KNOWLEDGE INTEGRATION IN SOFTWARE TEAMS: AN ANALYSIS OF TEAM, PROJECT, AND IT-RELATED ISSUES

Except where reference is made to the work of others, the work described in this dissertation is my own or was done in collaboration with my advisory committee. This dissertation does not include proprietary or classified information.

Nikhil Mehta

Certificate of Approval:

Dianne Hall Assistant Professor Management	Terry Anthony Byrd, Chair Professor Management
Casey G. Cegielski Associate Professor Management	Stephen L. McFarland Dean Graduate School

KNOWLEDGE INTEGRATION IN SOFTWARE TEAMS: AN ANALYSIS OF TEAM, PROJECT, AND IT-RELATED ISSUES

Nikhil Mehta

A Dissertation

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Doctor of Philosophy

Auburn University
December 15, 2006

KNOWLEDGE INTEGRATION IN SOFTWARE TEAMS: AN ANALYSIS OF TEAM, PROJECT, AND IT-RELATED ISSUES

Nikhil Mehta

Signature of Author

Date of Graduation

VITA

Nikhil Mehta, a native of India, was born on February 15, 1974, the first child of Ved Prakash Mehta and Lalita Mehta. After graduating from University Campus School, India, in 1991, he attended Maharshi Dayanand University, also in India, to complete his Bachelor of Arts (Honors) degree in Psychology. In 1996, he completed his MBA degree at Kurukshetra University, India. Subsequently, he worked in the industry for two years before accepting a faculty position at the Institute of Management Studies and Research (IMSAR) at Maharshi Dayanand University. Here, he taught both undergraduate and graduate business courses for three years, before beginning his doctoral studies at Auburn University in 2001. Now completing his Doctor of Philosophy degree in management information systems, he is married to his wife of three years.

KNOWLEDGE INTEGRATION IN SOFTWARE TEAMS: AN ANALYSIS OF TEAM, PROJECT, AND IT-RELATED ISSUES

Nikhil Mehta

Doctor of Philosophy, December 15, 2006
(M.B.A., Kurukshetra University, India, 1996)
(B.A. (Hons.), Maharshi Dayanand University, India, 1991)

241 Typed Pages

Directed by Terry Anthony Byrd

Contemporary organizations are increasingly depending on team-based structures to strategically consolidate their dispersed knowledge resources. Team members possess diverse knowledge resources, and these have to be combined with knowledge from external sources to achieve project goals. Teams achieve this objective by integrating knowledge from external sources and blending it with the skills, know-how, and expertise of the team members. Software teams are an appropriate example of the importance of team-level knowledge integration. Multiple project stakeholders, within and outside the team, possess diverse portfolios of requisite know-how, skills, and abilities and teams must integrate them to develop a timely and workable solution. Prior research suggests that software teams carry out two types of knowledge integration - external integration and internal integration.

The aim of this study is to examine the influence of various team, project, and IT-related antecedents on these two categories of knowledge integration in software teams. Team-related issues include teams' knowledge heterogeneity, relational capital, and boundary-buffering processes. Project-related issues include project uncertainty and project interdependence. IT-usage is examined in a moderating capacity. A research model connecting various categories of antecedents to the two types of knowledge integration was tested by collecting data on 300 projects in nine mid- to large-sized CMM Level 5 software firms. The respondents provided information in light of the most successful project and the least successful project they had experienced. PLS latent variable modeling was used to analyze the data. Two separate analyses were conducted: First, the combined sample of 300 projects was examined to test the research hypotheses; and second, separate analyses were conducted on 150 most successful projects and the same number of least successful ones.

The findings of this study support the influence of a number of team-, project-, and IT-related issues on external as well as internal knowledge integration in software teams. Among team-related issues, knowledge heterogeneity, relational capital, and sentry processes significantly improved knowledge integration, while guard processes had a negative impact on external knowledge integration. Among project-related antecedents, project uncertainty had a significantly negative influence on both internal as well as external knowledge integration, while project interdependence significantly improved external knowledge integration. Interestingly, IT-usage did not moderate the influence of either team- or project-related issues on internal knowledge integration, but significantly improved the influence of these issues on external knowledge integration.

These results provide scholars with a foundation for future research in developing a robust knowledge integration framework. Interesting implications are also in offering for practitioners.

ACKNOWLEDGEMENTS

Ralph Waldo Emerson said, “Our knowledge is the amassed thought and experience of innumerable minds.” The same stands true for the knowledge inherent in this dissertation. Sincere thanks to my dissertation committee for helping me realize my long-cherished dream. Dr. Terry Byrd, my committee chair, will remain my role model for being an excellent teacher and researcher. Dr. Dianne Hall enriched this study deeply with her intellectual inputs, and Dr. Casey Cegielski was always there with very pertinent advice. I also wish to express my gratitude to Dr. Christopher Craighead for his suggestions about improving the questionnaire.

Thanks are also due to Dr. Sharon Oswald for all her help during the PhD program. Her guidance helped me make some very critical decisions. It is also rare to come across a caring person like her. Hope future doctoral students benefit as much from her as I did. Speaking of benefits for doctoral students, I also wish to thank Dr. Nelson Ford for his counsel during difficult times. He has a way to assuage toughest of aggravations. Thanks also to Ms. Nancy Monroe for all her words of wisdom, I cherish them very closely.

I would also like to acknowledge the financial help I received from Auburn University’s Graduate School and from the Dean’s office of College of Business.

This dissertation benefited tremendously from the unflinching support I received from my family. My grandparents, my parents, my wife – Anju, and my brother – Vivek, have all been my pillars of strength. And Moti, I miss you a lot.

Finally, the speck of knowledge created by this dissertation is dedicated to Him, who is pure knowledge, pure Truth. We mortals only play with His reflections!

