Journal of the Association for Information Systems

RESEARCH ARTICLE

The Effects of User Partnering and User Non-Support on Project Performance [1]

James J. Jiang
Department of Management Information Systems
University of Central Florida
&
National Central University, Taiwan
james.jiang@bus.ucf.edu

Gary Klein
College of Business and Administration
University of Colorado at Colorado Springs
gklein@computer.org

Hong-Gee Chen
Institute of Technology Management
National Tsing Hua University
hgchen@mx.nthu.edu.tw

Abstract

Information system software development projects suffer from a high failure rate. One of many obstacles faced by project managers is non-supportive users, those not actively sharing in development responsibilities. The coordination activity of early partnering has been proposed in the literature to promote collaboration and enhance user support. The extent of partnering is considered in a model that relates partnering to the risks of user non-support and eventual project success. The model is developed from contingency theory, with residual performance risk as an intermediary variable. A survey of IS project membership provides the data, which indicates that partnering significantly relates to higher user support, less residual risk, and better project performance. Researchers may use variations on the model to examine other barriers to success and the techniques applied to lower the barriers. Practitioners should consider applying partnering techniques to improve software development project performance.

Keywords: coordination, partnering, information system users, project management

[1] Sirkka Jarvenpaa was the accepting senior editor. Ananth Srinivasan was the associate editor. Gretchen Irwin and Elke Wolf were reviewers for this paper. This paper was submitted on November 5, 2002, and went through two revisions.

Introduction

Software project failures cost companies millions of dollars each year and often prevent key business objectives from being met (Glass, 1998; Gordon, 1999; Johnson, 1999). Failure estimates, defined primarily by cost and time budgets, run as high as 85%. Project managers and researchers express that a prime cause of schedule slippages and cost overruns on their projects was users who did not meet obligations (Barki and Hartwick, 1989; Barki et al., 1993; Lyytinen et al., 1998; Newman and Robey, 1992). Project managers complain that users hold development teams tightly to meeting milestone commitments, yet when it comes time for users to meet their own obligations, they are lax in doing so. This lack of support, manifested in an unwillingness to participate, resistance to change, and doubts about the worth of the system, is associated with the unwilling user and has been linked to eventual success of the project (Ives and Olson, 1984; Tait and Vessey, 1988).

A contingency perspective promotes management of risk with approaches that fit demands imposed by the degree of risks (Burns and Stalker, 1961). Since user support is a viable factor, managing the relationship between users and team members is critically important in the development of an information system (IS) (Robey and Newman, 1996). To overcome difficulties presented by unwilling users, project managers propose building a foundation for collaboration between project team members and anticipated functional users (Frame, 1994). The foundation must promote a full, rather than a token, partnership (Kirsch and Beath, 1996). A token partnership describes an ineffective working relationship between IS staff and users, in which conflicts go unresolved. Full partnership is developed by "a method of transforming contractual relationships into a cohesive, project team with a single set of goals and established procedures for resolving disputes in a timely and effective manner" (Cowan et al., 1992; Larson, 1997).

The purpose of the present study is to examine relationships among user partnering, user non-support, and project performance to determine the effectiveness of partnering strategies employed prior to implementation of a project. We modify a contingency model of the impact of coordination on risks and success to fit the particulars of partnering and user related risks (Nidumolu, 1995 and 1996). Specifically, we propose to answer the following research question: *How do user non-support and user partnering relate to the performance of software development projects?* An answer to this question may allow managers to plan interventions that serve to mitigate user-related risks.

Background

Organizational contingency theorists, following Burns and Stalker (1961), proposed that successful organizations establish a fit between the degree of uncertainty of their environment and their structural and process characteristics (Duncan, 1972; McKeen et al., 1994; Barki et al., 2001). The view is represented in Figure 1, which shows how uncertainty and a response activity fit to impact performance, and IS researchers have recognized the model's relevance to software project risk management. Table 1 highlights several studies involving the contingency model. According to their view, software development projects managed with approaches that fit the demands imposed by the degree of environmental uncertainty will be more successful.

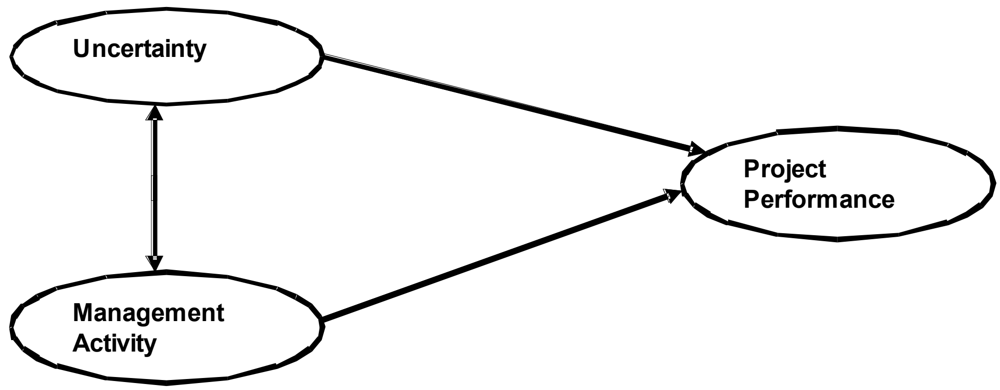

Figure 1. General Contingency Model of Software Project Risk

Table 1. Contingency Approaches in Project Management

Project Management Construct Studied	Authors	Recommended Course of Action	Support Provided
Project Management Activities (16 activities grouped into four categories)	Alter(1979) Alter and Ginzberg (1978)	Divide project, keep solution simple, develop good support base, meet user needs	Cross-sectional data from 85 projects
Relationship Between Parties Concerned	Beath (1983)	Use arm's length relationships	Case studies
User Participation	McKeen, et al. (1994) McKeen and Guimaraes (1997)	Use high levels of user participation	Cross-sectional data from 151 projects
Coordination Structure	Nidumolu (1995) (1996)	Use vertical coordination	Cross-sectional data from 64 projects
Risk Exposure	Barki, Rivard, and Talbot (2001)	Risk Management Profile	75 projects

In IS software development projects, project uncertainty and/or project risk is a key construct that needs to be taken into account when managing a project. Nidumolu presented an alternative model that related coordination to project risks and project outcomes (Nidumolu, 1995 and 1996). The unique aspect of his model, generalized in Figure 2, was the incorporation of software engineering philosophies that trace risks to multiple points in time, including late in the development life cycle. According to the model, risk late in the project lifecycle is characterized by an inability to estimate project parameters, but software that is in track loses risk as the project progresses. Formal coordination in the organization tends to mitigate the "residual" risks. The result of his study provides a framework to investigate specific risks and techniques that, based on theory, might have a relationship.

