronbach alpha = 0.76): Please evaluate the team's level of expertise in terms of the following:

11a. Development methodology used in this project
11b. Development support tools used in this project (e.g., DFD, flowcharts, ER model, CASE tools)
11c. Project management tools used in this project (e.g., PERT charts, Gantt diagrams, walkthroughs, project management software)
11d. Implementation tools used in this project (e.g., programming languages, database inquiry languages, screen generators)

12. *Team's Lack of Expertise with Application* (1-item 7-point semantic-differential scale; respondent: project leader): The members of the development team are: Very familiar with this type of application/Unfamiliar with this type of application
13. *Team's Lack of Expertise with Task* (4-item 7-point No Expertise/Outstanding Expertise Likert scale; respondent: project leader, Standardized Cronbach alpha = 0.87): Please evaluate the team's level of expertise in terms of the following:

13a. Overall knowledge of organizational operations
13b. In-depth knowledge of the functioning of user departments
13c. Overall administrative experience and skill
13d. Expertise in the specific application area of this system

14. *Team's Lack of General Expertise* (6-item 7-point Low/Outstanding Likert scale; respondent: project leader, Standardized Cronbach alpha = 0.80): Please evaluate the overall ability of the development team in terms of:

14a. Ability to work with undefined elements and uncertain objectives
14b. Ability to work with top management
14c. Ability to work effectively in a team
14d. Ability to successfully complete a task
14e. Ability to understand the human implications of a new information system
14f. Ability to carry out tasks quickly

15. *Lack of User Experience and Support* (15-item 7-point Strongly Disagree/ Strongly Agree Likert scale; respondent: project leader, Standardized Cronbach alpha = 0.81): Generally speaking, the users of this application:

* 15a. Have a positive opinion regarding the way in which the system can meet their needs
* 15b. Feel they need computerized support in carrying out the tasks for which the system is developed
  15c. Are not enthusiastic about the project
  15d. Have negative attitudes regarding the use of computers in their work
* 15e. Are ready to accept the various changes the system will entail
  15f. Do not actively participate in requirement definition

* 15g. Are available to answer the development team's questions
* 15h. Are aware of the importance of their role in successfully completing the project
  15i. Are not very familiar with information system development tasks and life cycle stages
* 15j. Are an integral part of the development team
  15k. Are not very familiar with data processing as a working tool
  15l. Have little experience with the activities to be supported by the future application
* 15m. Quickly respond to development team requests (for information, comments, approvals)
* 15n. Will have no constraints in fulfilling their development responsibilities for this system
  15o. Are not very familiar with this type of application

Application Complexity

16. *Technical Complexity* (3-item Slightly Complex/Highly Complex 7-point semantic-differential scale; respondent: project leader, Standardized Cronbach alpha = 0.65): Referring to the application currently being developed, how would you evaluate the technical complexity of each of the following elements:

    16a. The hardware (computers, networks)
    16b. The software
    16c. The database

17. *Number of Links to Existing Systems* (1-item ratio scale; respondent: project leader): How many existing information systems will be linked to this system?
18. *Number of Links to Future Systems* (1-item ratio scale; respondent: project leader): How many information systems currently under development will be linked to this system?

Organizational Environment

19. *Extent of Changes Brought* (4-item 7-point semantic-differential scale; respondent for 3a and 3b: project leader, for 3c and 3d: user representative, Standardized Cronbach alpha = 0.61)

    19a. The development of this system will require that user tasks be modified: Slightly/A great deal
    19b. In general, this system will lead to: Few organizational changes/Major organizational changes
    19c. The development of this system will require that user tasks be modified: Slightly/A great deal

19d. In general, this system will lead to: Few organizational changes/Major organizational changes

20. *Resource Insufficiency* (3-item More Than Enough/Extremely Insufficient 7-point semantic-differential scale; respondent: project leader, Standardized Cronbach alpha = 0.82)

