



## Complexity of Information Systems Development Projects: Conceptualization and Measurement Development

WEIDONG XIA & GWANHOO LEE

To cite this article: WEIDONG XIA & GWANHOO LEE (2005) Complexity of Information Systems Development Projects: Conceptualization and Measurement Development, Journal of Management Information Systems, 22:1, 45-83

To link to this article: <http://dx.doi.org/10.1080/07421222.2003.11045831>



Published online: 08 Dec 2014.



Submit your article to this journal [↗](#)



Article views: 116



View related articles [↗](#)

# Complexity of Information Systems Development Projects: Conceptualization and Measurement Development

WEIDONG XIA AND GWANHOO LEE

WEIDONG XIA is an Assistant Professor in the Carlson School of Management at the University of Minnesota. His research focuses on IT strategy, organizational capabilities, and business alignment; IT project complexity and flexibility; and IT adoption decisions. His writings have been published or are forthcoming in a number of academic journals and international conferences, including *Communications of the ACM*, *Decision Sciences*, *European Journal of Information Systems*, *International Journal of Career Development*, *Journal of Management Information Systems*, *MIS Quarterly*, *Journal of Statistics and Management Systems*, *Journal of End-User Computing*, and *International Conference on Information Systems*. At the University of Minnesota, he has taught nine different courses at Ph.D., executive, MBA, and undergraduate levels. He has served as MIS undergraduate and MBA program coordinators. He has been working closely with executives from companies such as 3M, AG Edwards, Ahold, Cargill, McDonald's, Medtronic, Musicland, Northwest Airlines, Pillsbury, ShopNBC, St. Paul Cos., Target, Union Pacific, U.S. Bank, and U.S. Cellular on issues related to his research areas. He received his Ph.D. in Information Systems from the University of Pittsburgh.

GWANHOO LEE is an Assistant Professor, Kogod Endowed Fellow, and UPS Scholar in the Kogod School of Business at American University in Washington, DC. His research focuses on IS development project management, IT complexity and flexibility, technology adoption, and global software teams. His research has been published or is forthcoming in several academic journals and at conferences, including *Communications of the ACM*, *European Journal of Information Systems*, *Americas Conference on Information Systems*, *Hawaii International Conference on System Sciences*, and *International Conference on Information Systems*. He received his Ph.D. in Management Information Systems from the University of Minnesota.

**ABSTRACT:** This paper conceptualizes and develops valid measurements of the key dimensions of information systems development project (ISDP) complexity. A conceptual framework is proposed to define four components of ISDP complexity: structural organizational complexity, structural IT complexity, dynamic organizational complexity, and dynamic IT complexity. Measures of ISDP complexity are generated based on literature review, field interviews, and focus group discussions. The measures are then refined through a systematic process and are tested using confirmatory data analyses with survey responses from 541 ISDP managers. Results support the final measurement model that consists of a second-order factor of ISDP complexity, four distinct first-order factors, and 15 measurement items. The measurement adequately satisfies the criteria for unidimensionality, convergent validity, discriminant validity, reliability, factorial invariance across different types of ISDPs, and nomo-

logical validity. Implications of the study results to research and practice as well as limitations of the study and directions for future research are discussed.

**KEY WORDS AND PHRASES:** complexity, conceptual framework, confirmatory factor analysis, information systems development project, scale development, scale validation.

INFORMATION SYSTEMS (IS) DEVELOPMENT is inherently complex because it must deal with not only technological issues but also organizational factors that, by and large, are outside of the project team's control [13, 40, 59, 60]. For example, as organizations increasingly depend on IS for one-stop customer service capabilities and cross-selling opportunities, new system development efforts must be seamlessly integrated with other existing systems, introducing a system-level complexity that is often constrained by the existing technology architecture and infrastructure [77]. In addition, as both information technology (IT) and business environments are fast changing, it becomes increasingly difficult to determine business requirements with any certainty, making system development a progressively more dynamic and complex process [73, 81].

The complexity of IS development is manifested by the historically high failure rate of information systems development projects (ISDPs). In the past four decades, there have been many reports about ISDP failures in various organizations across industries (e.g., [33, 47]). For example, the Standish Group reported that U.S. companies spent more than \$250 billion each year in the early 1990s on IS projects, with only 16.2 percent considered successful [67]. The Standish Group's 2001 report indicated that U.S. companies invested four times more money in IS projects in 2000 than they did annually in the 1990s; however, only 28 percent of the projects could be considered successful [68]. The results suggest that, although much improved, the success rate of IS projects is still very low. As organizations increasingly invest in IS with the intention to enhance top-line growth and bottom-line savings, IS project failures have significant organizational consequences, in terms of both wasted critical resources and lost business opportunities. Therefore, there is a strong economic incentive for companies to improve IS project performance.

In this paper, IS development refers to the analysis, design, and implementation of IS applications/systems to support business activities in an organizational context. ISDPs are temporary organizations that are formed to perform IS development work, including new applications/systems yet to be installed, as well as enhancement of existing applications/systems [69]. New applications/systems include both in-house application/systems and off-the-shelf packaged software.

ISDP failures are not isolated incidents, but rather, they recur with a great deal of regularity in organizations of all types and sizes [25, 26, 35, 68]. Many experts believe that ISDPs are uniquely more complex than other types of projects such as

construction and product development projects. Edward W. Felten, a computer science professor at Princeton University, was recently quoted: “A corporate computer system is one of the most complex things that humans have ever built” [5, p. 118]. Brooks, in his classic book, *The Mythical Man-Month*, states that “software entities are more complex for their size than perhaps any other human construct” and that “software systems differ profoundly from computers, buildings, or automobiles” [13, p. 182]. He contends that the complexity of software is an essential property, not an accidental one. Such essential complexity is unique to software development because software is invisible and unvisualizable, and is subject to conformity and continuous changes. Brooks concludes, “Many of the classical problems of developing software products derive from this essential complexity” [13, p. 183].

Projects are temporary organizations within organizations [20]. Different types of projects demonstrate different contingency characteristics that require different management approaches. Some researchers argue that one of the difficulties in advancing theory development on project complexity may stem from the conventional approach of developing “one size fits all” theories (e.g., [6, 24, 66]). In a recent study, Shenhar [65] shows “one size does not fit all” and calls for a more contingent approach to the study of project management. Given the call for contingency theories for different types of projects and the unique characteristics of ISDPs, it is necessary to develop theories for understanding and managing ISDP complexity. By building on the general project complexity concepts, research on ISDP complexity may provide new insights that will contribute to the general project management literature.

Managing ISDP complexity has become one of the most critical responsibilities of IS managers and executives [31, 61]. However, IS organizations have encountered great difficulties in coping with the progressively increasing complexity of ISDPs [10, 52]. Before we can develop effective strategies to control and manage ISDP complexity, it is necessary that we understand the project characteristics that constitute ISDP complexity and be able to use those characteristics to assess the complexity of an ISDP. However, research on ISDP complexity to date has mostly been conceptual and anecdotal. The ISDP complexity construct has often been used without precise definitions and appropriate operationalizations. Only a few researchers have attempted to develop measures related to ISDP complexity [7, 76]. However, no systematic, comprehensive frameworks and validated measurements of ISDP complexity have been reported.

This research represents a first step toward conceptualizing and developing measures of the key dimensions of ISDP complexity. The research was conducted using a systematic four-phase measurement development process involving multiple data sources and research methods. The conceptual framework and the initial measures of ISDP complexity were developed and refined through literature reviews, field interviews, focus group discussions, a sorting procedure, and two pilot tests. The conceptual framework and measures were then tested using survey data provided by 541 ISDP managers. The four-component conceptualization of the ISDP complexity construct was tested through analyzing the factor structures underlying the measures. Confirmatory analysis techniques were used to conduct a comprehensive set of mea-

surement tests, including reliability, unidimensionality, convergent validity, discriminant validity, factorial invariance across three types of ISDPs, and nomological validity. Using multiple data sources, multiple methods and multiple measurement validation criteria helped triangulate the analyses and enhance the quality and credibility of the results.

The development of such conceptual frameworks and measurements makes significant contributions to both research and practice. For researchers, well-developed conceptual frameworks of ISDP complexity provide the basis for consistently defining the construct across studies so that results of different studies can be meaningfully compared. Empirically developed theories involving ISDP complexity are not possible without valid measures of ISDP. In addition, theoretical development on ISDP complexity will contribute to the general project management literature by providing insights about project complexity under a specific contingent context. For practitioners, sound conceptual frameworks provide a critical language through which managers can describe and communicate the key dimensions of ISDP complexity. Operational indicators provide managers with the necessary tools to assess the complexity of specific ISDPs. The ability to assess ISDP complexity is a prerequisite for developing effective strategies to control and manage ISDP complexity.

## Theoretical Background

---

COMPLEXITY HAS BEEN STUDIED in various streams of research. Table 1 summarizes how prior literatures conceptualized and operationalized the complexity construct in several different contexts. Although the ISD literature is most relevant to our study, little research has attempted to rigorously conceptualize and measure complexity of ISD. Therefore, we expanded our literature review to other related literatures in order to bring in a broader perspective. These literatures include the project management literature, the task complexity literature, and the software project risk factor literature.

## IS Development Complexity

McKeen et al. [48] propose two important complexities in IS development: task complexity and system complexity. Task complexity is defined as the uncertainty and ambiguity that surround the practice of business, which are originated from the user's environment. System complexity is defined as the uncertainty and ambiguity that surround the practice of systems development, which are originated from the developer's environment. Ribbers and Schoo [56] propose three dimensions of system implementation complexity. The first dimension is variety, which is the multiplicity of project elements such as number of sites affected by the system implementation. The second dimension is variability, which refers to changes in project elements such as changes in project goal and scope. The third dimension, integration, taps into coordination of various project elements.

Table 1. Summary of Selected Prior Research

Research construct	Author(s)	Dimensions of complexity				Operationalized/implied measures
		Structural	Dynamic	Organization	Technology	
ISD complexity	McKeen et al. [48]		Ambiguity, uncertainty	Task (user's environment) complexity	System (developer's environment) complexity	Technology platform, design techniques and computing languages, project size, development methodologies, project management and control, dynamics of project team, required degree of system integration.
	Ribbers and Schoo [56]	Variety (number of elements), integration	Variability (dynamics)			Number of sites, project staff availability, goal and scope changes.
System complexity	Garmus and Herron [27]					Complexity of data communication, distributed data processing, transaction rate, online update, complex processing, multiple sites, and installation ease.
	Meyer and Curley [49]	Depth, scope, breadth	Rate of change, ambiguity	Knowledge (problem) complexity	Technological complexity	Diversity of platforms, diversity of technologies, database intensity, network intensity, scope of programming effort, system integration effort.
Project complexity	Tait and Vessey [70]	Multiplicity, interactions, lack of a structure/model				Difficulty of determining user requirements, complexity of processing; complexity of system design.
	Baccarini [2]	Differentiation, interdependency		Organizational complexity	Technological complexity	Number of organizational units, interdependencies between organizational units, number and diversity of inputs/ outputs, number of subcontractors, technological interdependencies.