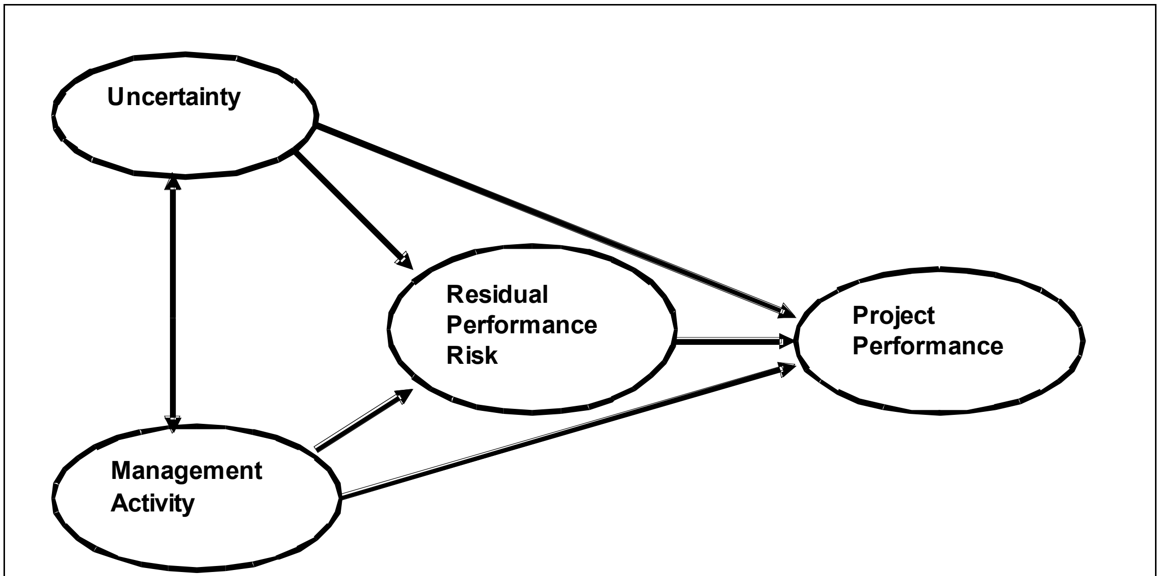

Figure 2. Incorporating Residual Performance Risk

One such type of risk is associated with users. Many studies relate user involvement, user participation, and user attitude to system quality, system usage, and user information satisfaction (Barki and Hartwick, 1989; Saleem, 1996; Shumskas, 1992). In general, user involvement refers to participation in the system development process by potential users or their representatives (Baroudi et al., 1986). However, a distinction is made to differentiate user involvement from the non-supportive user. Non-supportive users, as utilized in this study, are unwilling participants (Tait and Vessey, 1988), perceive no need for the system (Hirschheim and Newman, 1988), have few feelings of responsibility (Davis, 1982), and resist change (Jiang et al., 2000). According to Petty and Cacioppo's (1996) "attitude-behavior" theory, attitudes are individuals' general affective, cognitive, and intentional responses toward objects or other people. Furthermore, the stronger the attitude, the stronger is its link to behavior (Ajzen, 1991). Thus, when the users' overall attitude toward a new system is unfavorable, it is likely that they will not cooperate during a development effort, leading to an increased chance of project failure.

To increase the chance of success in light of such user risks, coordination efforts are required. Formal coordination theories require that tasks be allocated across organizational members, and communication and control mechanisms must facilitate the necessary information exchanges and decisional autonomy needed for effective collaboration and decision-making (Thompson, 1967; Van de Ven et al., 1976). Horizontal coordination between users and IS employees suggests a high degree of user participation in the project and open channels of communication among team members. Incomplete, ambiguous, or inconsistent requirements or frequent changes in them make it difficult to predict performance outcomes. Lack of information about requirements, therefore, makes it difficult to predict the performance outcomes.

Partnering is a specific management intervention designed to promote collaboration between the users and IS staff through coordination principles (Cowan et al.,1992). The partnership must be extended from the development team to the functional users of the proposed system.

Though these parties will never agree on all the goals of a system development, both should be interested in completing the project on time and within budget. Neither party wants rework. Neither party wants costly project failures. The existence of such common goals and the presence of limited resources (money, time) provide the basis for transforming an adversarial situation into an integrative, collaborative situation. Partnering processes should be established prior to the start of a system development project, however, different development and design methodologies can also impact the involvement of the user and eventual success of the project (Baskerville and Pries-Heje, 2004; Nerur et al., 2005; Roller et al., 2004).

The use of partnering to assist in the control of user risk results in the model in Figure 3. Project success is the ultimate goal and is associated with the ability to estimate project parameters. Likewise, success may be associated with the uncertainties as well as the devices used to mitigate uncertainty. Residual performance risk may also be associated with the technique used to mitigate the uncertainty. The model allows the testing of whether user partnering can lower the impact of the non-supportive user.

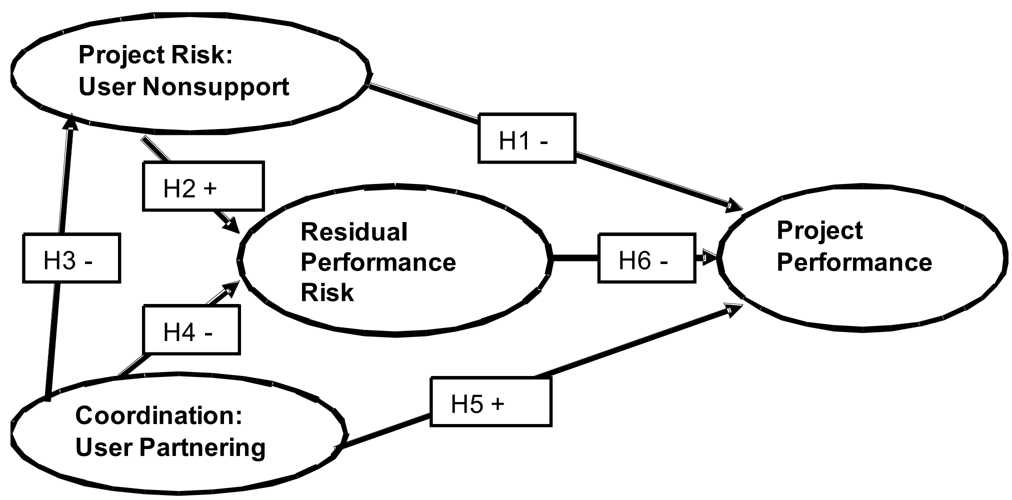

Figure 3: Research Model

Research Hypotheses

Organizational researchers have long recognized the impact of opposing interests between groups or individuals on team performance (Youngs, 1986). Non-supportive users have a negative attitude toward the system – they are not ready to accept the changes the system will entail, not enthusiastic about the system, and not willing to respond to the development team's requests. According to attitude-behavior theory, non-supportive users are not likely to facilitate the development or implementation of the system, to provide quick feedback to the system requirements, or to evaluate the system features that are critical to a successful system implementation (Beath and Orlikowski, 1994). User non-support is associated with resistance to change, users' feelings of responsibility, and an understanding of the need for the system (Anderson and Narashinian, 1979; Charette, 1989; Tait and Vessey, 1988). User risk is one component of project risks that indirectly impact performance of the process and product (Wallace et al., 2004).

Empirical evidence supports the idea that these and other critical functions and responsibilities of users during a system is development are important to the successful completion of a project, though the results are relatively weak and inconclusive (Ives and Olson, 1984; Kirsch and Beath, 1996). Users must perceive a need for a new system before it can be successfully implemented (Guthrie, 1974). Moreover, non-supportive users are less likely to describe the application and problem domains in a structured fashion, to specify requirements in cooperation with system developers, or to review the proposed design specifications of the system (Beath, 1991).

Accordingly, we expect both:

H1: User non-support is negatively associated with project performance.

H2: User non-support is positively associated with residual performance risk.

System users typically know more about work practices and procedures, information flow, organizational environments, and needs than analysts do. In fact, Beath and Orlikowski (1994) argue that a system development method must allow knowledgeable users to actively participate in critiquing, reviewing, and checking details to ensure a good system. But differences in the skills, perceptions, communication, culture, goals, and values between users and developers can cause potential conflict and derail projects (Larson, 1997). Hence, the project management literature has long suggested the importance of good communications between the IS project team and the IS user when defining project scope and controlling project changes (Boehm, 1989; Schwalbe, 2000).