Style manual or journal used: Auburn University's Guide to Preparation and Submission of Theses and Dissertations and MIS Quarterly

Computer software used: Microsoft Word

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Although firms have engaged in its creation, accumulation, and application for many years (Hansen et al. 1999), only recently has knowledge been identified as a strategic resource (Grant 1996b; Quinn 1992). Defined as a fluid mix of framed experience, values, contextual information, and expert insight (Davenport et al. 1997), knowledge underlies firms’ products and services. Thus, to remain competitive, firms must find better ways to manage their knowledge resources (Spender et al. 1996). However, knowledge typically exists in specialized pockets scattered across the firm and becomes a valuable corporate asset only if it is widely accessible (Davenport et al. 1997; Nonaka 1991). Thus, firms’ capacity to manage their knowledge resources is linked with their ability to better integrate dispersed pockets of specialized knowledge (Kogut et al. 1992; Tsoukas 1996). Teams, supported by information and communication technologies, facilitate this integration (Faraj et al. 2000). Defined as social systems of three or more people, who are interdependent in their tasks, and who share responsibility for their outcomes, teams bring together individually held knowledge, expertise, and specialized skills to bear on tasks of varied nature (Hoegl and Gemuden 2001).

Teams possess diverse knowledge resources and have access to knowledge from various external sources as well. To utilize their internal and external knowledge resources, teams perform knowledge integration, which is defined as the process of absorbing knowledge from external sources and blending it with the skills, know-how, and expertise of the members to create project outcomes (Grant 1996b; Tiwana et al.

2003). Software teams are an appropriate example of the importance of team-level knowledge integration. Software teams, which are typically formed anew for each project, need specialized knowledge and expertise to successfully execute the project (Mathiassen et al. 2003). Multiple stakeholders, within and outside the team, possess diverse portfolios of requisite know-how, skills, and abilities and teams must integrate them to develop a timely and workable solution (Tiwana 2003). Prior research suggests that software teams carry out two types of knowledge integration - *external integration*, i.e., absorbing new knowledge from external sources, and *internal integration*, which includes combining the stocks of internally available knowledge into collective (project) knowledge (Tiwana et al. 2003). Walz, Elam, and Curtis (1993) noted the presence of both external and internal knowledge integration in their four-month long observation of a software team. They found that in absence of the knowledge necessary to execute the project, the team had to integrate knowledge from external sources (peer teams, other departments, and company websites) with internally available knowledge to design the software solution. Faraj and Sproull (2000) also noticed that software teams combined internally available individual stocks of know-how, skills, and abilities into team-level expertise, and utilized this expertise to significantly improve software outcomes. In more current research, Tiwana (2004) examined 232 software development teams and observed that teams coherently integrated different types of external and internal project-related knowledge while designing the software solution. These studies support the presence of both external and internal knowledge integration in software teams. In light of this observation, two interesting questions merit attention: what are some of the antecedents to software teams' knowledge integration? And, what is the nature of their influence? The objective of this research is to answer these questions.

Possible antecedents were searched in the IS, management science, psychology, and knowledge management (KM) literatures. Relevant studies from these literatures, and their inputs, are discussed in greater detail in Chapter 2. Three categories of antecedents emerged. The first category includes team-related issues, such as the heterogeneity or diversity of software teams' internal knowledge resources, teams' relational capital, and their boundary-buffering processes. The second category of antecedents includes project characteristics such as uncertainty and interdependence. And, the third category of antecedents pertains to the teams' usage of various information technologies (IT).

Antecedents to Software Teams' Knowledge Integration

Software teams differ in terms of their ability to integrate external and internal knowledge, and a possible explanation is provided by the heterogeneity of teams' existing knowledge resources (Tiwana et al. 2005). Heterogeneous teams have members with diverse sets of expertise, and members from multiple functional domains. Members of such teams bring with them the access to diverse external networks they have established in their respective domains (Ancona et al. 1992b), which can be utilized to improve the teams' access to external knowledge. Also, the breadth of members' expertise may facilitate teams' integration of these resources, and assist the reception and assimilation of more relevant external knowledge as well (Cohen et al. 1990; Cummings 2004).

Additionally, software teams' knowledge integration may be influenced by the level of their relational capital, which refers to the interpersonal trust, reciprocity, and closeness of working relationships among the team members (Tiwana et al. 2005). Relational capital may influence how much the members trust each other to share their specialized knowledge, and the knowledge they can bring in from external sources (Kale et al. 2000). Team members sharing close working relationships also typically share strong interpersonal ties (Hansen 2002), which may improve their mutual understanding about each other's knowledge resources, and the relevance of these resources to project's knowledge requirements. This may facilitate the teams' efforts to integrate these specialized knowledge resources. Integration of these resources may also be influenced by members' reciprocal attitude towards each other's knowledge sharing efforts. Thus, relational capital may be a key factor influencing knowledge integration.

Another team-related antecedent includes teams' boundary-buffering processes. Previous literature on work unit boundaries forwards an interesting perspective that treats the environment beyond the unit's boundaries as a source of disturbance (Miller et al. 1967; Thompson 1967; Yan et al. 1999). Work units (e.g., teams) perform boundary-buffering processes, such as *sentry* and *guard* processes, to protect themselves from external disturbance (Scott 1992). Teams perform sentry processes to monitor and control the inflow of information and resources from external entities (Pawlowski et al. 2004). Teams also perform guard processes to monitor and restrict the outflow of information and resources to external sources (Guinan et al. 1998).

Sentry processes help teams protect their internal operations from external disturbances, sparing their time and effort to focus on their projects. This may improve teams' understanding of key project-related issues such as the project's knowledge requirements and how best to integrate their internal knowledge resources to fulfill those requirements.

Guard processes may make the teams vigilant towards the external requests for scarce internal resources (e.g., technical or functional experts) (Ancona et al. 1988). By reserving these resources, the teams may better utilize them for their own purposes, thereby facilitating knowledge integration.