20a. In order to develop and implement this system, the scheduled number of person-days is:
20b. In order to develop and implement this system, the scheduled number of months is:
20c. In order to develop and implement this system, the dollar budget provided is:

21. *Intensity of Conflicts* (6-item 7-point semantic-differential scale; respondent for 21a, 21b, and 21c: project leader, for 21d, 21e, and 21f: user representative, Standardized Cronbach alpha = 0.82): In this project, conflicts between team members:

21a. Rarely occur/Frequently occur
21b. Are not very serious/Are very serious
21c. Concern relatively unimportant matters/Concern very important matters

In this project, conflicts between the users and the team members:

21d. Rarely occur/Frequently occur
21e. Are not very serious/Are very serious
21f. Concern relatively unimportant matters/Concern very important matters

22. *Lack of Clarity of Role Definitions* (3-item 7-point semantic-differential scale; respondent: project leader, Standardized Cronbach alpha = 0.75)

22a. The role of each member of the project team is: Clearly Defined/Not Clearly Defined
22b. Communications between those involved in the project are: Pleasant/ Unpleasant
22c. The role of each person involved in the project is: Clearly Defined/ Not Clearly Defined

23. *Task Complexity* (20-item 7-point semantic-differential scale; respondent: user representative, Standardized Cronbach alpha = 0.85)

23a. The sequence of steps to be carried out to successfully complete these activities is: Easy to Identify/Hard to Identify
23b. Although the consequences of some activities are easy to predict, others are often unpredictable. The consequences of the activities in question are: Easy to Predict/Hard to Predict
23c. A well-defined body of knowledge on which to base the execution of these activities: Exists/Does Not Exist

23d. In general, one can determine whether or not the activities were successfully performed: Immediately/After a long period of time

23e. When problems arise in carrying out these activities, getting help is: Easy/Difficult

23f. When carrying out these activities, problems that cannot be immediately resolved arise: Rarely/Frequently

23g. Solving these problems typically requires: Little time/A Lot of time

23h. In your opinion, these activities are: Routine/Always new

* 23i. In general, carrying out these activities requires the use of: A Large Number of Methods and Procedures/A Small Number of Methods and Procedures

23j. These rules and procedures are: Rarely Subject to Change/Frequently Subject to Change

* 23k. Carrying out these activities requires: A Large Number of Different Steps/A Small Number of Different Steps

* 23l. These activities can be performed in: Many Different Ways/Only One Way

23m.Carrying out these activities generally involves: A Large Number of Repetitive Tasks/A Small Number of Repetitive Tasks

23n. When carrying out these activities, the extent of variety with respect to situations, actors, and tasks is: Low/High

23o. Regardless of the actors or the specific situations, the tasks and the procedures involved in carrying out these activities are: Always the Same/Extremely Varied

23p. In carrying out these activities: There Is a Single Objective to Reach/There Are Multiple Objectives to Reach

23q. When carrying out these activities all objectives: Can Be Reached/Cannot Be Reached

23r. When choosing a specific way to proceed: One Knows What the Result Will Be/One Does Not Know What the Result Will Be

23s. When evaluating the way in which all of these activities were carried out, the measure of their success is based on: One criterion/Several criteria

* 23t. Carrying out these activities depends on the execution of: Many Other Related Activities/Only a Few Other Related Activities

24. *Magnitude of Potential Loss* (11-item 7-point Little Impact/Large Impact Likert scale; respondent: project leader, Standardized Cronbach alpha = 0.80)

If, for some reason, the information system being developed is not implemented or if it has operational problems, what impact would this have on your organization in terms of the following:

24a. Customer Relations
24b. Financial Health

24c. Reputation of the Information System Department
24d. Profitability
24e. Competitive Position
24f. Organizational Efficiency
24g. Organizational Image
24h. The Survival of the Organization
24i. Market Share
24j. Reputation of the User Department
24k. Ability to Carry Out Current Operations