Organizational theory supports the notion that coordination is crucial to gaining the essential support of users (Thompson, 1967). Moreover, software engineering researchers suggest that partnering should avoid unclear project objectives, insufficient resources, and shifting goals, thus fostering a stable work environment conducive to project effectiveness. Specifically, a coordinated partnership between users and system developers provides a more accurate assessment of user information requirements; promotes greater user acceptance, support, and understanding of the system; and avoids the implementation of costly system features that are unacceptable to users (Beath and Orlikowski, 1994; Ives and Olson, 1984; Newman and Robey, 1992).

As a specific management intervention, partnering overcomes the tendency to manage IS projects in adversarial fashion and controls conflict between the user and the project staff (Cowan et al., 1992). Such partnering often formally cements users and system developers as a team (Kirsch and Beath, 1996). In general, the principal stakeholders in the project conduct the partnership activities, which include all development team members. Core partnering activities include 1) a team building session, where key people meet to build a collaborative relationship; 2) conflict identification; 3) establishment of a problem-solving process for joint resolution of issues that may arise; 4) formulation of a charter or agreement, including shared objectives and responsibilities; and 5) establishment of a continuous improvement process (Larson, 1997). Developing a strong, cooperative relationship between these different stakeholders is a key factor to system success.

Accordingly, we expect:

H3: User partnering has a negative relationship with user non-support.

The enhanced involvement and communication fostered by partnering activities will improve the accuracy of project estimations. As customers and providers seek ways to achieve mutual objectives (Pruitt, 1981), their combined efforts to determine the scope of the project, estimate the budget, review relevant goals, review business assumptions made in the system, and make decisions regarding information flow and work procedures will result in a smooth, cooperative relationship. As horizontal and vertical channels of communication provide for the free flow of information, the difficulty in estimating the project's costs, schedules, and other performance consequences is reduced, and the likelihood that the project ends in success is increased. IS researchers have called for better partnering between system developers and users for these reasons (Beath and Orlikowski, 1994; Ives and Olson, 1984).

Based on such coordination expectations and empirical findings, we propose the following:

H4: User partnering will have a negative relationship with residual performance risk.

H5: User partnering is positively associated with project performance.

On the other hand, poor project estimates negatively impact final project performance. Without accurate estimates, IS managers do not know what resources still need to be committed to a development effort; and resource-dependency theory relates poor resource allocation to poor performance (Salancik and Pfeffer, 1974). Poor estimates can also lead to excessive schedule pressure and unrealistic expectations (Lyytinen and Hirschheim, 1987). The failure to consider residual performance risk and to take corresponding corrective actions is why many projects are unsuccessful (McFarlan, 1981). Accordingly, project management research suggests the ability to accurately estimate the final project's cost, time, and quality will impact the final project performance (Shumskas, 1992; Thamhain and Wilemon, 1987). From the above discussion we expect:

H6: Residual performance risk has a negative relationship with project performance.

Research Methodology

Sample

Those most familiar with project activities and risks in an organization are those who are most involved in the management of the project and are either trained or intimately aware of the importance of project management techniques and methods. To tap into this knowledge, we mailed questionnaire to 500 randomly selected Project Management Institute (PMI) members in the U.S. from a mailing list of those in the IS special interest group purchased from the organization. PMI is the professional association for practitioners of project management, with more than 100,000 members worldwide. PMI provides global leadership in project management and its published body of knowledge is a recognized, international standard for the practice of project management. Members of PMI represent a cross-section of managerial positions. We requested responses from those with recent experience with an IS project, and enclosed a postage-paid envelope with each questionnaire. All the respondents were assured that their responses would be kept confidential.

A total of 78 questionnaires were returned for a response rate of about 16 percent. In order to increase the sample size, we sent a follow-up mailing. The responses from both mailings totaled 170, for an overall response rate of 34%: high for a study in the IS field. T-tests on the means of independent and dependent variables described below indicated no difference between the two samples, lowering concerns of non-response bias. We then combined the two samples for further analysis.

We present a summary of the demographic characteristics of the sample in Table 2. About one-third of the respondents were administrators within their firms having served on a recent IS project and being familiar with the project management process within their organization. Almost two-thirds were IS project leaders or other IS professionals who served on a recent IS project. Average project work experience reported was more than seven years for the sample. Overall, the sample is well qualified to judge the issues related to risks and success of IS development.

Table 2: Demographics		
1. Gender:		
Male:	72%	
Female:	25	
2. Position:		
IS management:	35%	
IS Project Leader:	41	
Other IS Professional:	22	
3. Age:		
30 or below	4%	
31 – 40	26	
41 – 50	42	
51 or above	22	
4. Most Recently Completed IS Project Size:		
	Mean	**Range**
Members	15.5	2 - 200
Duration (months)	14.4	2 - 60
Cost ($)	3,369,000	Up to 110,000,000
Person Days	2795	Up to 108,000

Constructs

User partnering in our study is the extent to which partnering is enabled at the start of the project through various activities. The later use of the processes and contracts established during this early partnering process are not considered, as we only have an interest in the establishment of the coordination channels as a predecessor to later attitudes and success. We adopt the user partnering activity measure from Larson (1997). It includes five items shown in Table 3. A questionnaire asked respondents to identify the extent to which each of the items occurred in their most recently completed IS project (within one year). We scored each scale using a Likert-type, five-point scale ranging from disagree (1) to agree (5). We presented all items such that the greater the score, the greater the extent of user partnering activities. The overall construct was taken as an average of the retained items.

The user non-support construct measures levels of support through user participation, involvement, and attitude encountered in their most recently completed IS project. user non-support measure is a subset of items identified by Barki et al. (1993), and we also show the specific items in Table 3. They are designed to measure the IS respondent's perception of user non-support in a representative fashion and may not include every facet of attitude or directly measure perceptions of the user. Each item was scored using a Likert-type, five-point scale ranging from disagree (1) to agree (5). We presented all items such that the greater the score, the greater the lack of the user support.

Table 3. Measurement Items

Construct	Item
Coordination: User Partnering	A1: Before the project began key people met to build a collaborative relationship among the management/users and project team.
	A2: Before the project began key people from both management/users and project team identified potential conflict/problem areas.
	A3: Before the project began a documented process was in place for joint resolution of problems.
	A4: Key parties involved formulated a formal charter/agreement that stated shared objectives and responsibilities.
	A5: The project included provisions for continuous improvement.
Risk: User Nonsupport	U1: Users have a negative opinion about the system meeting their need
	U2: Users are not enthusiastic about the project
	U3: Users are not available to answer questions
	U4: Users are not ready to accept the changes the system will entail
	U5: Users respond slowly to the development team's requests
Residual Performance Risk (extent of difficulty in estimating during the later stages)	*R1: What the cost of the project would be
	R2: What the project completion time would be
	R3: What the benefits of the project would be
	R4: If software would be compatible with environment
	R5: If software would meet user need
Project Performance	P1: Able to meet project goals
	P2: Efficient operations
	P3: Quality of work produced
	P4: Significant amount of work produced
	*P5: Innovative and creative
	P6: Adherence to budget
	P7: Adherence to schedule
*removed due to low indicator reliability	

Residual risk represents the difficulty in estimating the project scope, time, and costs during the later stages of the project. As such, the measure is an outcome of development steps and indicates estimation difficulty, not why estimation is difficult, or the methods used in estimation. The residual performance risk measure is as applied by Nidumolu (1996). The scales were originally derived from McFarlan (1981). The questionnaire asked respondents the extent of difficulty in estimating five items during the *later* phases of the project from very difficult (1) to very easy (5) on a Likert-type scale. We presented each item, seen in Table 3, such that the greater the score, the easier the estimation of project performance at the later stage. We then reverse the scale is then reversed in subsequent discussions to comply with the direction of the hypotheses.