This research will examine the potential influence of the above-mentioned team-related antecedents – the two categories of team characteristics (knowledge heterogeneity and relational capital) and the teams' boundary-buffering processes, on their internal and external knowledge integration.

The second category of antecedents includes project characteristics such as uncertainty and interdependence (Andres et al. 2001). As compared to the teams working on projects with less uncertainty, teams working on more uncertain projects have higher knowledge requirements, and they perform more internal as well as external communication and collaboration to fulfill those requirements (Nidumolu 1995; Zmud 1983). Thus, project uncertainty may have a positive influence on teams' internal as well as external knowledge integration.

Teams working on interdependent projects also coordinate and collaborate extensively among each other to avoid making costly mistakes (Andres et al. 2001). Previous research has also observed that interdependent teams possess widespread external networks (Kraut et al. 1995). Active coordination and collaboration between interdependent teams, supported by the usage of their extensive networks as external knowledge sources, may positively influence their external knowledge integration. On the other hand, the teams' focus on external collaboration may hinder their efforts to develop a within-team "shared mental model" of various issues like project problem under consideration, its knowledge requirements, and team members' know-how and skills (Struas et al. 1994). This lack of within team shared understanding about team members' knowledge resources may negatively influence the teams' efforts to integrate these resources.

Based on the above discussion, it will be useful to examine how project uncertainty and project interdependence each influence teams' internal and external knowledge integration.

The third category of antecedents refers to the influence of various information technologies (IT). IT-based systems have gradually evolved from stand-alone systems for individuals to networked systems used by teams for collaboration and communication (Bikson et al. 1999). Contemporary teams rely on various IT-based systems including collaborative systems (e.g., e-mail and group support systems) to help them communicate and collaborate within themselves and with their peer teams, and knowledge management systems (e.g., expert yellow pages and electronic discussion forums) to search for project-related knowledge (Alavi et al. 2001; Orlikowski et al. 1994; Zigers et al. 1994). This study investigates these issues further by examining the moderating influence of teams' IT-usage on the relationships between the team- and project-related antecedents, and teams' knowledge integration.

The IS literature is relatively silent on how these team-, project-, and IT-related issues influence external and internal knowledge integration in teams. To fill this void in literature, this study examines the dependencies of these issues with the integration of internal and external knowledge in software teams. The ensuing sections elaborate upon the two types of knowledge integration, and the three categories of antecedents (team, project, and IT-usage). Their discussion is followed by the research framework that identifies various variables of interest, their proposed interrelationships, the resulting research questions, and expected results. The chapter ends with a brief description of the research methodology and possible constraints of this research.

Knowledge Integration

Knowledge is an important, yet underutilized, resource of software teams (Okhuysen et al. 2002). Team members bring together a wide variety of know-how, skills, and abilities to the software development process. These individual stocks of knowledge are usually inadequate to create project outcomes unless they are integrated and applied to the design and functionality of the software solution (Constantine et al. 1993). Sometimes the required knowledge is not even available inside the team and has to be absorbed from external sources, and integrated with the internally available knowledge to accomplish project objectives. Thus knowledge integration is a critical process in software teams (Walz et al. 1993). As the definition conveys, knowledge integration has an external as well as internal component. They are discussed in the next few paragraphs.

Internal Knowledge Integration

Teams' internal knowledge integration refers to the synthesis and application of individually-held knowledge into team-level systemic knowledge to accomplish project objectives (Alavi et al. 2002; Tiwana et al. 2005). Previous research in strategy literature and the literature on knowledge-based view support this definition by proposing that teams provide an appropriate environment for integration of individually held knowledge into collective knowledge (Grant 1996b; Nahapiet et al. 1998; Schumpeter 1934). Team members integrate internally available knowledge through verbal communication about the project, exchanging tangible artifacts, coordinating their expertise, and sharing information about who knows what (Cummings 2004; Rulke et al. 2000).

In software teams, which are formed anew for each project, and thus lack a common understanding of various facets of the project, internal knowledge integration accrues multiple benefits. An initial round of knowledge integration helps team members develop a shared understanding of project-related issues like the problem under consideration, project's knowledge requirements, and potential solutions (Okhuysen et al. 2002; Walz et al. 1993). Team members then build upon this shared understanding by performing a more intense integration of each other's specialized skills and expertise, and converting it into collective knowledge for developing a robust software solution (Tiwana et al. 2005). This process is influenced by a host of team-related factors including teams' heterogeneity of their existing knowledge resources, team members' level of interpersonal trust, reciprocity, and closeness of relationships, and various external processes teams performs to protect themselves external disturbance (Cohen et al. 1990; Tiwana et al. 2003). Various project characteristics, such as uncertainty and interdependence, may also influence teams' internal knowledge integration (Andres et al. 2001; Nidumolu 1996a; Zmud 1980). For example, because of the lack of critical knowledge inputs in uncertain projects, internal knowledge integration may be less evident in teams working on such projects (Anand et al. 2003; Andres et al. 2001). Software teams' usage of various IT-based systems supports, augments, and reinforces their internal knowledge integration by enhancing the underlying dynamics, scope, timing, and overall synergy of the integration process (Alavi et al. 2001).

Internal knowledge integration is an emerging research area, and although previous research spans literatures as diverse as IS (Teigland et al. 2003; Tiwana 2003a), corporate strategy (Grant 1996a), groups (Okhuysen et al. 2002; Thomas-Hunt et al. 2003), and knowledge management (Ramesh 2002), studies have seldom examined how internal knowledge integration in software teams is affected by various team-, project-, and IT-related issues. This study fills that void.

External Knowledge Integration

External knowledge integration refers to the extent to which teams absorb knowledge from external sources and integrate it with internally available knowledge to bear on project outcomes (Tiwana et al. 2003). In their six-month long observation of a software team, Walz et al. (1993) found that the team members absorbed significant amount of external knowledge to develop the design document.