(continues)

Table 1. Continued

Research construct	Author(s)	Dimensions of complexity			Operationalized/implied measures
		Structural	Dynamic	Organization	Technology
Task complexity	Clark [19] Pich et al. [55]	Extent, scope Interaction, multiplicity Structural			
	Shenhar and Dvir [64]		Uncertainty	System scope complexity	Technological uncertainty
	Tatikonda and Rosenthal [72] Turner and Cochrane [74] Williams [80]	Interdependence	Novelty		
			Uncertainty		
		Structural (multiplicity, interdependency)	Uncertainty		
	Wozniak [83]				
Task complexity	Campbell [16]	Multiplicity of paths, multiplicity of outcomes, conflicting interdependence among paths	Uncertain links among paths and outcomes		
	Wood [82]	Components, coordination	Dynamic		

Extent of project scope.

Technology novelty, project objective novelty, project difficulty.  
Uncertainty in goals, uncertainty in methods.  
Number of elements, interdependence of elements, uncertainty in goals, uncertainty in methods.  
Criticality of project, level of technology required, scope, project team, relationships with external vendors, dollar value of project.

ISDP risk	Boehm [12] Barki et al. [7] Barki et al. [8]	Application complexity	Personnel shortfall, requirements changes. Hardware/software/database complexity, number of links to existing systems, number of links to future systems, project size (person-days), technological novelty, cross-functional project team composition, project team's lack of expertise, lack of user support, conflicts between user units, multiple external contractors and vendors, system integration, extent of changes brought, magnitude of potential loss. Organization changes caused by system, change in organizational structure, top management support, user support, technology novelty.
	McFarlan [47]		Personnel shortfall, insufficient expertise, requirement changes. Lack of required skills in project personnel, insufficient staffing, lack of user commitment, lack of top management support, technological novelty, changes in scope/objectives, change in IT architecture, multiple external contractors and vendors, coordination of multiple user units, conflict between user departments, change in users' needs.
	Ropponen and Lyytinen [58] Schmidt et al. [59]		Use of new technology, technical complexity, immaturity of technology, use of technology not used in prior projects.
	Wallace et al. [76]		



Some researchers have conceptualized technological or system complexity in the context of IS development. Tait and Vessey [70] measure system complexity in terms of the difficulty in determining the information requirements of the system, the complexity of processing, and the overall complexity of the system design. Meyer and Curley [49] define technology complexity of an expert system as a composite measure of diversity of technologies, database intensity, and systems integration effort.

The function point analysis literature [27] assessed the complexity of software systems by taking into account 14 general system characteristics. These general system characteristics included complexity with regard to data communication, distributed data processing, transaction rate, online update, complex processing, multiple sites, and installation ease. They are the basis for the value adjustment factor that is used for adjusting the basic complexity of system inputs, outputs, inquiries, logical files, and interface files.

### Project Complexity

Based on a review of the project complexity literature, Baccarini [2] defines project complexity in terms of the number of varied elements and the interdependency between those elements. Following this definition, he proposes two types of project complexity: organizational complexity and technological complexity. Organizational complexity refers to the number of, and relationships between, hierarchical levels, formal organizational units, and specializations. Technological complexity refers to the number of, and relationships between, inputs, outputs, tasks, and technologies. Turner and Cochrane [74] propose uncertainty as another dimension of project complexity, which is the extent to which the project goals and means are ill defined and are subject to future changes. Uncertainty in systems requirements/scope and uncertainty in new IT are examples of goal and mean uncertainties, respectively [10, 11, 21, 44, 45, 59].

By integrating the dimensions proposed by Baccarini [2] and Turner and Cochrane [74], Williams [80] proposed two distinct aspects of project complexity: structural complexity and uncertainty-based complexity. He contends that a complete picture of project complexity includes not only structural complexity originated from the underlying structure of the project but also uncertainty-based complexity originated from changes in project environments. The distinction between structural and uncertainty-based complexity is important because they require different types of organizational capabilities; organizations tend to handle structural complexity relatively well while they are not well equipped to effectively deal with uncertainty-based complexity that conventional management tools cannot cope with [62]. Shenhar and Dvir [64] argue that although the structural complexity dimension is based on the scope or a hierarchical framework of systems and subsystems, the uncertainty-based complexity dimension is based on the level of technological uncertainty at the initiation stage of the project.

Various complexity factors and measures have been proposed in the general project management literature, with nine complexity factors including criticality of the project,

level of technology required, scope, project team, relationships with external vendors, and dollar value of project [83].

### Task Complexity

The task complexity literature has identified a variety of dimensions and task characteristics that constitute task complexity. In particular, Wood [82] and Campbell [16] have had significant influence on the subsequent research on task complexity. Wood [82] defines three types of task complexity: component, coordinative, and dynamic. Component complexity of a task is a function of the number of distinct acts that need to be executed in the performance of the task and the number of distinct information cues that must be processed in the performance of those acts. Coordinative complexity refers to the nature of the relationships between task inputs and task outputs. The form and strength of the relationships between information cues, acts, and products are aspects of coordinative complexity. As the requirements for timing, frequency, intensity, and location of acts become more complex, the difficulty for coordination increases. Dynamic complexity is caused by changes in the states of the task environments.

Campbell [16] views complexity as a function of four task characteristics. He proposes that task complexity increases when (1) only one path leads to goal attainment while multiple paths exist, (2) multiple desired outcomes are required, (3) there exists conflicting interdependence among paths, and (4) the connection between path activities and desired outcomes cannot be established with certainty.

### Software Project Risk Factors

According to McFarlan [47], failure to assess and manage individual IS project risk is a major source of software development problems. ISDP risk and ISDP complexity are two related but different concepts. IS project risk factors are related to the probabilities of the negative conditions or events and the magnitude of losses associated with project failure. ISDP complexity, on the other hand, refers to the characteristics of the project that constitute difficulties for the development effort. In the context of ISDPs, although risk factors and complexity both represent some negative connotations, complexity tends to capture project characteristics that are inherently present in projects, whereas risk factors represent possible events or conditions that cause project failures. The two constructs are obviously related. As the inherent project characteristics, indicators of ISDP complexity represent the underlying factors that may drive project risks.

The IS project risk literature has primarily focused on identifying prioritized lists of project risk factors, although a few researchers have proposed contingency models that would help manage and reduce risks in software projects [39, 45]. Example risk factors include cross-functional project team composition, project team's lack of expertise, insufficient project staffing, extent of organizational changes brought, change in end-user information needs, change in organizational structure, change in IT architecture,

technological novelty, lack of user support, coordination of multiple user units, conflicts between user units, multiple external contractors and vendors, system integration, and magnitude of potential loss [7, 8, 47, 58, 59].

A few measures for assessing complexity as part of software development risk and software project risk have been reported in the literature. For example, Barki et al. [7] developed measures for application complexity as a dimension of software project risk. These measures tapped into technical complexity and number of links to existing and future systems. Wallace et al. [76] developed measures for project complexity risk. These measures include novelty and immaturity of technology.

In summary, our review of the IS literature suggests that, although the ISDP complexity construct has been frequently mentioned, there exist no well-developed frameworks that can be used to delineate its conceptual meanings. No systematically validated measures for ISDP complexity have been reported. However, a careful review on the prior literatures revealed a few common dimensions and characteristics of complexity that can be adapted for conceptualizing and measuring ISDP complexity. As shown in Table 1, two dimensions of ISDP complexity emerged in various streams of research: (1) structural versus dynamic complexity and (2) organizational versus technological complexity.

### A Conceptual Framework for ISDP Complexity

*MERRIAM-WEBSTER'S COLLEGIATE DICTIONARY* defines complexity as "the quality or state of being complex," and, in turn, defines complex as "composed of two or more parts." However, there is no agreed-upon definition of ISDP complexity in the literature. By building on the existing literature and incorporating insights gained from ISDP managers through field interviews and focus group discussions, we define ISDP complexity as the ISDP's state of consisting of many varied organizational and technological elements that are interrelated and change over time. Based on this definition, we developed a conceptual framework of ISDP complexity and defined key dimensions of the construct as shown in Figure 1. The first dimension of the conceptual framework captures whether the complexity is about the structural setup or the dynamic aspects of the project; the second dimension captures whether the complexity is about the organizational aspects or the technological aspects of the project. In this framework, each dimension suggests two distinct aspects of ISDP complexity rather than being a continuum-based variable.

The first dimension, the distinction between structural and dynamic complexity, is supported by prior research [16, 49, 56, 64, 72, 80, 82]. Consistent with prior research, we define structural complexity as (1) variety [56], multiplicity [16, 55, 70, 80, 82], and differentiation [2] of project elements; and (2) interdependency [2, 7, 16, 72, 80], interaction [55, 70], coordination [82] and integration [56] of project elements. ISDPs typically involve a number of organizational and technological elements including the existing systems, infrastructure, new technology, user units, stakeholders, the project team, vendors, and external service providers. As the number of project elements increases, it becomes more difficult to monitor and control the

Organizational	Structural Organizational Complexity (SORG)	Dynamic Organizational Complexity (DORG)
Technological	Structural IT Complexity (SIT)	Dynamic IT Complexity (DIT)
	Structural	Dynamic

Figure 1. A Conceptual Framework of ISDP Complexity

project. Interdependencies among these elements make it even more difficult to predict the project's process and outcome. According to Leveson [43], the problems in building complex systems often arise in the interfaces between the various elements such as hardware, software, and human components.

We define dynamic complexity as uncertainty [16, 48, 64, 74, 80], ambiguity [48, 49], variability [56], and dynamism [49, 82], which are caused by changes in organizational and technological project environments. Changes may result from either the stochastic nature of the environment or a lack of information and knowledge about the project environment. As these changes occur, the cause-and-effect relationship becomes ambiguous and nonlinear. Dynamic complexity is increasingly relevant to and critical for ISDPs because their environments, both business and IT, are changing with unprecedented rapidity [11].

The second dimension, the distinction between organizational and technological complexity, is well informed by the ISD literature [48, 49] and the project management literature [2, 64]. We define organizational complexity as the complexity of organizational environments surrounding the project. Organizational elements of an ISDP include user groups, top management, project team, external contractors and vendors, organizational structure, and business processes. We define technological complexity as the complexity of technological environment of the ISDP [2, 48, 49, 64]. McKeen et al. [48] proposes that technological complexity (system complexity) includes the complexity of technology platform, design techniques and computing languages, development methodologies, and system integration. Similarly, Meyer and Curley [49] proposes that the technological complexity in the context of expert systems consists of such technological variables as diversity of platforms, diversity of technologies, database intensity, and systems integration effort.

Combining the two dimensions, we define four components of ISDP complexity: structural organizational complexity (SORG), structural IT complexity (SIT), dynamic organizational complexity (DORG), and dynamic IT complexity (DIT). We define the SORG component as the multiplicity and interdependency of organizational elements of an ISDP. In particular, various stakeholders constitute important organizational elements of an ISDP. These stakeholders include end users [8, 34, 59], project team [7, 8], top management [7, 34, 39, 45, 59], and external contractors and vendors [2, 7, 34, 59, 83].