Many authors argue three dimensions of project performance; meeting budget, meeting schedule, and meeting user requirements (Wateridge, 1995). However, project performance is viewed differently by the various stakeholders in a system development as well as by researchers in information systems (DeLone and McLean, 1992; Linberg, 1999). Some researchers suggest further dimensions of project performance: operation efficiency, amount of work produced, quality of work produced, and effectiveness or ability to meet project and user goals (Henderson and Lee, 1992). The items we selected come from multiple sources and are intended to reflect a variety of success measures that can be judged by IS professionals (Henderson and Lee, 1992; Jones and Harrison, 1996). The items are listed in Table 3. The questionnaire asked respondents' satisfaction in their most recently completed IS project. Each scale was scored using a Likert-type, five-point satisfaction scale ranging from disagreement (1) to agreement (5) with the items of satisfaction. We presented all items such that the greater the score, the greater the satisfaction of the particular item.

Construct Analysis

A threat to external validity could occur if the sample showed systematic biases in terms of demographics, such as age, gender, and position. We conducted an ANOVA by using project performance as the dependent variable (defined fully below) against each demographic category (as independent variables). Results did not indicate any significant relationship (none is significantly related to project performance). We found similar results for the other variables in the study. The external validity of the findings is also threatened if the sample is systematically biased – for example, if the responses were generally from more successful projects. Table 4 shows the descriptive statistics for the constructs. The responses had a good distribution for project performance, since the means and medians were similar, skewness was less than two, and kurtosis was less than five (Ghiselli et al., 1981). Similar results held for the remaining variables. Finally, external validity is improved if the measures are similar to those found in other studies. In our case, the residual performance risk measure is the same as that in another study (Nidumolu, 1996), at the 94% confidence interval and our project performance measure, comes within 1% of two other studies (Henderson and Lee, 1992; Jones and Harrison, 1996). These similarities reduce concerns for non-response and systematic bias.

The first stage of the analysis involved assessment of the reliability of the measures used to operationalize the variables in the study. We used confirmatory factor analysis (CFA), which is a multivariate technique that facilitates testing of the psychometric properties of the scales used to measure a variable. And we also estimated the parameters of a structural model – that is, the magnitude and direction of the relationships among the model variables (Anderson and Gerbing, 1988). The model describes the relationships or paths among theoretical constructs. Behind the structural model in Figure 3, there is a related measurement model shown in Figure 4, which links the constructs in the diagram with a set of items.

Table 4: Descriptive Statistics for the Constructs

	Cronbach's alpha	Mean	Std Dev	Median	Skewness	Kurtosis
Coordination: User Partnering	.86	2.85	1.03	2.80	-.02	-.93
Project Risk: User Nonsupport	.87	2.81	1.03	2.75	-.04	-.91
Residual Performance Risk	.74	3.52	.83	3.50	-.27	-.38
Project Performance	.83	3.56	.75	3.67	-.46	-.33

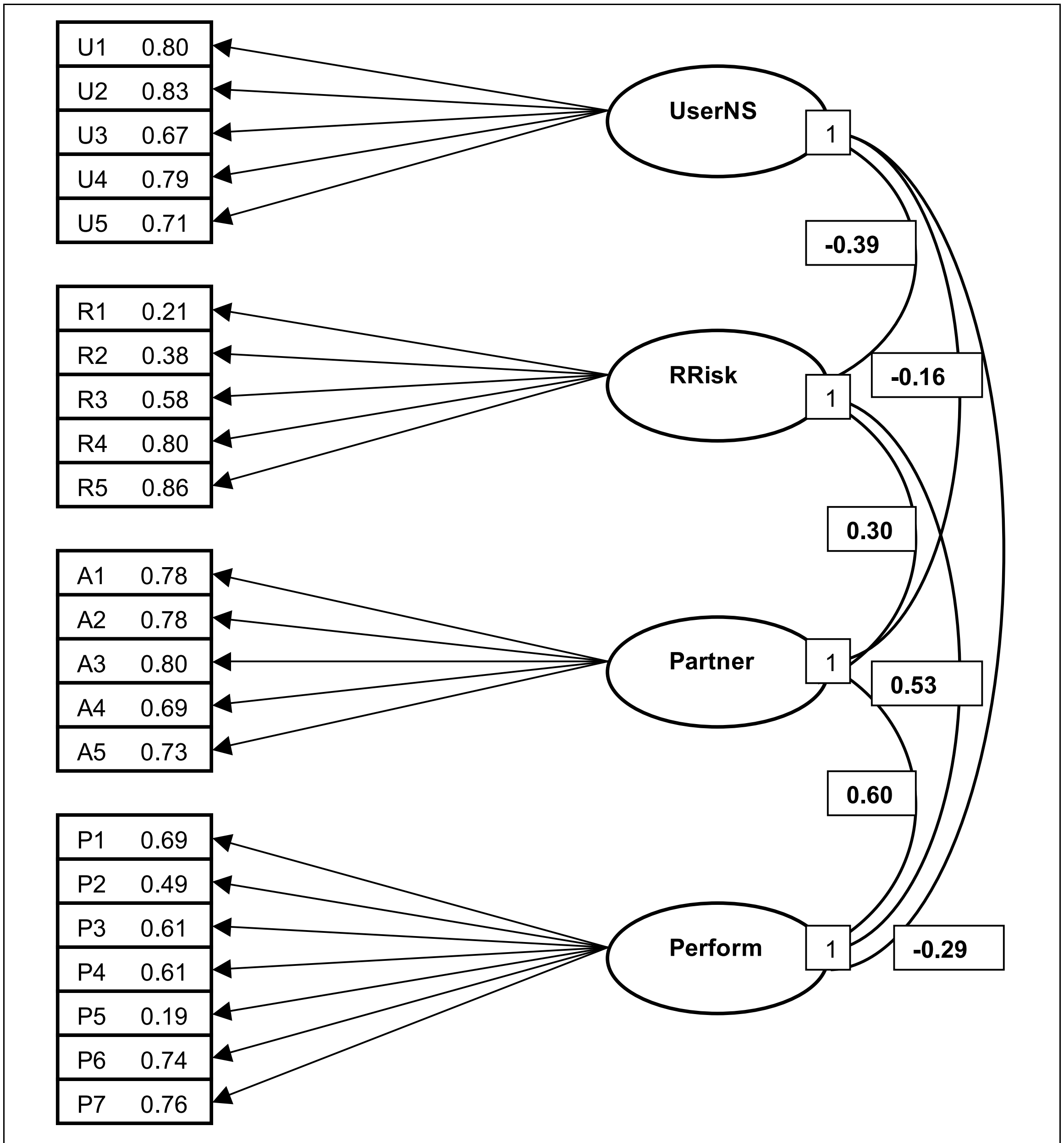

Figure 4. Measurement Model

When conducting a CFA, if the model provides a reasonably good approximation to reality, it should provide a good fit to the data. The CFA for the measurement model resulted in an Adjusted Goodness of Fit Index of .82 (>.80 is recommended), a Root Mean Square Residual of .08 (< .10 is recommended), a Chi-square/Degree of Freedom ratio of 1.71 (< 3 is recommended), a Comparative Fit Index of .92(>= .90 recommended), and a Non-normed Fit Index of .91 (>= .90 recommended). The recommended values are based on research traditions in IS and established authors in the field of structural equation modeling (Bentler, 1989; Bentler and Chou, 1987; Hair, et al., 1992; Kettinger and Lee, 1994; Kline, 1998; Mulaik, et al., 1989; Segars and Grover, 1993; Williams and Hazer, 1986).