Teams typically improve operational efficiency by absorbing knowledge from sources outside themselves (Nonaka et al. 1995). Benefits of external knowledge integration are more pronounced for software teams working on interdependent projects with inflated information requirements (Kim et al. 1992-93), or those working on highly uncertain projects with a severe dearth of critical knowledge inputs (Zmud 1980). Teams working on such projects typically exploit their interpersonal networks inside as well as outside the organization to fulfill their knowledge requirements (Kraut et al. 1995).

Previous research suggests that despite its benefits, external knowledge integration is difficult (Szulanski 1996; Zellmer-Bruhn 2003). Teams may lack external processes required to interact with outside knowledge sources (Ancona et al. 1988). Or, the team members may not trust each other enough to share the knowledge they absorb from their interpersonal networks – a key external source of knowledge (Kraut et al. 1995). Additionally, the teams may not possess adequate diversity of internal knowledge resources to apprehend the relevance of knowledge absorbed from external sources (Cohen et al. 1990). It is also possible that the teams are not using IT-based systems (e.g., KMS) enough to absorb externally available knowledge, or using a system with insufficient depth (e.g., e-mail) to absorb context-rich external knowledge (Kankanhalli et al. 2005; Struas et al. 1994).

Little is known about these team-, project-, and IT-related antecedents of external knowledge integration in software teams. Much of the previous research on the topic has adopted an inter-firm perspective (Cummings 2004; Szulanski 1996; Zellmer-Bruhn 2003). There is, thus, a clear need to improve our empirical understanding of external knowledge integration in software teams.

Team Antecedents

This study examines three categories of team antecedents – teams' knowledge heterogeneity, teams' relational capital, and teams' boundary-buffering processes. Each of these is discussed below.

Teams' Knowledge Heterogeneity

Previous research proposes that software teams with more heterogeneity or diversity in their knowledge resources are more likely to possess high levels of knowledge in their project domains (Tiwana et al. 2005). Such teams typically have a better ability to absorb external knowledge in multiple domains as compared to teams with less heterogeneous knowledge (Anand et al. 2003; Cohen et al. 1990). Software teams rarely have all the knowledge required to successfully execute a project and need to absorb knowledge inputs from external sources to compensate for that (Walz et al. 1993). Teams with heterogeneous knowledge resources have a better ability to absorb external knowledge in diverse domains, and may better integrate external knowledge for project purposes (Mowery et al. 1995; Zahra et al. 2002).

On the other hand, teams with heterogeneous knowledge resources may have a high level of dissimilarity among their internal knowledge resources, which may result in an insufficient overlap of the team members' specialized expertise. Members of such teams may lack an appreciation for the relevance and importance of each other's knowledge. This may inhibit a constructive dialogue among the team members about how to combine each other's knowledge resources to fulfill project's knowledge requirements, thus negatively influencing teams' internal knowledge integration.

Not many studies have examined the potential relationship between teams' knowledge heterogeneity and their external as well as internal knowledge integration. As an exception, Tiwana and McLean (2005) studied one of these relationships (heterogeneity and internal integration). They hypothesized a positive relationship between the two, but found a negative relationship. Thus, the relationship between knowledge heterogeneity and knowledge integration requires further empirical examination.

Teams' Relational Capital

Teams' relational capital refers to the level of mutual trust, reciprocity, and the closeness of relationships between team members (Kale et al. 2000). Higher levels of mutual trust among the team members reduces the fear of opportunistic behavior from their colleagues, and improves the confidence that everyone will meet their commitment to one another, and to the project (Bradach et al. 1989; Dasgupta 1988; Nelson et al. 1996). Trusting team members are more willing to share their individual know-how and skills, as well as the knowledge they absorb from external sources. This may positively influence teams' external as well as internal knowledge integration.

Team members with close working relationships enjoy better work coordination (Tiwana et al. 2005), as a result of which they have a better idea of issues like who possesses what knowledge and expertise, what kind of external knowledge sources do the members have access to, and how the knowledge from various internal and external sources be combined to fulfill project's knowledge requirements. Combined with high levels of mutual trust, strong interpersonal ties also elicit reciprocal behavior among team members, which improves the integration of sticky and tacit knowledge among them (Marsden 1990; von Hipple 1988). This may positively influence teams' external as well as internal knowledge integration. Teams' relational capital thus offers an interesting behavioral perspective toward knowledge integration. This study will empirically examine this perspective.

Teams' Boundary-Buffering Processes

During the course of their projects, software teams need to manage their dependencies on various entities inside as well as outside the organization. For example, teams have to manage both the inflow and outflow of knowledge and other resources from and to external entities (Yan et al. 1999). Effective management of these across-boundary dependencies requires teams to perform special external processes (Aldrich et al. 1977; March et al. 1958). Previous studies have observed that as compared to teams that only carry out internal processes, teams performing both internal and external processes are not only able to obtain key project-related resources from external entities but also manage their internal dynamics better, thereby producing better project outcomes (Allen 1977; Pfeffer 1986).

This study focuses on a specific category of inward-looking external processes, called boundary-buffering processes that a team carries out to: (1) prevent its internal operations from undesirable external interference; and (2) cope with external resource requests that may be detrimental to how effectively teams utilize their internal resources.

Teams perform boundary-buffering processes to prevent their internal operations from unwarranted external interference, and to cope with requests for their internal resources. Yan and Louis aptly explain the importance of boundary buffering processes:

"The buffering function of boundaries stresses the need to close the system off from environmental disturbances in order to enhance the possibility of rational action within the system" (1999: 31).