We define the SIT component as the multiplicity and interdependency of technological elements of an ISDP. These technological elements include technology platform [49], software environments [49], data processing requirements [27], and other integrated systems [7, 8, 48, 49].

We define the DORG component as the rate and pattern of changes in the ISDP organizational environments, including changes in user information needs [12, 58, 59], business processes [8, 60], and organizational structures [7, 8, 60]. In today's hypercompetitive business environment, users' information needs change frequently, which requires substantial coordination and rework [12, 58, 59]. Business process changes increase ISDP complexity because they necessitate dynamic alignment between business processes and IS under development. In addition, changes in organizational structure cause changes in the information flows and systems scope of the project, which in turn increase the complexity of the ISDP.

Finally, we define the DIT component as the rate and pattern of changes in the IT environment of the ISDP, including changes in IT infrastructure [48, 60], architecture [48, 59] and software development tools [48]. As application systems are developed under the constraints of the existing IT infrastructure and IT architecture of the organization, changes in IT infrastructure and IT architecture cause significant uncertainty in ISDPs [14, 23, 77]. In addition, adoption of new development tools during an ongoing ISDP causes interruptions, because the developers must take time to learn the new tools and adjust their initial analysis and design to fit in the new development environments.

ISDP managers helped us a great deal in confirming the validity of the conceptual framework of ISDP complexity. One particular area in which the inputs from the field interviews were helpful was the validity of structural complexity. While the prior literature supported both the validity of structural complexity as a single concept combining component and coordinative complexities (e.g., [80]) and the validity of component complexity and coordinative complexity as two separate concepts (e.g., [82]), ISDP managers suggested that the combined structural complexity makes more sense. The rationale was that, in practice, component complexity and coordinative complexity tended to be highly correlated. In addition, using a combined construct would provide a more parsimonious framework of ISDP complexity that would be easier to understand and used in practice. ISDP managers in our interviews also confirmed that this distinction between organizational and technological complexity is significant and meaningful, because, due to different degrees of controllability, organizational and technological complexities require different project capabilities and have different implications for project management.

## Measurement Development Process

RECOGNIZING THAT IT IS IMPOSSIBLE TO CAPTURE all dimensions and characteristics of ISDP complexity, we intend to develop a parsimonious set of measures that capture the most critical aspects and provide a starting point for developing measurement theories of ISDP complexity. We used a systematic four-phase process involving a

variety of methods to develop and validate the measurement of ISDP complexity. This four-phase process was developed based on Churchill [18] and Sethi and King [63]. As shown in Figure 2, the four phases were (1) conceptual development and initial item generation, (2) conceptual refinement and item modification, (3) survey data collection, and (4) data analysis and measurement validation. Table 2 shows the major roles and results of research participants in developing the ISDP complexity measure throughout the four stages.

### Phase 1—Conceptual Development and Initial Item Generation

In this phase, the conceptual framework and an initial pool of measurement items were first developed through literature reviews. In developing the measures, whenever possible, we adapted relevant measures in the literature. Then the initial framework and measurement items were confirmed and modified through field interviews and focus group discussions. Twelve interviews were conducted with ISDP managers who had, on average, more than ten years of work experience. A focus group discussion with 45 IS managers using nominal group techniques was conducted to independently generate a ranked list of ISDP complexity items. By combining literature reviews with field interviews and focus group discussions, we attempted to ensure the face and content validity of the measures—that is, to ensure that the measures covered the appropriate scope/domain of ISDP complexity. As a result, a total of 30 items were generated for the initial pool of measures. Appendix Table A1 summarizes these 30 initial items along with their literature sources and measurement refinement results.

### Phase 2—Conceptual Refinement and Item Modification

The framework and initial pool of 30 measures resulted from Phase 1 were refined and modified through a sorting procedure and two pilot tests.

#### Sorting procedure

A sorting procedure was used to qualitatively assess the face validity and the construct validity of the initial items [51]. Four IS researchers with an average of eight years of IS work experience participated in the sorting procedure. Each item in the initial pool was printed on a 3×5-inch index card. In the sorting procedure, each judge was asked to carefully read the card and place it in one of the four components of ISDP complexity as defined in Figure 1. An additional category, “too ambiguous/unclear,” was included for the judges to put a card into if they felt it did not belong to any of the four predefined categories. Prior to actually sorting the cards, the judges read a standard set of instructions. Each judge then individually sorted the ISDP complexity item cards. After completing the sorting procedure, they explained why they sorted cards (if any) into the “too ambiguous/unclear” category. As shown in Appendix Table A1, four items were dropped after the sorting procedure. These four items were commonly placed in the “ambiguous/unclear” category by the judges. For ex-

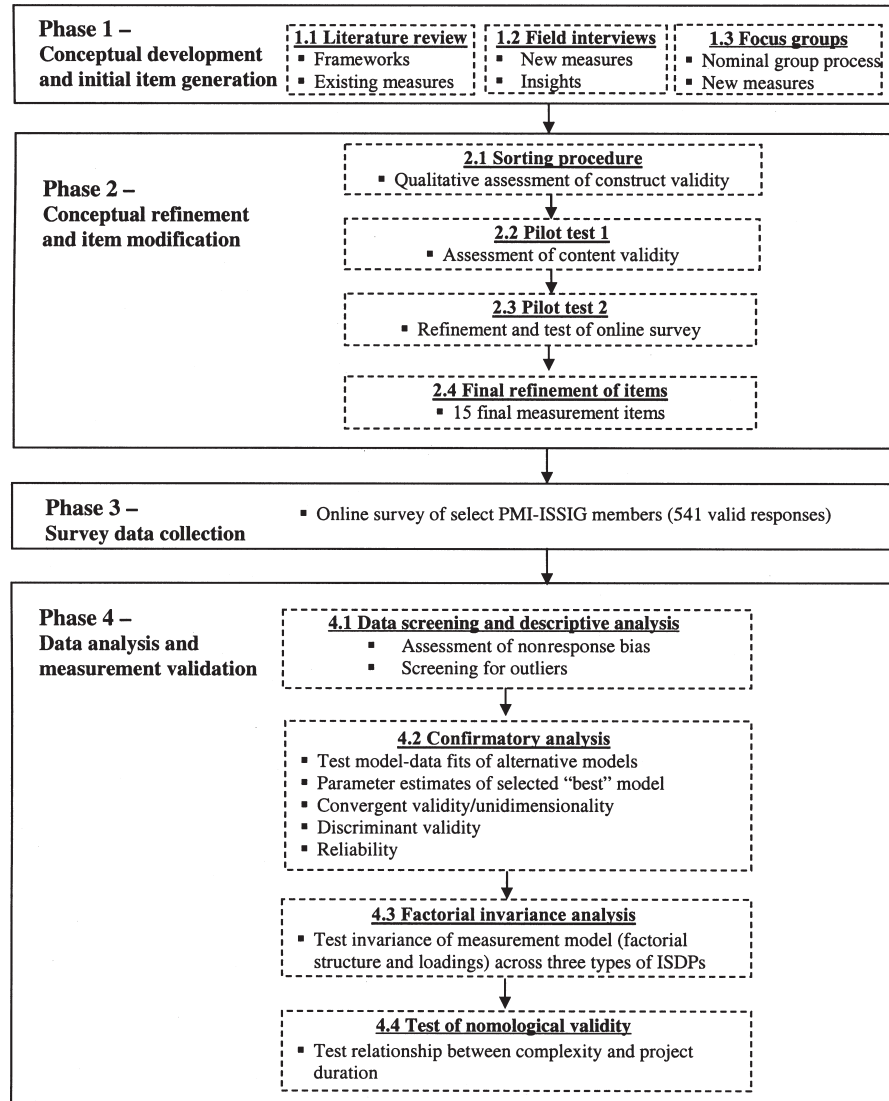


Figure 2. A Four-Phase Process of Measure Development and Validation

ample, the judges felt that ISDPC8 (“The project was given a fixed deadline”) and ISDPC13 (“It was a new project that we had never done before”) were indirectly associated with the complexity construct. ISDPC18 (“The project required many person-hours”) and ISDPC28 (“There would have been a significant effect on the business if this project had failed”) seemed to be too broad to fit into any of the four complexity components.



Table 2. Roles of Research Participants

Phase	Participants	Number of participants	Roles	Results
Field interviews	ISDP managers	12	Confirmation of the framework, generation of initial items.	Conceptual framework, initial items.
Focus group discussion	IS managers	45	Generation of a ranked list of initial items.	
Sorting procedure	IS researchers	4	Assessment of construct validity of initial items using a sorting procedure.	Four items dropped.
Pilot test 1	ISDP managers, IS researchers	7 (4,3)	Assessment of content validity.	Five items dropped, two items combined.
Pilot test 2	ISDP managers	15	Refinement of items and survey instrument.	Editorial changes to survey instrument.
Survey data collection	ISDP managers	565	Response to the final survey instrument.	541 valid responses.



### Pilot Test 1

To further validate the relevance, coverage, and clarity of the measurement items, we conducted two pilot tests. The first pilot test was conducted through one-hour individual interviews with four ISDP managers and three IS researchers who had more than five years of work experience. In the interview, the participant first filled out a questionnaire regarding the importance and relevance of each item to ISDP complexity. They were then asked to identify items that appeared to be inappropriate or irrelevant to ISDP complexity. Participants also made suggestions for improving the relevance, coverage, understandability, and clarity of the items. As shown in Appendix Table A1, five items were dropped based on the results of the first pilot test. For example, the ISDP managers felt that ISDPC10 (“The project was a major capital investment”) was remotely associated with project complexity. Other dropped items were judged to be less critical measures of ISDP complexity. Two items—ISDPC19 (“The project involved multiple external vendors”) and ISDPC20 (“The project involved multiple external contractors”)—were combined into a single item because of their similarity.

### Pilot Test 2

After refining the items, we created an online survey questionnaire using the remaining 20 items. To reveal any potential problems or issues with Web-based online survey, a second pilot test was conducted with 15 ISDP managers who had an average of seven years of experience in ISDP management. The ISDP managers logged on to the Web survey and filled out the questionnaire based on their experience with the most recently completed ISDP they were involved in. After finishing the survey, each manager provided suggestions for improving the content and the format of the online survey. Overall, only minor editorial issues related to the format and wordings of the questionnaire were reported and resolved.

### Final Refinement of Measurement Items

A further examination on the measurement items resulted in dropping five items in this phase, as shown in Appendix Table A1. These five items tapped into either (1) other theoretical constructs such as user involvement and top management support or (2) project risk factors that did not seem closely related to complexity, such as lack of control over project resources, lack of staffing, and lack of knowledge/skill. Therefore, we concluded that inclusion of these items could make the scope and boundary of the ISDP complexity construct too broad and ambiguous. As a result, 15 items were retained for confirmatory factor analysis. Appendix Table A2 summarizes the final items. Below, we discuss the rationales and sources of the final 15 measures that are used to capture the four components of ISDP complexity.