In determining the appropriate minimum loadings required for the inclusion of an item (in the boxes with each item link in Figure 4), we retained items that loaded highly on their respective constructs. Loadings greater than .30 are considered significant; loadings greater than .40 are considered more important; and loadings .50 or greater are considered to be very significant (Hair, et al., 1992). Convergent validity is assessed through the t-tests on the factor loadings, such that the loadings are greater than twice their standard error (Anderson and Gerbing, 1988). The t-test results indicated that the constructs demonstrate convergent validity, as all t-values were significant at the .05 level. In addition, the internal consistency reliability of each construct is examined by the Cronbach alpha value, which will be high if the various items that constitute the construct are strongly correlated with one another. The Cronbach alpha values (Table 4) all exceeded the recommend level of .70 (Nunnally, 1978), as reported in Table 4.

Discriminant validity is demonstrated when different instruments are used to measure different constructs, and the correlations between the measures of these different constructs are relatively weak. Discriminant validity is assessed by the confidence interval test (Anderson and Gerbing, 1988). The confidence interval test to assess the discriminant validity between two factors involves calculating a confidence interval of plus or minus 2 standard errors around the correlation between the two factors, and determining whether this interval includes 1.0 (or –1.0). If it does not include 1.0, discriminant validity is demonstrated (Table 5). The results support the discriminant validity of the factors in this study.

Table 5: Discriminant validity (Confidence Interval Tests)

	Estimate	**Standard Error**	**Lower Bound**	**Higher Bound**
Coordination: Partnering Risk: User Non-support	-0.16	0.09	-0.34	0.02
Coordination: Partnering Residual performance risk	0.3	0.08	0.14	0.46
Coordination: Partnering Project Performance	0.6	0.06	0.48	0.72
Risk: Non-user Support Residual performance risk	-0.39	0.08	-55	-0.23
Risk: Non-user Support Project Performance	-0.29	0.08	-0.45	0.13
Residual Performance Risk Project Performance	0.53	0.07	0.39	0.67

Results

The present analysis followed a two-step procedure based in part on an approach recommended by Anderson and Gerbing (1988) and adopted in the IS literature (Segars and Grover, 1993). In the first step, confirmatory factor analysis was used to develop a measurement model. A measurement model describes the nature of the relationship between a number of latent factors and the manifest indictor variable that measures those latent variables. In step two, we modified the measurement model so that it came to represent the theoretical model of interest. The indicators used to present the latent variables in the theoretical model tested in the present study are identical to those presented in our measurement model. The analysis of this model may be described as a path analysis with latent variables. The path model consists of the unobservable constructs and the theoretical relationships among them (the paths). It evaluates the explanatory power of the model and the significance of paths in the structural model, which represent hypotheses to be tested. The estimated path coefficients indicate the strength and the sign of the theoretical relationships.

We tested the structural model using path analysis with unobservable constructs, specifically, structural equation modeling techniques using the SAS CALIS procedures. Three important assumptions associated with path analysis are: 1) normal distribution of examined variables, 2) absence of multicollinearity among examined variables, and 3) a limit on the maximum number of variables in the model. To test for normality, we conducted Mardia's multivariate kurtosis and normalized multivariate kurtosis tests and found no violation (< .05). Multicollinearity is a condition in which one or more variables exhibit very strong correlations (> .80) with one another (Anderson and Gerbing, 1988). The correlations between variables were all less than .80, thus no significant violation of multicollinearity was present. Although there is no theoretical limit on the total number of variables in a structural equation modeling, the total number of variables in this model was five (four to six is recommended for a stable result (Bentler and Chou, 1987)).

Table 6: Results of Path Analysis

Independent Variable	Corresponding Dependent	Hypothesis	Path Coefficient	t-Statistics	p-value
Risk: User Non-support	Project Performance	H1	-0.05	-0.62	0.54
Risk: User Non-support	Residual Performance Risk	H2	0.33	3.84*	0
Coordination: Partnering	Risk: User Non-support	H3	-0.24	-2.73*	0.01
Coordination: Partnering	Residual Performance Risk	H4	-0.3	-3.49*	0
Coordination: Partnering	Project Performance	H5	0.51	6.56*	0
Residual Performance Risk	Project Performance	H6	-0.34	-3.99*	0

Note: * indicates significant at p = .05 level

The theorized model fit the data well with an Adjusted Goodness of Fit Index of .81 (>.80 is recommended), a Root Mean Square Residual of .09 (< .10 is recommended), a Chi-square/Degree of Freedom Fit of 1.75 (< 3 is recommended), a Comparative Fit Index of .92 (>= .90 recommended), a Non-normed Fit Index of .90 (>= .90 recommended), a Relative Normed Fit Index of .92 (>= .90 recommended), and a Parsimony Normed Fit Index (PNFI) of .73 (>= .60 recommended). Table 6 summarizes the results. Hypotheses H2, H3, H4, H5, and H6 were all supported with respective path coefficients of .33, -.24, -.30, .51, and -.34, respectively. The t-statistics for these five hypotheses all exceeded significance at the .05 level, indicating these relationships hold statistical significance. The total variance explained of the project performance in the examined model was .54. We illustrate the relationships in Figure 5.

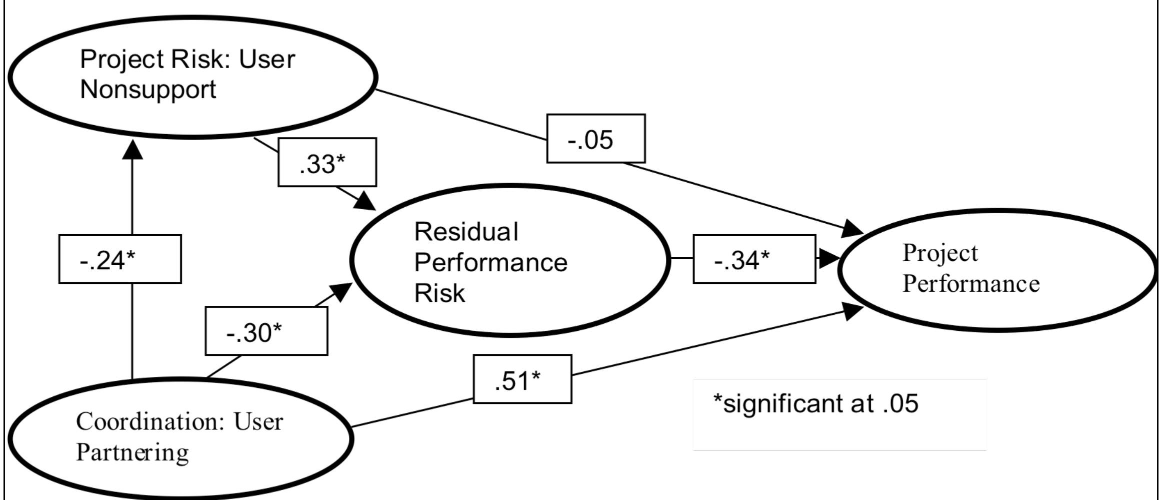

Figure 5: Research Model Path Coefficients

Hypothesis 1, the direct relation between user non-support and project performance, was not statistically supported. One reason for this may be the narrow focus on project success that does not include global benefits to the organization such as learning and process improvement. It could also be that user involvement or participation is often symbolic or problematic due to poor communication (Hirschheim and Newman, 1988; Lyytinen and Hirschheim, 1987; Newman and Robey, 1992). Barki and Hartwick (1989) argue that some variables exist between user involvement and various indicators of system success. In this study, another indicator of project performance was included in the model (Residual performance risk) as an intermediate variable that was significantly related with project performance (-.34). This explanation falls in with others, who propose user risk to be a component of social risks that do not impact success directly but serve to increase overall project risk (Wallace et al., 2004). The inclusion of residual risk may have dampened the expected effect of user non-support, and the strong link between partnering and success may overshadow the one between non-support and success.