Previous literature has discussed *sentry* and *guard* processes as two types of boundary-buffering processes (Ancona et al. 1988; Yan et al. 1999). Sentry processes help monitor and control the inflow of information from the external environment (Pawlowski et al. 2004). Teams that perform sentry processes avoid external interference and are able to focus on their internal operations. This helps them better understand various project and team-related issues, such as the project's knowledge requirements and how to integrate the teams' internal knowledge resources to fulfill those requirements. Additionally, teams performing sentry processes keep their information-processing infrastructure relatively free of unwarranted external information inputs, and can better utilize that infrastructure to integrate their internal knowledge resources.

Guard processes, as the name suggests, monitor the outflow of teams' resources to external environment (Pawlowski et al. 2004). Software teams initiate guard processes typically to evaluate the external requests based upon their legitimacy and their cost to the team (Ancona et al. 1988). For example, resource-related requests may be denied if the team requires the requested resources (e.g., a technical expert) for its own purpose. Thus, by being selective in fulfilling external requests, teams performing guard processes are able to better utilize their internal knowledge resources for their own purposes, which may positively influence their internal knowledge integration.

Although researchers in the IS field have stressed the importance of external processes (Guinan et al. 1998; Markus 1983; Zmud 1983), literature is generally silent regarding their influence on teams' knowledge integration. This study examines this influence empirically.

Project Antecedents

Software development is a rapidly evolving field. Knowledge in the field is extensive, and is growing quickly. Software teams increasingly struggle with projects characterized by complex technologies and software architectures, ambiguous customer requirements, and unpredictable outcomes (Komi-Sirvio et al. 2002). Additionally, more and more teams now work on interdependent projects, thus facing ever-increasing demands of coordination with their peer teams and other sources inside as well as outside their organization (Kraut et al. 1995). These changes underscore the importance of two project-related variables – project uncertainty and project interdependence, to knowledge integration in software teams. Although an emerging body of literature in the field has examined these project-related issues in various capacities, their respective influence on teams' internal and external knowledge integration has seldom been examined, especially in light of IT-usage as a moderating factor. This study fills that void. These issues are discussed in the following paragraphs.

Project Uncertainty

Project uncertainty can be broadly defined as the inadequacy of critical knowledge about the project, which reduces the teams' ability to successfully plan project execution and to predict its outcomes (Nidumolu 1996b). Software teams working on uncertain projects frequently reach junctures where they experience a dearth of critical knowledge inputs (Zmud 1980). Team members typically engage in formal as well as informal communication and collaboration among themselves to improve their understanding of the required knowledge inputs (Galeghar et al. 1994). Teams then fulfill these requirements typically by substantiating their internal knowledge stocks with knowledge inputs absorbed from their external networks (Kraut et al. 1995). Thus, it seems that teams working on uncertain projects actively engage in internal collaboration to better understand their knowledge requirements, and then try to offset their knowledge inadequacies by integrating external knowledge inputs with internally available knowledge (Anand et al. 2003).

We don't know much about this proposition, and given the fact that most problems in software projects can be traced to the uncertainty that pervades them (Zmud 1980), it does not bode well for both research and practice in the field of software development. By empirically examining the influence of project uncertainty on teams' external as well as internal knowledge integration, this study sheds some light on the relationships between these variables.

Project Interdependence

Project interdependence refers to "the extent to which a project requires various organizational units to engage in workflow exchanges of product, information, skills, or resources, and to where actions taken in one unit affect the actions and work outcomes of other units" (Andres & Zmud, 2001: 44). Teams working on interdependent projects need to synchronize various technical details, and sequence the connected activities to meet the given schedule and budgetary constraints (Sabbagh 1996). To avoid making costly mistakes, these teams coordinate and communicate extensively with each other (Andres et al. 2001). Additionally, they use their widespread external networks to absorb project-related knowledge from external sources (Kraut et al. 1995). Thus, it appears that teams working on interdependent projects tend to engage more in external knowledge integration as compared to the teams working on standalone projects.

On the other hand, members of such teams spend much of their time in external communication and coordination activities, and thus are not spared with enough time to develop a within-team “shared mental model” of various project and team-related issues (Struas et al. 1994). This may have a number of negative repercussions. For example, team members may lack an awareness of each other’s know-how, skills, and abilities, a fact that may inhibit teams’ integration of its internal knowledge resources. In light of these issues, this study empirically examines the relationship between project interdependence and teams’ internal as well as external knowledge integration. Previous studies on project interdependence have seldom examined these relationships. Additionally, most other studies examining project interdependence have utilized student teams as their sample. Student teams are not an accurate indicator of more experienced professional software teams, and earlier studies, while accepting it as a limitation, have suggested empirical testing of interdependence-related issues in real world software teams (Andres & Zmud, 2001-02; Straus & McGrath, 1994).

Antecedents Related to IT-Usage

In their groundbreaking review, (Alavi et al. 2001) raised a number of engaging questions about the influence of information technology (IT) to knowledge processes. Their assertion that IT influences knowledge processes such as integration, inspires yet another question for this research – how? Prior studies have fueled a lively debate on whether IT-usage facilitates or inhibits knowledge integration. Some argue that IT-usage improves integration of fragmented stocks of explicit knowledge by improving its storage and transfer (Lee et al. 2003; Roberts 2000; Scott 1998). Others argue that using explicit knowledge captured in IT-based systems (e.g. a KM System) is not a true example of knowledge integration (Cole 1998). Moreover, using IT-based systems for explicit knowledge integration is only helpful if the knowledge seeker clearly knows what knowledge to obtain (Alavi et al. 2001; Powell 1998). In light of these diverse viewpoints, this study adopts a different perspective to study the influence of IT-usage on knowledge integration. It examines how teams' IT-usage moderates the influence of team- and project-related determinants on external as well as internal knowledge integration.