Measures of the first complexity component, *structural organizational complexity*, were generally associated with the multiplicity or coordination of user groups, exter-

nal contractors and vendors, and project team's functional composition. Three items were used to measure SORG. These items assessed the complexity associated with managing cross-functional project teams [7, 8], working with multiple external contractors and vendors [2, 7, 34, 59, 83], and coordinating multiple user units [8, 34, 59].

Measures of the second component of ISDP complexity, *structural IT complexity*, captured the complexity associated with (1) the multiplicity of technological components and (2) integration with other systems. Four items were used to measure SIT. These items were used to capture the complexity related to the multiplicity of the software development environments and technological platform [49], real-time data processing [27], and system integration [7, 8, 48, 49].

Measures of the third component of ISDP complexity, *dynamic organizational complexity*, captured the complexity associated with changes in the organization. The interactions between IS development and organizational changes are bidirectional [54]. As such, two items were used to capture changes in the organizational structure and business processes that affect business requirements [47]. Two additional items were used to capture the business changes that are caused by the IS delivered by the ISDP [7, 8, 60]. Finally, one item was used to assess changes in users' information needs [12, 58, 59].

Measures of the last component of ISDP complexity, *dynamic IT complexity*, captured the complexity associated with changes in the technological environments and development tools. Three items were used to assess DIT. One item was used to measure complexity related to changes in IT architecture [48, 59]. IT architecture refers to the overall structure of the corporate IS and consists of the applications and databases for various levels of the organization [22, 57]. The second item measured changes in IT infrastructure [48, 60]. The third item captured changes in systems development tools [48].

### Phase 3—Survey Data Collections

The Web-based online survey resulted from the second phase of the research process was used to collect the large-scale data for validating the conceptual framework and the measures of ISDP complexity. The items were randomly ordered to minimize any bias from the survey method. Seven-point Likert scales were used for the items measuring ISDP complexity. The source of the survey respondents was the Information Systems Specific Interest Group of the Project Management Institute (PMI-ISSIG), which is an international organization of IS professionals with about 15,000 members. We used three criteria to select our target respondents: (1) North American PMI-ISSIG members who (2) were project managers (not specialists such as programmers or systems analysts), and (3) had managed a recently completed ISDP. The reason for choosing North American members was to avoid bias and problems that might be caused by language barriers that the non-English-speaking members in the other regions might have.

The PMI-ISSIG sponsored this research by providing their membership information and promoting the survey to their members. A PMI-ISSIG-sponsored e-mail

letter with a hyperlink to the Web-based survey was sent to the target group. To encourage participation, survey participants were entered into a drawing to receive 10 awards of a PMI-ISSIG official shirt and 40 awards of a \$25 gift certificate from a well-known online store. A reminder was sent two weeks after the initial PMI-ISSIG-sponsored e-mail was sent out. A second e-mail reminder was sent two weeks later.

The total number of potential respondents was 1,740. In total, 565 responses were received, representing a response rate of 32.5 percent. Twenty-four incomplete responses were dropped, resulting in a usable sample size of 541, and a final response rate of 31.1 percent. Table 3 illustrates the characteristics of the study sample. The sample represents various industry sectors, ranging through manufacturing, financial services, software, consulting, retailing, transportation, health care, and utility. On average, companies in the sample had annual sales of \$2.55 billion, with 14,800 employees. Three types of ISDPs—in-house new development, packaged software implementation, and enhancement of existing software—were evenly represented in the sample. On average, projects in the sample had a budget of \$2.1 million, team size of 34 members, and duration of 12 months.

## Phase 4—Data Analysis and Measurement Validation

### Data Screening and Descriptive Analysis

The survey data were carefully screened for unusual patterns, nonresponse bias, and outliers. A careful screening of the responses did not reveal any unusual patterns or careless responses, indicating that the questionnaire's design was appropriate and the respondents were serious and careful in completing the questionnaires. To examine nonresponse bias, we recorded the dates on which the responses were received. Comparisons of early responses and later responses on key demographic and complexity scores did not reveal any significant differences, indicating that response bias is unlikely to be a problem.

### Confirmatory Factor Analysis

Confirmatory factor analysis with LISREL 8.54 was used to compare five alternative measurement models of ISDP complexity, as shown in Figure 3. Based on the prior literature and our conceptualization of the ISDP complexity construct, we developed a measurement model in which the four first-order factors were loaded onto a second-order factor of ISDP complexity. The other four models were: (1) a null model in which all measures were uncorrelated to each other, (2) a model in which all measures were loaded onto a single first-order factor, (3) a model in which the measures were loaded onto four uncorrelated first-order factors, and (4) a model in which the measures were loaded onto four correlated first-order factors.

These five alternative models were compared using two groups of goodness-of-fit indices. The first group is absolute indices that are sensitive to sample size. They include the  $p$ -value of the chi-square statistic, the ratio of chi-square to degrees of

Table 3. Characteristics of the Study Sample ( $n = 541$ )

Characteristics of organizations	Percent	Characteristics of projects	Percent
Industry		Type of project	
Consulting	6.3	In-house new development	38.1
Finance/insurance	20.6	Packaged software implementation	33.9
Government	9.2	Enhancement of existing software	28.0
Health care	5.9		
Manufacturing	13.7	Number of project members	
Retail	5.3	Less than 10	25.0
Software	9.7	10–50	55.4
Telecom/network	5.0	Over 50	19.6
Transportation	4.0		
Utility	7.4	Project budget	
Other	12.9	Less than \$100,000	17.5
Company annual sales		\$100,000–\$1 million	41.8
Less than \$100 million	26.0	Over \$1 million	40.7
\$100 million–\$1 billion	31.2		
Over \$1 billion	42.8	Project duration	
Number of employees		Less than 6 months	24.8
Less than 1,000	26.6	6–12 months	40.9
1,000–10,000	40.5	Over 12 months	34.3
Over 10,000	32.9		

freedom, the goodness-of-fit index (GFI), the adjusted goodness-of-fit index (AGFI), the root mean square error of approximation (RMSEA), and the standardized root mean square residual (RMR). The second group is relative indices that are less sensitive to sample size. They include the comparative fit index (CFI) and the normed fit index (NFI) [28, 37]. A model is considered to have good model-data fit if the  $p$ -value of the chi-square is above 0.05, the ratio of chi-square to degrees of freedom is smaller than 5, the GFI is above 0.90, the AGFI is above 0.80, the RMSEA is less than 0.08, the standardized RMR is less than 0.10, the CFI is above 0.90, and the NFI is above 0.90 [15, 17, 32, 36, 79].

The existence of a second-order factor is justified by examining the target coefficient ( $T$ ) [46]. The  $t$ -coefficient can be calculated using the following formula:

$$T = \chi^2(\text{first-order model}) / \chi^2(\text{second-order model}).$$

Since the  $t$ -coefficient indicates the extent to which the second-order factor accounts for the variance among the first-order factors, it has an upper limit of 1.0. As such, the goodness-of-fit of the second-order model can never be better than the corresponding first-order model. A high  $t$ -coefficient implies that the relationship among first-order factors is sufficiently captured by the higher-order factor, thus indicating the validity of a second-order model.

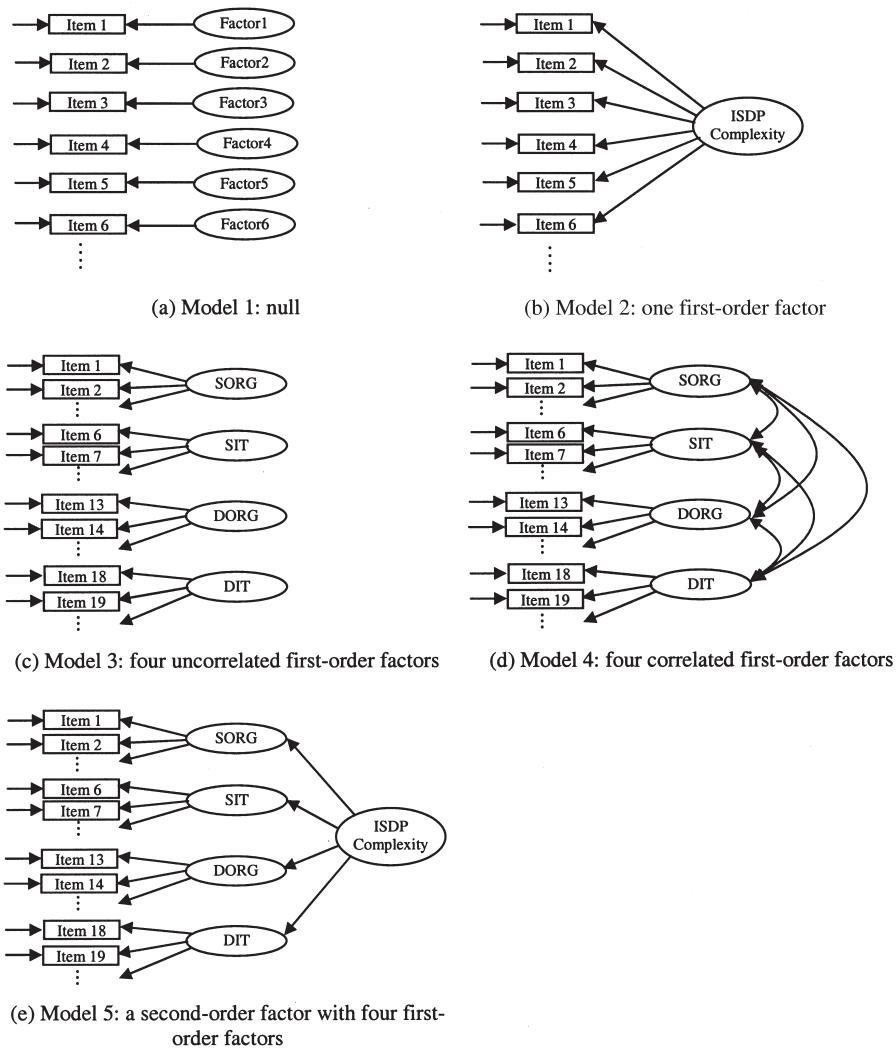


Figure 3. Alternative Models Tested in the Confirmatory Analysis

Based on the results of the model comparisons, the measurement model that best represented the ISDP complexity construct was chosen for further measurement validation. The unidimensionality and convergent validity of the four latent components were assessed by specifying a single-factor model for each latent variable. The reliability was assessed by the composite reliability index that was calculated based on factor loadings and variances [78]. One advantage of the composite reliability index is that it is free from the restricted assumption of equal importance of all indicators on which the Cronbach's alpha is based. Following Sethi and King [63], Venkatraman [75], and Werts et al. [78], the composite reliability was calculated from the factor loadings of each indicator and error variances using the following formula:

$$\rho_c = (\sum \lambda_i)^2 \text{ Variance}(A) / ((\sum \lambda_i)^2 \text{ Variance}(A) + \sum \theta_\delta).$$

The discriminant validity of the first-order factors was assessed using the techniques suggested by Sethi and King [63] and Venkatraman [75]. The discriminant validity is supported if the correlation between a pair of latent variables was significantly different from unity. A model with the correlation constrained to one was compared with an unconstrained model. A significantly lower chi-square value for the unconstrained model, when compared with that of the constrained model, indicates discriminant validity between the pair of latent variables.