To determine if the alternate model without the direct path from User Nonsupport Risk to Project Performance is viable, the path was dropped in a subsequent analysis. Bentler

(1989) suggests it is possible to delete the path if the resulting model has no significant increase to chi-square. The resulting model had no changes to any goodness of fit metric (to two decimals) and the change in chi-square was not significant. Thus, the more parsimonious model is also acceptable and consistent with previous results (Nidumolu, 1995 and 1996).

The internal validity of a model tests whether alternative explanations of the results can be provided, such as the effects of missing variables (Mitchell, 1985). In this study, project size as a missing variable was important enough to be controlled for explicitly, while random selection of respondents was a partial methodological control for other potential confounds. For example, the relationships between the user partnering, user non-support, residual performance risk, and project performance in the model may be more an artifact of their correlation with project size, rather than the presence of any effects among them. When project size (measured by the number of team members) was explicitly included in the model, the Wald test suggested that its effects on user partnering, user non-support, and project performance were not significant and should be dropped. Project size did not appear to provide an alternative explanation of the effects.

Discussion

In the last two decades, non-supportive users have been identified as one of the most encountered problems in IS development. This study confirms a relationship between non-support and project success, even if the link is not direct. To combat the problem, it is suggested that the IS staff-users relationship must be more like a predetermined partnership (Kirsch and Beath, 1996). The results demonstrate the positive effects of user partnering in the face of non-supportive user risks. The model and results also support recent work on the relation between residual performance risk and project performance (Nidumolu, 1996). The proposed model builds on theory and results from a variety of disciplines and serves to confirm earlier, more general works. The case here includes a unique look at specific risks and interventions that can be taken prior to project commencement, while others consider control techniques during the development process.

In past research, ideas have been proposed about how to increase user participation and user involvement during the system development process. For example, researchers suggested using a prototyping or evolutionary approach to increase user inputs and interactions (Alter, 1979). Tait and Vessey (1988) suggested that users be members of the design team to increase the degree of user participation. However, the introduction of user partnering focuses on conditions earlier in the process, including non-support. While software management techniques traditionally focus on control during the project development, partnering is directed at lowering the uncertainty prior to beginning development tasks (Larson, 1997).

In an early study of residual risk as an intervening variable, Nidumolu (1995) found a negative relationship (-.40) between residual performance risk and project performance, which included project process performance (learning, control, and quality of interactions) and product performance (operational efficiency, responsiveness, and flexibility). Nidumolu (1996) found a negative relationship between software performance risk and process control (-.42), but a non-significant relationship between software performance risk and product flexibility. Our study confirms these results. The confirmation of this hypothesis increases the internal of validity of Nidumolu's proposed relationship and the external validity of both studies.

With these components pieced into a single model, there is demonstration of the effectiveness of user partnering in reducing residual risks, leading to a greater level of success. The resulting model can be employed in other variations to examine the effectiveness of proposed methodologies in reducing measurable risk that impedes project performance. User partnering may also be effective in reducing other process variables, such as stakeholder conflict, open communications, and knowledge sharing. The strong effects of residual performance risk on project performance provide considerable support for the arguments advanced by software engineering managers: that a major purpose of software engineering is to effectively control and monitor the project with respect to cost, schedule, quality, and user needs (Charette, 1989; Lewis, 1995). Methodologies to be tested could include CASE, JAD, formal specification techniques, or group techniques to involve users.

Researchers may test other success measures focusing on particular areas of concern for a particular setting or organization. Potential uncertainties that could be examined include level of top management support, definition of roles and responsibilities among users and developers, organizational support and resources, and project size and complexity. Control theories on the coordination of activities during development may be incorporated into the model. Knowledge management issues may prove to be influential as well, including experiential knowledge and the presence of organizational knowledge systems and practices.

For managers, the research indicates that partnering activities can be effective for increasing user support. If so, then certain organizations should work up-front to build a foundation for collaboration between IS developers and users before disputes and problems arise. This typically begins with a meeting of the project principals to establish goals and guidelines to initiate the user partnering process. The principles make them hold team-building sessions prior to implementation of the project, ranging in length from one day to five days of intensive workshops. Such sessions often combine lectures with exercises to illustrate the principles of effective communication, teamwork, and negotiation (Mosley and Moore, 1994).

Focus then shifts to implementation of the designated project. Key people from the different parties discuss characteristics of good and bad project management, jointly establish a set of common objectives, identify potential problems areas, and develop guidelines for how they wish to resolve disputes without jeopardizing the integrity of the user partnering endeavor. The sessions generally conclude with a common charter for the project, agreement on performance criteria and how they will be measured, and procedural rules for resolving conflicts and disagreements. As a part of the process, participants establish provisions for continuous improvement. Continuous improvement is a joint effort to eliminate waste and barriers that inhibit progress and quality. Users respond quickly to development requests and share the responsibility as well as the benefits.

Within the project life cycle, these steps must all be accomplished in the early concept phase. Once the project has slipped into the development phase, user partnering needs to have already been established. Champions promoting any project should be expected to include the activities needed to accomplish good user partnering into their plans, or face the prospect of losing out to other projects that may demand attention. Cost estimates for partnering activities can be readily incorporated, and common elements of partnering can be brought into the functions of a project office or thought of as part of a common thread through the portfolio to avoid duplication of resources (Benko and McFarlan, 2003). These, however, are structural changes to an organization and must, therefore, be part of a planned initiative on the part of top management to implement best practices in effective project portfolio management, requiring the translation of corporate strategy and structure to the project level (Morris and Jamieson, 2004). Organizations without a more inclusive view of projects are not likely to succeed in promoting a user partnering effort.

Although the results of this study are encouraging, like any research, certain limitations exist. First, this study was unable to capture the dynamic nature of software projects. The present study focused only on whether partnering activities took place at the start of the project. As a consequence, the ways in which these activities interacted with other factors to influence project performance were not explored. We did not examine whether the agreements made during the partnering process were followed, or if any competing processes already in place could have accounted for the success achieved. An organization that tends to be effective in partnering aspects prior to project commencement may also tend to be effective in continuous controls and communication. Second, project performance measurement is limited to a subset of possible dimensions. Other dimensions of system success such as impacts to the organization and impacts to individuals, user satisfaction, and system usage are not considered. A third potential limitation of this study is that respondents had to reconstruct their experiences to complete the questionnaire. Such reconstruction could be subject to biases, e.g., the respondents may not remember accurately key aspects of the project. However, a survey involving recall was the only viable way to collect and analyze large-sample data.

A fourth limitation is that self-perception of success may not provide a firm basis for evaluation, although self-evaluation of performance has been widely adopted in the areas of organizational behavior and human resources management. For example, studies suggest that self-appraisal is a valid predictor of performance. Individuals are often the best judges of their own performance and can tap dimensions of performance that are overlooked by other resources (Campbell and Lee, 1988). Furthermore, self-rating reduces some of the perceptual errors made by other raters (Mabe and West, 1982). The fifth limitation is the exclusion from the study of development methods. Agile and prototyping methodologies differ greatly from traditional waterfall approaches to development in terms of user involvement and knowledge (Baskerville and Pries-Heje, 2004; Roller et al., 2004). Methods that require more frequent and active involvement can force communication structures on an organization, a transition that may be both effective and hard to manage (Nerur et al., 2005).