Previous research supports the moderating role of IT-usage. For example, past studies suggest that by allowing team members to exchange knowledgeable inputs with external sources, IT systems may moderate the negative influence of teams' boundary buffering processes on their external knowledge integration (Guinan et al. 1998). IT-usage may also positively moderate the efforts of teams with diverse expertise to absorb external knowledge, as experts are better aware of what knowledge to obtain in their respective domains (Alavi et al. 2001; Powell 1998).

Teams' IT-usage may also have a moderating influence on how project uncertainty influences the external and internal knowledge integration. Teams working on uncertain projects may use more IT-based systems to integrate knowledge from external sources. On the other hand, uncertain projects may entail team members to carry out internal knowledge integration in more face-to-face settings (as compared to IT-based settings), thus requiring less usage of IT-based systems. Teams working on interdependent projects may use more IT-based systems to communicate and collaborate with each other, thus facilitating their external knowledge integration. On the other hand, such teams may already lack a within-team shared mental model about the project, and more usage of IT-based systems (as compared to face-to-face interactions) may inhibit the already restrained internal knowledge integration in such teams.

These moderating effects of IT-usage, discussed and proposed in the research hypotheses in Chapter 2, will be examined in this study.

Research Objectives and Plan

This dissertation is a methodical and empirical assessment of the various team, project, and IT-based antecedents to external and internal knowledge integration in software teams. This section addresses the purpose of this dissertation, the research questions and expected findings, assumptions, limitations, and expected contributions.

Conceptual Framework

To enhance clarity, the research model is proposed (see Figure 1). The model includes inputs from multiple literature streams including information systems, knowledge management, organizational behavior, teams, and boundary spanning. These inputs are discussed in detail in the next chapter. Research questions emerging from the model are discussed further.

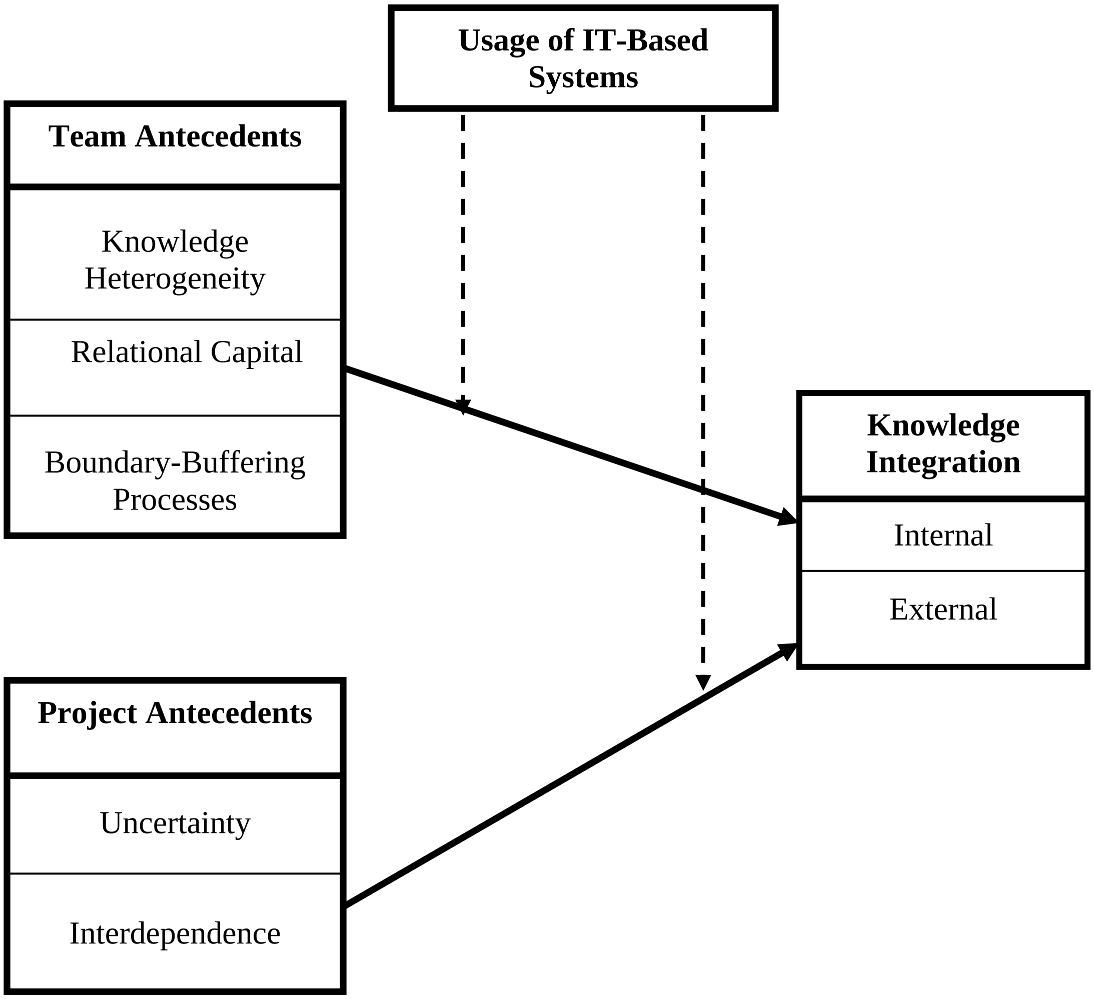


Figure 1.1 Conceptual model

Purpose of Study and Research Questions

The aim of this study is to assess the influence of various team- and project-related antecedents might play in teams' external and internal knowledge integration, particularly in light of teams' IT usage. Thus, the central research questions are:

1. How does heterogeneity of software teams' knowledge resources influence their internal and external knowledge integration?
2. How does software teams' level of relational capital influence their internal and external knowledge integration?
3. What is the influence of software teams' boundary-buffering processes on their external and internal knowledge integration?
4. What influence do project characteristics such as uncertainty and interdependence have on software teams' internal and external knowledge integration?
5. How does software teams' IT-usage moderate the influence of team-related antecedents (knowledge heterogeneity, relational capital, boundary-buffering processes), and project-related antecedents (uncertainty and interdependence) on teams' internal and external knowledge integration?