#### Factorial Invariance Analysis

Factorial invariance analysis was conducted to establish the generalizability of the measures across three types of ISDPs. Tests of factorial invariance examine whether a measure operates equivalently across different subpopulations [15]. The value of a measurement model is greatly enhanced if the same factorial structure and properties can be replicated across various subpopulations [46]. ISDPs were defined in this study to include three types of system development: in-house new development, packaged software implementation, and major enhancement of existing software. As such, it is important to examine whether the factorial structure and properties of the measure are invariant across the three types of ISDPs.

After eliminating 30 data cases with missing data on project type, we segmented the remaining data of the overall sample by project type. The sample size was 195 for in-house new development projects, 173 for packaged software implementation projects, and 143 for major enhancement of existing software, respectively. In conducting the factorial invariance analysis, we first established an initial model and tested for model-data fit. Measurement models with invariant factor loadings were then specified and tested. If the difference in model fits between the initial model and the models with invariance constraints was not significant, invariance of the factorial structure of the measurement across the three ISDP types was supported.

#### Nomological Validity

Finally, the nomological (or predictive) validity of the ISDP complexity measure was examined. Nomological validity assesses whether a construct measured by the new measure is associated with other constructs whose measures are known to be valid, as the theory would predict. In this study, a positive association between ISDP complexity and project duration was predicted. In order to test this predicted relationship, a composite score was calculated for each of the four factors based on their corresponding items. In addition, an overall ISDP complexity score was obtained by averaging the four factor scores. Since projects in the sample were completed at the time of data collection, project duration data was available. A path analysis was used to test the nomological validity of the ISDP complexity measures based on whether or

not the relationship between ISDP complexity and project duration was positive as predicted.

## Results

---

### Confirmatory Factor Analysis

#### Model-Data Fits of Alternative Models

TABLE 4 SHOWS THE MODEL FIT RESULTS of alternative measurement models for ISDP complexity (see Figure 3 for model specifications). All models had significant chi-square statistics, indicating an unsatisfactory model fit. However, the significant chi-square statistics are likely due to large sample size [4]. Therefore, other indices that are independent of sample size provide more meaningful information for model comparison. Models 1, 2, and 3 were not acceptable because most of their fit indices did not meet the threshold criteria. Both Models 4 and 5 were acceptable because all fit indices met the threshold criteria. Figures 4 and 5 show the results of the parameter estimations of the first-order model (Model 4) and the second-order model (Model 5).

We assessed the validity of a second-order model (Model 5) through examination of the  $t$ -coefficient [46]. The  $t$ -coefficient between Model 4 and Model 5 is a very high 0.94, supporting that the second-order model explains a significant proportion of the covariation among first-order factors, thus indicating the efficacy of Model 5. Furthermore, the paths from the second-order factor of ISDP complexity to the first-order factors were all significant. Based on these results, we concluded that Model 5 best represented the ISDP complexity construct.

#### Unidimensionality and Convergent Validity

Unidimensionality and convergent validity require that one single latent variable underlies a set of measures [1]. To test unidimensionality and convergent validity, we generated four first-order models with each corresponding to one of the four components of ISDP complexity. The results shown in Table 5 suggest that all four latent variables demonstrated adequate levels of model fit. Overall, the results indicate that the measures of each of the four ISDP components satisfy the unidimensionality and convergent validity requirements.

#### Internal Consistency Reliability

The composite reliability ( $\rho_c$ ), which represents the proportion of measure variance attributable to the underlying latent variable, was calculated to assess the reliability of the measure [78]. Values of  $\rho_c$  in excess of 0.50 indicate that the variance captured by the measures is greater than that captured by error components, thus suggesting satisfactory levels of reliability [3]. The results in Table 6 show the composite reliability

Table 4. Model Fit Test Results of Alternative Models (*n* = 541)

Criteria	Threshold	Model 1 Null	Model 2 One first-order factor	Model 3 Four uncorrelated first-order factors	Model 4 Four correlated first-order factors	Model 5 A second-order factor with four first-order factors
$\chi^2$		4,494.00*	2,488.40*	566.92*	347.06*	370.46*
d.f.		105	90	90	84	86
$\chi^2/\text{d.f.}$		42.8	27.6	6.3	4.1	4.3
GFI	(< 5.0)	0.47	0.62	0.88	0.92	0.92
AGFI	(> 0.90)	0.40	0.49	0.84	0.89	0.88
RMSEA	(> 0.80)	0.278	0.222	0.097	0.076	0.078
RMR	(< 0.080)	0.26	0.15	0.16	0.062	0.074
CFI	(< 0.10)	0.30	0.58	0.89	0.94	0.93
NFI	(> 0.90)	0.29	0.57	0.87	0.92	0.92

\* *p* < 0.05.



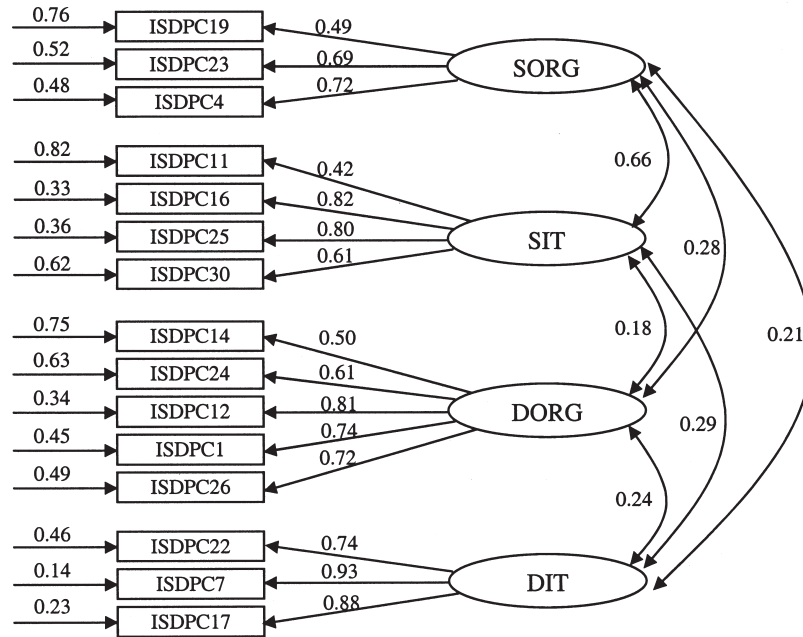


Figure 4. Parameter Estimates of the First-Order Model (Model 4,  $n = 541$ )

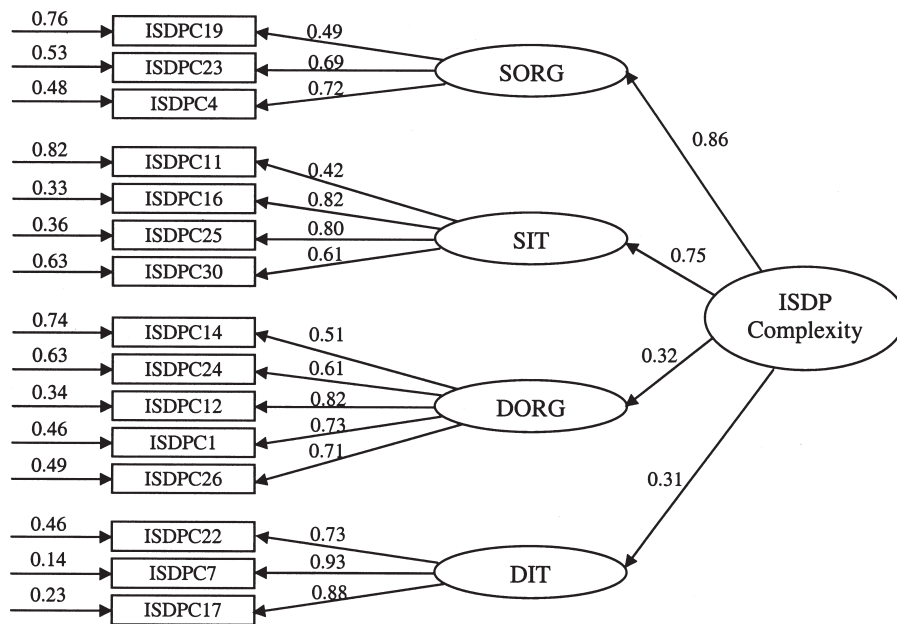


Figure 5. Parameter Estimates of the Second-Order Model (Model 5,  $n = 541$ )

Table 5. Unidimensionality/Convergent Validity ( $n = 541$ )

Factor	Number of indicators	$\chi^2$	d.f.	$\chi^2/\text{d.f.}$	GFI	AGFI	RMR	CFI
SORG <sup>a</sup>	3	19.22*	8	2.4	0.99	0.97	0.047	0.99
SIT	4	12.63*	2	6.3	0.99	0.94	0.034	0.98
DORG	5	86.05*	5	17.2	0.94	0.82	0.059	0.92
DIT <sup>a</sup>	3	19.22*	8	2.4	0.99	0.97	0.047	0.99

Notes: <sup>a</sup> This model is saturated because the number of indicators is 3. Therefore, the fit indices are not available. Fit indices for this factor were produced from a two-factor model including SORG and DIT. \*  $p < 0.05$ .

Table 6. Composite Reliability  $\rho_c$  ( $n = 541$ )

Factor	Number of indicators	$\rho_c$
SORG	3	0.68
SIT	4	0.76
DORG	5	0.81
DIT	3	0.89

$$\rho_c = (\sum \lambda_i)^2 \text{ Variance}(A) / ((\sum \lambda_i)^2 \text{ Variance}(A) + \sum \theta_{\delta}).$$

for the four latent variables. SIT ( $\rho_c = 0.76$ ), DORG ( $\rho_c = 0.81$ ), and DIT ( $\rho_c = 0.89$ ) demonstrated extensive evidence of reliability [9]. The composite reliability for SORG was 0.68, which was suboptimal, although acceptable [30, 53].

#### Discriminant Validity

Discriminant validity assesses the degree to which measures of different components of ISDP complexity are unique from one another. The results of the pairwise tests are shown in Table 7. The results suggest that all six pairs were statistically different, indicating that the four components of ISDP complexity demonstrated adequate levels of discriminant validity.

#### Analysis of Factorial Invariance

Table 8 shows the results of factorial invariance tests. Tests of factorial invariance take a hierarchical approach. First, an initial second-order model (Model A in Table 8) as specified as Model 5 in Figure 5 was constructed and tested for each of the three types of ISDPs. The initial model did not have any invariance constraints across the three types of ISDPs. The purpose of testing the initial second-order models across the project types was to ensure that the measurement model fit the data well for each of the project types.