Future studies may use variations of the model to test specific drivers of uncertainty and the ability of methodologies to lower the presence of uncertainty. In addition to the specific variations of the model discussed earlier, extensions to the model could be examined that investigate impacts on project management and partnership formation. From a project management view, one might consider how the internal processes would have to change in order to employ user partnering and whether more strategic issues involving the management of project teams and project portfolios are impacted.

Partnership formation issues can be added as a precursor to user partnering in order to explore the foundations for establishing an effective partnership and the organizational structures that facilitate effective partnering.

References

Ajzen, I. (1991) "The Theory of Planned Behavior", O*rganizational Behavior and Human Decision Processes*, (50)2, pp.179-211.

Alter, S. (1979) "Implementation Risk Analysis", *TIMS Studies in the Management Sciences*, (13)2, pp.103-119.

Alter, S. and M. Ginzberg (1978) "Managing Uncertainty in MIS Implementation", *Sloan Management Review*, (20)1, pp. 23-31.

Anderson, J. C. and D. W. Gerbing (1988) "Structural Equation Modeling in Practice: A Review and Recommended Two-step Approach", *Psychological Bulletin*, (103)3, pp. 411-423.

Anderson, J. and R. Narashinian (1979) "Assessing Implementation Risk: A Methodological Approach", *Management Sciences*, (25)6, pp. 512-521.

Barki, H. and J. Hartwick (1989) "Rethinking the Concept of User Involvement", *MIS Quarterly*, (13)1, pp. 53-63.

Barki, H., S. Rivard, and J. Talbot (1993) "Toward an Assessment of Software Development Risk", *Journal of Management Information Systems*, (10)2, pp. 203-223.

Barki, H., S. Rivard, and J. Talbot (2001) "An Integrative Contingency Model of Software Project Risk Management", *Journal of Management Information Systems*, (17)4, pp. 37-69.

Baroudi, J. J., M. H. Olson, and B. Ives (1986) "An Empirical Study of the Impact of User Involvement on System Usage and User Satisfaction", *Communications of the ACM*, (29)3, pp. 232-238.

Baskerville, R. and J. Pries-Heje (2004) "Short Cycle Time Systems Development," Information Systems Journal, (14)3, pp. 237-264.

Beath, C. M. (1991) "Supporting the Information Technology Champion", *MIS Quarterly*, (15)3, pp. 355-372.

Beath, C. M. (1983) "Strategies for Managing MIS Projects' Transaction Cost Approach", *Proceedings of the Fourth International Conference on Information Systems*, Houston, TX: ACM/SIGBDP, New York, pp. 133-147.

Beath, C.M. and W. J. Orlikowski (1994) "The Contradictory Structure of Systems Development Methodologies: Deconstructing the IS-user Relationship in Information Engineering", *Information Systems Research*, (5)4, pp. 350-377.

Benko, C. and McFarlan, F. W. (2003) *Connecting the Dots: Aligning Projects with Objectives in Unpredictable Times*, Boston, MA: Harvard Business School Press.

Bentler, P. (1989) EQS: Structural Equation *Program Manual*, Los Angeles, CA: BMPD Statistical Software.

Bentler, P. M. and C. P.Chou (1987) "Practical Issues in Structural Modeling", *Sociological Methods and Research*, (16)1, pp. 78-117.

Boehm, B.W. (1989) *Software Risk Management*, Washington D. C.: IEEE Computer Society Press.

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

Campbell, D. J. and C. Lee, "Self-appraisal in Performance Evaluation: Development Versus Evaluation", *Academy of Management Review*, (13)2, pp.302-314.

Charette, R.N. (1989) *Software Engineering Risk Analysis and Management*, New York:

Multiscience Press Inc.
Cowan, C., C. Gray, and E. Larson (1992) "Project Partnering", *Project Management Journal*, (22)4, pp. 5-11.
Davis, G. B. (1982) "Strategies for Information Requirements Determination", *IBM Systems Journal*, (21)1, pp. 4-30.
DeLone, W. H. and E. R. McLean (1992) "Information Systems Success: The Quest for the Dependent Variable", *Information Systems Research*, (3)1, pp. 60 - 95.
Duncan, R.B. (1972) "Characteristics of Organizational Environment and Perceived Environmental Uncertainty", *Administrative Science Quarterly*, (17)3, pp. 313-327.
Frame, J.D. (1994) *The New Project Management: Corporate Reengineering and Other Business Realities*, San Francisco: Jossey Bass.
Ghiselli, E. E., J. P. Campbell, and J. P. Zedeck (1981) *Measurement Theory for the Behavioral Sciences*, San Francisco, CA: Freeman.
Glass, R.L. (1998) *Software Runaways*, New York: Prentice-Hall.
Gordon, P. (1999) "To Err is Human, to Estimate, Divine", *Information Week*, (January 18), pp. 65 – 72.
Greenbaum, J. and M. Kyng (1991) *Design at Work: Cooperative Design of Computer Systems*, Hillsdale, NJ: Lawrence Erlbaum.
Grover, V., M. J. Cheon, and J. T. C. Teng (1996) "The Effect of Service Quality and Partnership on Outsourcing of Information Systems Functions", *Journal of Management Information Systems*, (12)4, pp. 89-116.
Gummesson, E. (1987) "The New Marketing – Developing Long-term Interactive Relationships", *Long Range Planning*, (20)4, pp.10-20.
Guthrie, A. (1974) "Attitudes of User-managers Towards Management Information Systems", *Management Informatics*, (3)5, pp. 221-232.
Hair, J. F. et al. (1992) *Multivariate Data Analysis with Readings*, New York: Macmillan.
Henderson, J. C. and S. Lee (1992) "Managing I/S Design Teams: A Control Theories Perspective", *Management Science*, (38)6, pp. 757-777.
Hirschheim, R. and M. Newman (1988) "Information Systems and User Resistance: Theory and Practice", *The Computer Journal*, (31)5, pp. 398-408.
Ives B. and M. H. Olson (1984) "User Involvement and MIS Success: A Review of Research", *Management Science*, (30)5, pp. 586-603.
Jiang, J. J., W. A. Muhanna, and G. Klein (2000) "User Resistance and Strategies for Promoting Acceptance Across System Types", *Information and Management*, (37)1, 25-36.
Johnson, J. (1999) "Turning Chaos into Success", *Software Magazine*, (19)3, pp. 30.
Jones, M.C. and A. W. Harrison (1996) "IS Project Team Performance: An Empirical Assessment", *Information and Management*, (31)2, pp. 57-65.
Kettinger, W. J. and C. C. Lee (1994) "Perceived Service Quality and User Satisfaction with the Information Services Function", *Decision Sciences*, (25)5/6, pp. 737-766.
Kirsch, L.J. and C. M. Beath "The Enactments and Consequences of Token, Shared, and Compliant Participation in Information Systems Development" *Accounting, Management and Information Technologies*, (6)4, pp. 221-254.
Kline, R. B. (1998) *Principles and Practice of Structural Equation Modeling*, New York: Guilford Press.
Larson, E. W. (1997) "Partnering on Projects: A Study of the Relationship Between Partnering Activities and Project Success", *IEEE Transactions on Engineering Management*, (EM-44)2, pp. 88-195.
Lewis, J. (1995) *Fundamentals of Project Management*, Amacom Books.
Linberg, K. R. (1999) "Software Developer Perceptions about Software Project Failure: A

Case Study", *The Journal of Systems and Software*, (49)2/3, pp. 177-192.