It is expected that the results of the study will help uncover a positive relationship between teams' knowledge heterogeneity and their external and internal knowledge integration respectively. A similar relationship is expected between the level of teams' relational capital and their internal integration. Teams' boundary-buffering processes are expected to negatively influence their external knowledge integration. IT-usage is expected to moderate these relationships. Teams' boundary-buffering processes are also expected to positively influence their internal knowledge integration.

Project uncertainty is expected to positively influence external as well as internal knowledge integration respectively, though project interdependence may have a positive influence on external knowledge integration and a negative influence on teams' internal knowledge integration. IT-usage is expected to moderate these relationships too.

To examine the proposed relationships, questionnaire-based data will be collected from 150-175 project leaders in nine mid- to large-size software firms. These firms provide custom-made software solutions to Fortune 1000 clients. They were chosen because they are comparable in terms of their size, their nature of operations, and their financial performance. The firms possess CMM Level 5 certification. Within these firms:

- Project leaders will be selected from multiple sites to maximize geographical diversity.
- Only project leaders that have prior experience with, or are currently leading, teams developing business applications software or developing packaged software will be selected.
- Project leaders of teams with a minimum of 5-7 full-time members but fewer than 20 members will be selected to control for the dynamics and processes related to team size.

The questionnaire will be developed using existing measurement scales where possible. New measures will be developed only in the absence of existing measures for a construct. These will be developed based on detailed discussion with six academic experts and 12 software project leaders in three firms, which are different from the nine firms identified for data collection.

Each construct will be measured using four to six questionnaire items. All responses will be measured using seven-point Likert scales and using objective data. The instrument will be refined using the traditional convergent validity and discriminant validity procedures, and the items exhibiting sufficient discriminant validity and construct scale reliability will be retained.

Expected Contributions

The primary contribution of this research is to improve our understanding of software teams' knowledge integration by examining the influence of various team-, project-, and IT-related issues. The empirical results from this study are expected to help academics and practitioners alike. In academia, the discussion of teams' knowledge integration is a relatively new area of exploration, and this study builds upon the few (but nonetheless significant) past research inquests within in the fields of IS, management science, KM, and psychology. Thus, the results of this study will interlink these fields and provide a foundation for future inter-disciplinary research. Additionally, a better understanding of teams' knowledge integration will also contribute to KM literature that is characterized more by the firm-level examination of this process.

For practitioners, especially ones in team-based organizations, the results of this study will improve their understanding of knowledge integration *within* their core building blocks i.e., teams. Managers will also be able to develop a knowledge integration profile of their teams in light of the antecedents considered in this study. For example, they can identify the characteristics of teams that are more of 'external knowledge integrators' versus those that are more of 'internal knowledge integrators'.

Additionally, managers can gain better understanding of how teams use IT-based systems to buttress knowledge integration.

Organization of Dissertation

Five chapters constitute this dissertation, including this introduction, which discusses the importance of external and internal knowledge integration in software teams, and their relationships with the team-, project-, and IT-related antecedents. In light of that discussion, chapter one also outlines the purpose of this research, the research questions addressed, the methodology, and a brief statement of the contributions of the study in view of its limitations. Chapter 2 reviews the existing literature that motivates the conceptual model of this dissertation, and develops pertinent hypotheses. Chapter 3 explains and justifies the methodology. Chapter 4 presents the empirical findings of the study, and chapter 5 discusses those findings and their implications. The final chapter, while admitting the limitations, also highlights the recommendations for further research.

CHAPTER 2: THEORETICAL BACKGROUND

The dependent variable in this study is software teams' integration of external and internal knowledge resources. Knowledge integration refers to teams' assimilation of knowledge from external sources, and synthesizing it with members' know-how, skills, and expertise to produce project outcomes (Alavi et al. 2002; Tiwana et al. 2003). The objective of this study is to examine how various team- and project-related antecedents influence both external as well as internal knowledge integration in software teams. Team-related antecedents include teams' knowledge heterogeneity, teams' relational capital, and teams' boundary buffering processes. Project-related antecedents include project uncertainty and project interdependence. Usage of various IT-based systems is examined for its moderating influence on various interrelationships between the team and project antecedents and teams' external and internal knowledge integration.

Given the lack of a coherent body of research on knowledge integration, contributions from multiple streams of literature helped develop the theoretical framework for this study. For example, theoretical support for teams' knowledge heterogeneity was drawn primarily from management science literature, while the organizational behavior literature contributed to the development of the relational capital construct. Literature on boundary spanning was found to be helpful in discussing the boundary buffering processes. Theoretical underpinnings for the two project characteristics – uncertainty and interdependence were developed from literature in IS, software development, management science, and psychology. Previous research in IS and knowledge management underlies the discussions of the moderating influence of IT-usage.

In the following sections, team- and project-related antecedents are discussed followed by the development of hypotheses regarding their respective relationships with internal and external knowledge integration. Following that, the IT-usage construct is discussed, and the research hypotheses predicting its moderating influence on the relationships between the independent and dependent variables are developed. The last section presents the research models (see Figure 2.1 and Figure 2.2).

Team Antecedents

Knowledge heterogeneity and relational capital are discussed in more granularity below to better understand their importance in this research. The discussion also operationalizes these variables.

Teams' Knowledge Heterogeneity

Team members are the primary repositories of teams' knowledge (Argote 1999). Software teams' knowledge heterogeneity is defined as the diversity of members' technical and functional background and their expertise and skills (Anand et al. 2003; Smith et al. 2005). Heterogeneous teams, as compared to homogeneous teams, have members with more diverse backgrounds, who bring in multiple sets of expertise and skills (Tiwana et al. 2005).