Table 7. Discriminant Validity of Four First-Order Factors ( $n = 541$ )

Construct pair	Maximum-likelihood estimate $\phi$	$t$ -value	$\chi^2$ statistic		
			Constrained model (d.f.)	Unconstrained model (d.f.)	Difference
SORG-SIT	0.75	5.99**	125.55 (14)	113.59 (13)	11.97**
SORG-DORG	0.26	3.41**	186.93 (20)	133.80 (19)	53.13**
SORG-DIT	0.15	2.44*	66.78 (9)	19.22 (8)	47.56**
SIT-DORG	0.18	3.10**	236.36 (27)	143.81 (26)	92.55**
SIT-DIT	0.24	4.09**	98.94 (14)	39.51 (13)	59.43**
DORG-DIT	0.29	5.02**	161.36 (20)	115.41 (19)	45.95**

\*  $p < 0.05$ , \*\*  $p < 0.01$ .

Table 8. Invariance of the Second-Order Model Across Three Types of ISDP  
( $n = 511$ )

Model description	$\chi^2$ (d.f.)	$\chi^2/\text{d.f.}$	$\chi^2$ difference with Model A	$t$
Model A: Second-order model with no invariance constraints across three types of ISDPs	470.83 (258)	1.82	—	1.00
Model B: Model A with invariant items loadings to the first-order factors	476.83 (264)	1.81	6.00 (6), n.s.	0.99
Model C: Model B with invariant structural coefficients between first- and second-order factors	490.63 (272)	1.80	19.80 (14), n.s.	0.96

Once the initial model was established, invariance of first-order factor loadings was tested (Model B). In this step, the second-order factor loadings were not constrained to be invariant. The rationale behind this approach was that tests of higher-order invariance would make sense only when there was reasonable invariance among the first-order factors [46]. Model B was compared with Model A to examine if the first-order factor loadings were invariant. Difference in the chi-square between the two models was tested. If the first-order factor loadings turned out to be invariant, a more restricted model with invariant first- and second-order factor loadings (Model C) was tested. Again, this model was compared with the initial model to examine if the second-order factor loadings were invariant.

The results shown in Table 8 suggest that the first-order factor loadings were invariant because the chi-square difference between Model A and Model B were not significant. In addition, Model B showed an acceptable ratio of chi-square to degrees of freedom, and its target coefficient against Model A was very high. Similarly, the second-order factor loadings were invariant because the chi-square difference between Model A and Model C was not significant. Model C showed an acceptable ratio of chi-square to degrees of freedom, and its target coefficient against Model A was very high. Therefore, we concluded that the factorial structure of the second-order measurement model of ISDP complexity was invariant across the three types of ISDPs. The results provide initial empirical evidence of the generalizability of the measurement model across the three types of ISDPs.

### Nomological Validity

We analyzed the nomological validity of the ISDP complexity measures by testing a hypothesized positive relationship between ISDP complexity and project duration. Our proposed positive relationship between ISDP complexity and project duration is supported by the argument that project complexity imposes more workload and thus

Table 9. Nomological Validity—Regression Analysis Results ( $n = 467$ )

Relationship	Adjusted $R^2$	Beta
Model I	0.127	
Project duration = Overall ISDP complexity		0.359**
Model II	0.132	
Project duration = Structural organizational complexity + Structural IT complexity + Dynamic organizational complexity + Dynamic IT complexity		0.127* 0.126* 0.137** 0.155**

\*  $p < 0.05$ , \*\*  $p < 0.01$ .

causes longer project duration [19, 29, 50]. For example, Meyer and Utterback [50] found that technological complexity as measured by the number of technologies in the development effort was positively associated with the development time.

Table 9 shows the results of the path analyses of the effects of (1) overall ISDP complexity (second-order factor) and (2) the four components of ISDP complexity (first-order factors) on project duration, respectively. The results indicate that all four factors of ISDP complexity as well as the overall ISDP complexity positively affected project duration. Therefore, the prediction was supported by the data. We concluded that the measurement of ISDP complexity demonstrated adequate nomological validity in predicting project duration.

## Discussions and Conclusions

THE RESULTS OF CONFIRMATORY DATA ANALYSES suggested that the 15-item measurement of ISDP complexity developed in this research exhibited adequate levels of measurement properties. The hypothesized measurement model had adequate levels of goodness of fit. They also supported the validity of a second-order factor, which can be interpreted as the overall ISDP complexity. The measures were shown to satisfy criteria related to unidimensionality, convergent validity, discriminant validity, internal consistency reliability, factorial invariance across three types of projects, and nomological validity.

## Contributions to Research

The central contribution of this research is the unpacking of the complexity construct in the ISDP context. We believe that ISDP complexity will be an important construct in the vocabulary of IS researchers and practitioners for the following two reasons. First, an increasing portion of IS activities in organizations are organized around projects. Therefore, projects constitute an important context and a unit of analysis for research. Second, the constantly changing IT and business environments coupled with

the growing needs for IT application integrations will cause the level of ISDP complexity to continue to increase. Given the historically high project failure rate, managing ISDP complexity is critical to IS success. Given this increasing significance of ISDP complexity, it is timely and important to develop a conceptual framework and a valid, reliable measure of the construct.

Although there are other related constructs such as software complexity, general project complexity, and task complexity, they are not substitutes for ISDP complexity. Task complexity and project complexity are too general to tap into the unique context of ISDPs. Software complexity is too limited and narrow to assess the various aspects of ISDP complexity. The new measure developed in this research overcomes these limitations and covers a wide range of the domain of the ISDP complexity construct with enhanced specificity. There is a critical need for rigorous theories that managers can use as guidelines to understand and manage ISDP complexity. A significant gap in the research literature is the lack of clear definitions and valid measures of ISDP complexity. Without commonly accepted definitions and measurements, it is difficult to develop and test theories, compare results across empirical studies, and accumulate theories that managers can use. This paper serves as a starting point to address these deficiencies.

In addition, by defining four distinct components of ISDP complexity, this research enables researchers to more precisely theorize the construct. As Baccarini [2] argued, since complexity is multidimensional, when referring to project complexity, it is important to state clearly the type of complexity being dealt with. The conceptual framework and the measure of ISDP complexity developed in this research will enable researchers to use these measures to build and test theories that explain the determinants and effects of ISDP complexity. Depending on their study purposes, researchers can select either the second-order factor or the first-order factors of ISDP complexity as the focal constructs to develop theories.

The factorial invariance test results suggest that the measurement scales can be generalized or applied across different types of ISDPs. Validation of measurement applicability across different segments of the population has significant research implications. Without establishing measurement applicability, if the measurement results of ISDP complexity are significantly different across the various segments, it is difficult to conclude whether ISDPs in different segments have different levels of complexity, or the differences are simply measurement biases because different measurements are needed for different segments. Therefore, the current research assures that the ISDP complexity measures developed here are valid for different types of ISDPs and can be used to compare complexity measurement results across project types.

## Practical Implications

The results of our study also have important practical implications. Although the importance of assessing and managing complexity of ISDPs has been widely recognized, organizations are not well equipped to cope with these challenges. As Kotter [41] suggests, managing structural and dynamic complexities has become a key responsi-

bility of managers and executives. As such, this research provides a much needed language and measurement tool that managers can use to describe and communicate ISDP complexity. First, the empirically validated four-component framework of ISDP complexity serves as a useful language for defining and communicating ISDP complexity. By using this framework, project managers can clearly define the specific aspects and components of ISDP complexity that they must consider and manage. Second, the measures developed in this study can be used to assess and manage the complexity of ISDPs in the early planning stages and during implementation. Without such an assessment tool, it would be difficult for project managers to identify areas of concerns and take appropriate measures. It has been found that complexity influences the selection of project inputs including project organizational form, budget, manpower, expertise, and experience requirements of the project team [2, 38]. Therefore, being able to accurately assess ISDP complexity enables organizations to better plan, coordinate, and control their projects.

A valid and reliable measurement tool would allow organizations to learn from past experiences and establish a knowledge base of organizational techniques that have been proven to be effective in dealing with different aspects of ISDP complexity. If ISDPs are tracked over a long period of time in an organization, the organization can relate different levels of ISDP complexity to project profiles (e.g., project resources, staff, governance structure, process, development methodology, etc.), organizational interventions to mitigate ISDP complexity, and project performance. Used together, the assessment tool and the knowledge base enable organizations to estimate project resources and duration required by an ISDP with a certain level of complexity, to develop critical capabilities needed for planning and controlling their ISDPs, and to identify most effective organizational interventions to mitigate the negative effect of ISDP complexity on project performance.

The second-order factor and the four first-order factors can serve different purposes in the practice of estimating and managing ISDP complexity. The second-order factor is useful for communicating the overall level of ISDP complexity with users and business unit managers. It is also useful for overall project planning in the early stages of project life cycle. In contrast, the four first-factors (or the four components) of ISDP complexity can be used to facilitate detailed assessments and communications within the project team. They are useful for identifying specific complexity areas, thus enabling the managers to strategically manage and control the most important aspects or components of ISDP complexity during project implementations.

### Limitations of the Study

Due to the limitations of our study, cautions should be taken when interpreting the study findings and applying the measures developed in this research. Although this research took a systematic, multiple-stage approach to measurement development, the final measurement items may have some weaknesses. Although they reflected the most critical aspects of ISDP complexity from the practitioners' point of view, they may not cover the whole range of the theoretical domain of the construct. For ex-

ample, SORG items included multiplicity and interdependency among project team members, user units, and external contractors/vendors. However, this list may not be exhaustive, because there are other potentially important organizational components such as business process and organizational structure.

Another weakness is that DORG and DIT items are not exact mirrors to SORG and SIT, respectively. The final items show that DORG and DIT items tended to tap into macro-level project environments (e.g., organizational structure, business process, IT infrastructure, IT architecture), whereas SORG and SIT items tapped into relatively micro-level project environments (e.g., project team composition, data processing mode, software environment, etc.). Our speculation is that ISDP managers perceived changes in macro-level environments as more salient and critical for managing their projects than changes in micro-level environments.

The composite reliability for SORG ( $\rho_c = 0.68$ ), was relatively suboptimal, although acceptable. One plausible explanation for this suboptimal reliability is that the factor loading of one item related to the multiplicity of external vendors was relatively low, indicating its lack of internal consistency with the other two items.

In testing the nomological validity of the measurement, the same respondent provided information about both the independent and the dependent variables. Although the dependent variable (project duration) is relatively objective data, the same respondent might have caused common-source biases. By using different sources for information about the dependent and the independent variables, future research can further confirm the nomological validity of the ISDP complexity measurement.

We used ISDP managers as the primary source of data collection. However, we also used general IS managers and IS researchers for a focus group discussion and a sorting procedure, respectively. The use of different groups of subjects in different phases might create concerns about potential biases. However, we do not think that the use of different subjects would undermine the validity of the final items significantly, because different groups of subjects in different phases appropriately served our purpose of developing the ISDP complexity measure. A broader range of IS managers in the focus group discussion helped us in creating a comprehensive pool of initial items. IS researchers were used in a sorting procedure because the sorting procedure required some degree of theoretical understanding of the construct. In addition, all subjects had extensive work experience in the IS field.