Lyytinen K. and R. Hirschheim (1987) "Information Systems Failures – A Survey and Classification of the Empirical Literature", *Oxford Surveys in Information Technology*, (4), pp. 257-309.

Lyytinen, K., L. Mathiassen, and J. Roppopen (1998) "Attention Shaping and Software Risk – A Categorical Analysis of Four Classical Risk Management Approaches", *Information Systems Research*, (9)3, pp. 233-255.

Mabe III, P.A. and S. G. West (1982) "Validity of Self-evaluation of Ability: A Review and Meta-analysis", *Journal of Applied Psychology*, (67), pp. 280-296.

McFarlan, F. W. (1981) "Portfolio Approach to Information Systems", *Harvard Business Review*, (59)5, pp. 142-150.

McKeen, J.D., T. Guimaraes, and J. C. Wetherbe (1994) "The Relationship between User Participation and User Satisfaction: An Investigation of Four Contingency Factors", *MIS Quarterly*, (18)3, pp. 427-451.

McKeen, J. D. and T. Guimaraes (1997) "Successful Strategies for User Participation in Systems Development", *Journal of Management Information Systems*, (14)2, pp. 133-150.

Mitchell, T. R. (1985) "An Evaluation of the Validity of Correlational Research Conducted in Organizations", *Academy of Management Review*, (10)2, pp. 192-205.

Morris, P. and Jamieson, A. (2004) *Translating Corporate Strategy Into Project Strategy*, Newton Square, PA: Project Management Institute.

Mosley, D. and C. Moore, (1994) "TQM and Partnering: An Assessment of Two Major Change Strategies", *PMNETwork*, (8)9, pp. 22-25.

Mulaik, S. A. et al. (1989) "Evaluation of Goodness-of-fit Indices for Structural Equation Models", *Psychological Bulletin*, (105)3, pp. 430-445.

Nerur, S., R. Mahapatra, and G. Mangalaraj (2005) "Challenges of Migrating to Agile Methodologies," *Communications of the ACM*, (48)5, pp. 73-78.

Newman, M. and D. Robey (1992) "A Social Process Model of User-analyst Relationships", *MIS Quarterly*, (16)2, pp. 249-266.

Nidumolu, S. (1995) "The Effect of Coordination and Uncertainty on Software Project Performance: Residual Performance Risk as an Intervening Variable", *Information System Research*, (6)3, pp. 191-219.

Nidumolu, S. (1996) "A Comparison of the Structural Contingency and Risk-based Perspectives on Coordination in Software-Development Process", *Journal of Management Information Systems*, (13)2, pp. 77-113.

Nunnally, J.C. (1978) *Psychometric Theory*, New York: McGraw-Hill.

Petty R. E. and J. T. Cacioppo (1996) *Attitudes and Persuasion: Classic and Contemporary Approaches*, Boulder, CO: Westview Press.

Pruitt, D. G. (1981) *Negotiation Behavior*, New York: Academic Press.

Robey, D. and M. Newman (1996) "Sequential Patterns in Information Systems Development: An Application of a Social Process Model", *ACM Transactions on Information Systems*, (14)1, pp. 30-63.

Roller, D., O. Eck, and S. Dalakakis (2004) "Knowledge-based Support of Rapid Product Development," *Journal of Engineering Design*, (15)4, pp. 367-388.

Saleem, N. (1996) "An Empirical Test of the Contingency Approach to User Participation in Information System Development", *Journal of Management Information Systems*, (13)1, pp. 145-166.

Schwalbe, K. (2000), *Information Technology Project Management*, 2000 Course Technology.

Segars, A. H. and V. Grover (1993) "Re-examining Perceived Ease of Use and Usefulness: A Confirmatory Factor Analysis", *MIS Quarterly*, (17)4, pp.517-525.

Salancik, G. and J. Pfeffer (1974) "The Bases and Use of Power in Organizational Decision-Making", *Administrative Science Quarterly*, (19)4, pp. 453-473.

Shumskas, A. S. (1992) "Software Risk Mitigation" in G.G. Schulmeyer and J. I. McManus (Eds.) *Total Quality Management for Software*, New York: Van Nostrand Reinhold.

Tait, P. and I. Vessey (1988) "The Effect of User Involvement on Systems Success: A Contingency Approach", *MIS Quarterly*, (12)1, pp. 91-107.

Thamhain, H. J. and D. L. Wilemon (1987) "Building High Performing Engineering Project Teams", *IEEE Transactions on Engineering Management*, (EM-34)3, pp. 130-137.

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

Van De Ven, A. H., A. L. Delbecq, and R. Koening (1976) "Determinants of Coordination Modes within Organizations", *American Sociological Review*, (41)2, pp. 322-338.

Wallace, L., M. Keil, and A. Rai (2004) "How Software Project Risk Affects Project Performance: An Investigation of the Dimensions of Risk and an Exploratory Model", *Decision Sciences*, (35)2, pp. 289-322.

Wateridge, J. (1995) "IT Projects: A Basis for Success", *International Journal of Project Management*, (13)3, pp. 169-172.

Williams, L. J. and J. T. Hazer (1986) "Antecedents and Consequences of Satisfaction and Commitment in Turnover Models: A Reanalysis Using Latent Variable Structural Equation Methods", *Journal of Applied Psychology*, (71)2, pp. 219 – 231.

Youngs, G. A., Jr. (1986) "Patterns of Threat and Punishment Reciprocity in a Conflict Setting", *Journal of Personality and Social Psychology*, (51)3, pp. 541-546.

About the Authors

Dr. James Jiang is Professor of Management Information Systems at the University of Central Florida, U.S. He also holds the honorary Jin-Ding Professor at the National Central University in Taiwan. He was visiting research professor in Taiwan National Tshing-Hwa University, Taiwan National Chung-Cheng University, National Taiwan University, and City University of Hong-Kong. He obtained his masers degrees in Applied Mathematics, Computer Sciences, and Management Sciences. His Ph.D. in Information Systems was granted by the University of Cincinnati. His research interests are IS project management and IS service quality management. He has published over 100 academic articles in these areas.

Dr. Gary Klein is the Couger Professor of Information Systems at the University of Colorado in Colorado Springs. He obtained his Ph.D. in Management Science from Purdue University. Before that time, he served with the company now known as Accenture in Kansas City and was director of the Information Systems Department for a regional financial institution. His research interests include project management, system development, technology transfer, and mathematical modeling with over 100 academic publications in these areas. He teaches programming, analysis and design, statistics, management science, and knowledge management courses. In addition to being an active participant in international conferences, he has made professional presentations on *Decision Support Systems* in the U.S. and Japan where he once served as a guest professor to Kwansei Gakuin University. He is an active member of the Institute of Electrical and Electronic Engineers, the Association for Computing Machinery, the Decision Science Institute, and the Project Management Institute.

Dr. Houn-Gee Chen is a professor of Information Management at National Tsing-Hwa University, Taiwan. He earned his PhD in Industrial Engineering from the University of Wisconsin-Madison. His research interests include information technology management, project management, and information resources management. He has published numerous academic articles in journals such *as IEEE Transactions on Systems, Man, and Cybernetics, International Journal of Man-Machine Studies, International Journal of Production Research, Information & Management, Journal of Manufacturing Systems, Computers and Operation Research, International Journal of Business, International Journal of Project Management, Project Management Journal, Decision Sciences, Journal of Information Science, Communications of ACM, and International Journal of Information Management.*

Journal of the Association for Information Systems
ISSN: 1536-9323