### Directions for Future Research

This research represents a first step toward building theories that provide insights about the conceptualization and measurement of ISDP complexity. The framework and measures developed in this study can help organizations improve understanding of ISDP complexity and can provide the initial tools for assessing and managing the complexity of their ISDPs. Since sound measurements can only be established through a series of replications and validations, future research should further validate and refine the measurement scales with comparative data. In particular, future research needs to further refine the SORG measure in order to increase the internal consis-



tency reliability of the construct. In particular, as shown in Figure 4, the loading of ISDPC19 ("The project involved multiple external vendors") to SORG was lower than that of the other two SORG items. Therefore, the relationship between the complexity of external contractors and vendors and the complexity of internal organizational elements such as users and project teams needs to be further investigated. In addition, in order to ensure the appropriateness of applying the measurement scales to different types and contexts of ISDPs, future research is needed to examine the generalizability of the ISDP complexity measurement scales across different contexts of projects. In this study, we used the factorial invariance test to validate the generalizability of the measurement scales across three different types of ISDPs. Similar factorial invariance methods can be used to test the generalizability of the measurement scales across other project segmentations, such as different project scales/duration and different systems development methodologies.

Using the measure developed in this research, future research should build and test theories about ISDP complexity. Some of the important questions that need to be answered include: (1) What are the organizational and technological antecedents to ISDP complexity? (2) What are the effects of the four ISDP complexity components on project performance and organizational effectiveness? (3) Under what conditions does ISDP complexity become more or less critical to project success? (4) How can organizations mitigate the negative effect of ISDP complexity? Future research answering these questions would help organizations minimize unnecessary complexity and effectively manage necessary complexity to enhance the success rate of their ISDPs by creating effective strategies, methods, and coping mechanisms to control and manage those organizational factors that influence ISDP complexity.

Future research should take ISDP life cycles into consideration. The degree of ISDP complexity may differ across the various phases of an ISDP's life cycle. The effect of ISDP complexity on project performance may vary across the different phases in the system development life cycle. In addition, organizations and ISDP teams may require different coping strategies for ISDP complexity in different phases of life cycles. Longitudinal studies can be useful for enhancing our understanding of the dynamics of ISDP complexity and provide insights that cross-sectional studies may not be able to produce.

Finally, it is important to examine project portfolio complexity. Organizations are most likely to run more than one ISDP at a time. Optimizing one project may create global suboptimization that hinders the performance of the overall project portfolio. Therefore, understanding complexity at the portfolio level enables IT organizations to achieve efficiency and effectiveness not only for individual projects but also with a portfolio of projects. Our hope is that this research serves as a starting point for stimulating researchers to develop theories for understanding and managing ISDP complexity, and, ultimately, stopping the dollar drain of ISDP failures.

---

*Acknowledgments:* This study was supported by research grants provided by the University of Minnesota, the Juran Center for Leadership in Quality, and the UPS (United Parcel Service)

Foundation. The Information Systems Special Interest Group of the Project Management Institute sponsored the collection of the survey data. The authors thank Carl Adams, Shawn Curley, Gordon Davis, Paul Johnson, Rob Kauffman, and research workshop participants at American University, Lehigh University, Syracuse University, Tulane University, the University of British Columbia, and the University of Minnesota for helpful comments on earlier versions of the paper. They also thank Vladimir Zwass, the Editor-in-Chief of *JMIS*, and three anonymous reviewers for their comments and suggestions that helped a great deal in improving the paper.

## REFERENCES

1. Anderson, J.C., and Gerbing, D.W. Structural equation modeling in practice: A review and recommended two-step approach. *Psychological Bulletin*, 103, 3 (1988), 411–423.
2. Baccarini, D. The concept of project complexity—A review. *International Journal of Project Management*, 14, 4 (1996), 201–204.
3. Bagozzi, R.P. An Examination of the validity of two models of attitude. *Multivariate Behavioral Research*, 16, 3 (1981), 323–359.
4. Bagozzi, R.P.; Yi, Y.; and Phillips, L. Assessing construct validity in organizational research. *Administrative Science Quarterly*, 36, 3 (1991), 421–458.
5. Baker, S. Where danger lurks: Spam, complexity and piracy could hinder tech's recovery. *Business Week* (August 25, 2003), 114–118.
6. Balachandra, R., and Friar, J.H. Factors for success in R&D project and new product innovation: A contextual framework. *IEEE Transactions on Engineering Management*, 44, 3 (1997), 276–287.
7. Barki, H.; Rivard, S.; and Talbot, J. Toward an assessment of software development risk. *Journal of Management Information Systems*, 10, 2 (Fall 1993), 203–225.
8. Barki, H.; Rivard, S.; and Talbot, J. An integrative contingency model of software project risk management. *Journal of Management Information Systems*, 17, 4 (Spring 2001), 37–69.
9. Bearden, W.O., and Netemeyer, R.G. *Handbook of Marking Scales: Multi-Item Measures for Marketing and Consumer Behavior Research*. Newbury Park, CA: Sage, 1999.
10. Benamati, J., and Lederer, A.L. Coping with rapid changes in IT. *Communications of the ACM*, 44, 8 (2001), 83–88.
11. Bhattacharjee, A., and Hirschheim, R. IT and organizational change: Lessons from client/server technology implementation. *Journal of General Management*, 23, 2 (1997), 31–46.
12. Boehm, B.W. Software risk management: Principles and practices. *IEEE Software*, 8, 1 (1991), 32–41.
13. Brooks, F.P., Jr. *The Mythical Man-Month*. Reading, MA: Addison-Wesley, 1995.
14. Byrd, T.A., and Turner, D.E. Measuring the flexibility of information technology infrastructure: Exploratory analysis of a construct. *Journal of Management Information Systems*, 17, 1 (Summer 2000), 167–208.
15. Byrne, B.M. *Structural Equation Modeling with LISREL, PRELIS, and SIMPLIS: Basic Concepts, Applications, and Programming*. Mahwah, NJ: Lawrence Erlbaum, 1998.
16. Campbell, D.J. Task complexity: A review and analysis. *Academy of Management Review*, 13, 1 (1988), 40–52.
17. Chin, W.W., and Todd, P.A. On the use, usefulness, and ease of use of structural equation modeling in MIS research: A note of caution. *MIS Quarterly*, 19, 2 (1995), 237–246.
18. Churchill, G.A., Jr. A paradigm for developing better measures of marketing constructs. *Journal of Marketing Research*, 16, 1 (1979), 64–73.
19. Clark, K.B. Project scope and project performance: The effect of parts strategy and supplier involvement on product development. *Management Science*, 35, 10 (1989), 1247–1263.
20. Cleland, D.I., and King, W.R. *Systems Analysis and Project Management*. New York: McGraw-Hill, 1983.
21. Davis, G.B. Strategies for information requirements determination. *IBM Systems Journal*, 21, 1 (1982), 4–30.
22. Dickson, G.W., and Wetherbe, J.C. *The Management of Information Systems*. New York: McGraw-Hill, 1985.

23. Duncan, N.B. Capturing flexibility of information technology infrastructure: A study of resource characteristics and their measure. *Journal of Management Information Systems*, 12, 2 (Fall 1995), 37–57.
24. Eisenhardt, K.M., and Tabrizi, B.N. Accelerating adaptive processes: Product innovation in the global computer industry. *Administrative Science Quarterly*, 40, 1 (1995), 84–110.
25. Ewusi-Mensah, K. Critical issues in abandoned information systems development projects. *Communications of the ACM*, 40, 9 (1997), 74–80.
26. Field, T. When bad things happen to good projects. *CIO Magazine* (October 15, 1997), 55–62.
27. Garmus, D., and Herron, D. *Function Point Analysis: Measurement Practices for Successful Software Projects*. Boston: Addison-Wesley, 2001.
28. Gefen, D. It is not enough to be responsive: The role of cooperative intentions in MRP II adoption. *DATA BASE for Advances in Information Systems*, 31, 2 (2000), 65–79.
29. Griffin, A. The effect of project and process characteristics on product development cycle time. *Journal of Marketing Research*, 34, 1 (1997), 24–35.
30. Hair, J.F.; Tatham, R.L.; Anderson, R.E.; and Black, W.C. *Multivariate Data Analysis with Readings*. Englewood Cliffs, NJ: Prentice Hall, 1998.
31. Hopper, M.D. Complexity: The weed that could choke IS. *Computerworld*, 30, 28 (July 8, 1996), 37.
32. Hu, L.T., and Bentler, P.M. Evaluating model fit. In R.H. Hoyle (ed.), *Structural Equation Modeling: Concepts, Issues, and Applications*. Thousand Oaks, CA: Sage, 1995, pp. 76–99.
33. Jesitus, J. Broken promises? Foxmeyer's project was a disaster: Was the company too aggressive or was it misled? *Industry Week* (November 3, 1997), 31–37.
34. Jiang, J.J., and Klein, G. Information system success as impacted by risks and development strategies. *IEEE Transactions on Engineering Management*, 48, 1 (2001), 46–55.
35. Johnson, J. Chaos: The dollar drain of IT project failures. *Application Development Trends*, 2, 1 (1995), 41–47.
36. Jöreskog, K.G., and Sörbom, D. *LISREL7: A Guide to the Program and Applications*. Chicago: SPSS, 1989.
37. Jöreskog, K.G., and Sörbom, D. *LISREL VIII User's Guide*. Mooresville, IN: Scientific Software, 1993.
38. Kearney, J.K.; Sedlmeyer, R.L.; Thompson, W.B.; Gray, M.A.; and Adler, M.A. Software complexity measurement. *Communications of the ACM*, 29, 11 (1986), 1044–1050.
39. 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.
40. Kirsch, L.J. The management of complex tasks in organizations: Controlling the systems development process. *Organization Science*, 7, 1 (1996), 1–21.
41. Kotter, J.P. What leaders really do. *Harvard Business Review*, 68, 3 (May–June 1990), 103–111.
42. Kraut, R.E., and Streeter, L.A. Coordination in software development. *Communications of the ACM*, 38, 3 (1995), 69–81.
43. Leveson, N.G. Software engineering: Stretching the limits of complexity. *Communications of the ACM*, 40, 2 (1997), 129–131.
44. Lyytinen, K., and Hirschheim, R. Information systems failures—A survey and classification of the empirical literature. *Oxford Surveys in Information Technology*, 4, 1 (1987), 257–309.
45. Lyytinen, 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.
46. Marsh, H.W., and Hocevar, D. Application of confirmatory factor analysis to the study of self-concept: First and higher order factor models and their invariance across groups. *Psychological Bulletin*, 97, 3 (1985), 562–582.
47. McFarlan, F.W. Portfolio approach to information systems. *Harvard Business Review*, 59, 5 (September–October 1981), 142–150.
48. 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, 4 (1994), 427–451.

49. Meyer, M.H., and Curley, K.F. An applied framework for classifying the complexity of knowledge-based systems. *MIS Quarterly*, 15, 4 (1991), 455–472.
50. Meyer, M.H., and Utterback, J.M. Product development cycle time and commercial success. *IEEE Transactions on Engineering Management*, 42, 4 (November 1995), 297–304.
51. Moore, G.C., and Benbasat, I. Development of an instrument to measure the perceptions of adopting an information technology innovation. *Information Systems Research*, 2, 3 (1991), 192–222.
52. Murray, J.P. Reducing IT project complexity. *Information Strategy: The Executive's Journal*, 16, 3 (2000), 30–38.
53. Nunnally, J.C. *Psychometric Theory*. New York: McGraw-Hill, 1967.
54. Orlikowski, W.J., and Robey, D. Information technology and the structuring of organizations. *Information Systems Research*, 2, 2 (1991), 143–169.
55. Pich, M.T.; Loch, C.H.; and De Meyer, A. On uncertainty, ambiguity, and complexity in project management. *Management Science*, 48, 8 (2002), 1008–1023.
56. Ribbers, P.M., and Schoo, K.-C. Program management and complexity of ERP implementation. *Engineering Management Journal*, 14, 2 (2002), 45–52.
57. Richardson, G.L., and Jackson, B.M. A principles-based enterprise architecture: Lessons from Texaco and Star Enterprise. *MIS Quarterly*, 14, 4 (1990), 385–403.
58. Ropponen, J., and Lyytinen, K. Components of software development risk: How to address them? A project manager survey. *IEEE Transactions on Software Engineering*, 26, 2 (2000), 98–112.
59. Schmidt, R.; Lyytinen, K.; Keil, M.; and Cule, P. Identifying software project risks: An international Delphi study. *Journal of Management Information Systems*, 17, 4 (Spring 2001), 5–36.
60. Scott, J.E., and Vessey, I. Managing risks in enterprise systems implementations. *Communications of the ACM*, 45, 4 (2002), 74–81.
61. Scott, K. Battle complexity to add profitability. *InformationWeek* (September 14, 1998), 18ER.
62. Senge, P.M. *The Fifth Discipline*. New York: Doubleday, 1990.
63. Sethi, V., and King, W.R. Development of measures to assess the extent to which an information technology application provides competitive advantage. *Management Science*, 40, 12 (1994), 1601–1627.
64. Shenhar, A., and Dvir, J.D. Toward a typological theory of project management. *Research Policy*, 25, 4 (1996), 607–632.
65. Shenhar, A.J. One size does not fit all projects: Exploring classical contingency domains. *Management Science*, 47, 3 (2001), 394–414.
66. Souder, W.E., and Song, X.M. Contingency product design and marketing strategies influencing new product success and failure in U.S. and Japanese electronics firms. *Journal of Product Innovation Management*, 14, 1 (1997), 21–34.
67. Standish Group. The chaos report. West Yarmouth, MA, 1994.
68. Standish Group. The chaos report. West Yarmouth, MA, 2001.
69. Swanson, E.B., and Beath, C.M. Reconstructing the systems development organization. *MIS Quarterly*, 13, 3 (1989), 293–307.
70. Tait, P., and Vessey, I. The effect of user involvement on system success: A contingency approach. *MIS Quarterly*, 12, 1 (1988), 91–108.
71. Tatikonda, M.V., and Rosenthal, S.R. Successful execution of product development projects: Balancing firmness and flexibility in the innovation process. *Journal of Operations Management*, 18, 4 (2000), 401–425.
72. Tatikonda, M.V., and Rosenthal, S.R. Technology novelty, project complexity, and product development project execution success: A deeper look at task uncertainty in product innovation. *IEEE Transactions on Engineering Management*, 47, 1 (2000), 74–87.
73. Truex, D.P.; Baskerville, R.; and Klein, H. Growing systems in emergent organizations. *Communications of the ACM*, 42, 8 (1999), 117–123.
74. Turner, J.R., and Cochrane, R.A. Goals-and-methods matrix: Coping with projects with ill defined goals and/or methods of achieving them. *International Journal of Project Management*, 11, 2 (1993), 93–102.
75. Venkatraman, N. Strategic orientation of business enterprises: The construct, dimensionality, and measurement. *Management Science*, 35, 8 (1989), 942–962.

76. Wallace, L.; Keil, M.; and Rai, A. How software project risk affects project performance: An investigation of the dimensions of risk and exploratory model. *Decision Sciences*, 35, 2 (2004), 289–321.

77. Weill, P., and Broadbent, M. *Leveraging the New Infrastructure: How Market Leaders Capitalize on Information Technology*. Boston: Harvard Business School Press, 1998.

78. Werts, C.E.; Linn, R.L.; and Joreskog, K.G. Intraclass reliability estimates: Testing structural assumptions. *Educational and Psychological Measurement*, 34, 1 (1974), 25–33.

79. Wheaton, B.; Muthen, B.; Alwin, D.; and Summers, G. Assessing reliability and stability in panel models. In D. Heise (ed.), *Sociological Methodology*. San Francisco: Jossey-Bass, 1977, pp. 84–136.

80. Williams, T.M. The need for new paradigms for complex projects. *International Journal of Project Management*, 17, 5 (1999), 269–273.

81. Winklhofer, H. Information systems project management during organizational changes. *Engineering Management Journal*, 14, 2 (2002), 33–37.

82. Wood, R.E. Task complexity: Definition of the construct. *Organizational Behavior and Human Decision Processes*, 37, 1 (1986), 60–82.

83. Wozniak, T.M. Significance vs. capability: “Fit for use” project controls. *Transactions of AACE International*, 37, 1 (1993), 1–8.

## Appendix A. Measurement Items

Table A1. Initial Items, References, and Refinement Results

Item	Item description	Result
ISDPC1	The end-users' organizational structure changed rapidly [47].	Final item
ISDPC2	The system involved data processing from multiple sites [27, 56].	Deleted (2.2)
ISDPC3	The project involved new technologies [7, 8, 39, 45, 59, 60, 71].	Deleted (2.2)
ISDPC4	The project team was cross-functional [7, 8].	Final item
ISDPC5	The system involved high transaction rate [27].	Deleted (2.2)
ISDPC6	The project manager did not have direct control over project resources [48].	Deleted (2.4)
ISDPC7	IT architecture that the project depended on changed rapidly [48, 59].	Final item
ISDPC8	The project was given a fixed deadline.	Deleted (2.1)
ISDPC9	Business users provided insufficient support and involvement [34, 39, 59].	Deleted (2.4)
ISDPC10	The project was a major capital investment by company [7, 8, 83].	Deleted (2.2)
ISDPC11	The system involved real-time data processing [27].	Final item
ISDPC12	The end-users' business processes changed rapidly.	Final item
ISDPC13	It was a new project that we had never done before [71, 83].	Deleted (2.1)
ISDPC14	Implementing the project caused changes in the users' business processes [8, 60].	Final item
ISDPC15	There were conflicts between user units involved in the project [7].	Deleted (2.2)
ISDPC16	The project involved multiple software environments [49].	Final item
ISDPC17	IT infrastructure that the project depended on changed rapidly [48, 60].	Final item
ISDPC18	The project required many person-hours [7, 8, 42, 45].	Deleted (2.1)
ISDPC19	The project involved multiple external vendors [2, 7, 34, 59, 83].	Final item
ISDPC20	The project involved multiple external contractors [2, 7, 34, 59, 83].	Combined into ISDPC19 (2.2)
ISDPC21	There was no sufficient commitment/support from the top management [7, 34, 39, 45, 59].	Deleted (2.4)
ISDPC22	Software development tools that the project depended on changed rapidly [48].	Final item
ISDPC23	The project involved coordinating multiple user units [8, 34, 59].	Final item
ISDPC24	Implementing the project caused changes in the users' organizational structure [7, 8, 60].	Final item
ISDPC25	The project involved multiple technology platforms [49].	Final item
ISDPC26	The end-users' information needs changed rapidly [12, 58, 59].	Final item

(continues)



Table A1. Continued

Item	Item description (reference)	Result
ISDPC27	There was no sufficient/appropriate staffing for the project [12, 39, 45, 56, 58, 59].	Deleted (2.4)
ISDPC28	There would have been a significant effect on the business if this project had failed [8, 83].	Deleted (2.1)
ISDPC29	The project personnel did not have required knowledge/skills [7, 8, 34, 39, 45, 58, 59, 60].	Deleted (2.4)
ISDPC30	The project involved a lot of integration with other systems [7, 8, 48, 49].	Final item

*Notes:* The items without references were obtained solely from field interviews and focus group discussions with project managers. The numbers in the Result column indicate the phase number in Figure 2 in which the item was deleted.

Table A2. Final Items By Group

ISDP component	Item	Item description
SORG	ISDPC4	The project team was cross-functional.
	ISDPC19	The project involved multiple external contractors and vendors.
SIT	ISDPC23	The project involved coordinating multiple user units.
	ISDPC11	The system involved real-time data processing.
	ISDPC16	The project involved multiple software environments.
	ISDPC25	The project involved multiple technology platforms.
DORG	ISDPC30	The project involved a lot of integration with other systems.
	ISDPC1	The end-users' organizational structure changed rapidly.
	ISDPC12	The end-users' business processes changed rapidly.
	ISDPC14	Implementing the project caused changes in the users' business processes.
	ISDPC24	Implementing the project caused changes in the users' organizational structure.
DIT	ISDPC26	The end-users' information needs changed rapidly.
	ISDPC7	IT architecture that the project depended on changed rapidly.
	ISDPC17	IT infrastructure that the project depended on changed rapidly.
	ISDPC22	Software development tools that the project depended on changed rapidly.

Appendix B. Covariance Matrix of the Study Sample

Table B1. Covariance Matrix ( $n = 541$ )															
	ISDPC1	ISDPC4	ISDPC7	ISDPC11	ISDPC12	ISDPC14	ISDPC16	ISDPC17	ISDPC19	ISDPC22	ISDPC23	ISDPC24	ISDPC25	ISDPC26	ISDPC30
ISDPC1	2.832														
ISDPC4	0.261	2.643													
ISDPC7	0.737	0.328	3.219												
ISDPC11	0.060	1.037	0.444	4.184											
ISDPC12	1.921	0.422	0.570	0.349	3.567										
ISDPC14	0.743	0.515	0.052	0.594	1.442	2.969									
ISDPC16	0.154	0.997	0.372	1.288	0.222	0.560	3.728								
ISDPC17	0.705	0.331	2.635	0.344	0.658	0.109	0.430	3.259							
ISDPC19	0.204	1.367	0.796	1.063	0.228	0.481	1.472	0.768	5.293						
ISDPC22	0.658	0.317	2.009	0.473	0.639	0.114	0.414	1.892	0.692	2.690					
ISDPC23	0.178	1.529	0.316	1.187	0.603	0.561	1.240	0.450	0.994	0.266	3.117				
ISDPC24	1.942	0.789	0.695	0.422	1.651	1.577	0.522	0.616	0.966	0.571	0.597	4.471			
ISDPC25	0.461	0.943	0.664	1.108	0.369	0.587	2.605	0.616	1.492	0.580	1.032	0.627	3.800		
ISDPC26	1.543	0.398	0.582	0.304	2.096	1.011	0.136	0.671	0.252	0.688	0.558	1.450	0.290	3.212	
ISDPC30	0.343	0.967	0.775	1.192	0.350	0.381	1.636	0.767	1.617	0.594	1.335	0.597	1.681	0.458	3.377