Previous research in the field of knowledge heterogeneity presents two conflicting perspectives on its probable relation with knowledge integration. The classical viewpoint suggests that teams with members holding multiple sets of expertise and skills also possess the opportunity to integrate them and develop better project outcomes such as the problem formulation document, requirements definition, software architecture design, and the software solution (Curtis et al. 1988; Nahapiet et al. 1998).

On the other hand, some recent studies have proposed that software team members from different technical and functional domains lack a sufficient overlap in their understanding of each other's uniquely held knowledge, which may hinder with the teams' efforts to integrate these knowledge resources (Rulke et al. 2000; Tiwana et al. 2005).

This study adopts the classical perspective, and argues that although software teams consist of members from diverse technical and functional domains, most of the members typically have common educational backgrounds, and are aware of various techniques of software engineering and computer science (Curtis et al. 1988). This provides for some degree of overlap in their mutual understanding of each other's domain specific knowledge, however diverse their domains may be.

Also, heterogeneous teams have large knowledge bases, so they bring in more relevant inputs to the knowledge integration process (Smith et al. 2005). Members of heterogeneous teams have diverse expertise, and they typically have a better understanding of the knowledge-related inconsistencies in the project, a better ability to recall project relevant information, and a more elaborate schema of how to apply their knowledge to the project (Fiske et al. 1983; Lord et al. 1990). The influence of these heterogeneity-related issues on knowledge integration will be discussed in greater detail while developing the research hypotheses later in this chapter.

Teams' Relational Capital

A certain degree of mutual trust develops among team members over the course of a project (Gulati 1995; Lewicki et al. 1995), which helps members form close working relationships (Kale et al. 2000), characterized by a positive give-and-take attitude. Such mutual trust, closeness of relationships, and reciprocity that dwells among the team members is referred to as the relational capital (Tiwana et al. 2005).

Mutual trust is defined as the belief that the "results of somebody's intended actions will be appropriate from our point of view" (Misztal 1996: 9-10). A team high in trust has more confidence among its members that everyone will meet their knowledge commitments toward each other, and toward the team goals. Additionally, trusting team members are less suspicious of each other's opportunistic behavior (Bradach et al. 1989), which may reduce their apprehensions about sharing their uniquely held knowledge with their colleagues.

Team members with higher levels of relational capital also enjoy close working relationships (Kale et al. 2000). Previous studies have reported that team members may exchange knowledge relatively easily with coworkers with whom they share close working relationships (Teigland et al. 2003). Close working relationships among team members also reduce the emotional labor typically involved in people's relationships (Szulanski 1996), thereby improving the quality of interactions among them, and increasing the likelihood of interpersonal knowledge exchange. This may facilitate the teams' efforts to integrate the expertise of their members.

Teams' relational capital is also characterized by the reciprocal behavior of their members. People expect reciprocal behavior when engaging in knowledge exchange with others (Lakhani et al. 2000). For example, the “norms of participation in electronic networks typically dictate that those who seek and receive help from the network must also pay back by helping others” (Teigland et al. 2003: 267). In context of teams, members of teams with low relational capital are less expected to voluntarily reciprocate with their uniquely held knowledge, beyond their knowledge commitments to the project (Tiwana et al. 2005). Thus, members of teams with high relational capital may exhibit a healthy knowledge-reciprocating attitude.

Software teams’ relational capital, as indicated by the level of mutual trust, closeness of relationships, and reciprocity among their members will be discussed in greater detail while hypothesizing its influence on teams’ internal and external knowledge integration.

Teams’ Boundary-Buffering Processes

Discussion of teams’ boundary-buffering processes is motivated by the conception of teams’ boundary. Alternative perspectives on teams’ boundary can be associated with different categories of boundary-buffering processes. One perspective describes boundary as a perimeter that defines the sphere of influence of teams, and considers the external environment as sources of interference and disturbance to the teams’ internal functioning (Yan et al. 1999). Even teams’ interaction with the external entities, on which they depend for critical resources, is not free of such disturbance. Inputs (e.g., members and knowledge) from such sources may adversely affect teams’ stability by challenging their existing setup, philosophy, and in extreme cases, their functioning (Friedlander 1985). Therefore, teams typically perform *sentry* processes to protect their internal operations from external interference (Thompson 1967). Sentry processes help teams manage external influence (Pawlowski et al. 2004). For example, teams performing sentry processes typically close their boundaries to unwanted information inputs from external sources. Useful inputs are filtered-in and converted into the desired form (Ancona et al. 1988). By allowing selectivity in accepting external information inputs, sentry processes keep teams' information-processing capabilities relatively free, which can be utilized to support their internal knowledge integration efforts.

External entities may also create disturbance by requesting for the teams' limited internal resources (e.g., a technical expert), thus challenging teams' utilization of these resources. Teams typically perform *guard* processes to supervise such external requests (Pawlowski et al. 2004). By performing guard processes, teams are able to evaluate the requests on two grounds – legitimacy and cost to the team (Ancona et al. 1988). Teams may consider relevant organizational policies and discuss among members to decide on these parameters. Legitimate and less expensive requests are fulfilled, while the rest are denied to protect teams' interests. Thus, by astutely sifting the external requests for their internal resources, teams performing guard processes may also effectively utilize their internal knowledge assets (Guinan et al. 1998), a factor that my facilitate the teams' integration of those assets for project purposes.

Project Antecedents

The two project-related antecedents being examined in this study include project uncertainty and project interdependence. In the next few sections, relevant literatures in

IS, software development, management science, and psychology are explored to discuss these antecedents in greater detail.

Project Uncertainty

Earlier research in psychology, management science, and IS characterize uncertainty as absence of information (Daft et al. 1986; Garner 1962; Kydd 1989). A key difference between project uncertainty and another related concept - project equivocality, is that uncertainty suggests a lack of information about the project, while equivocality connotes the existence of multiple and conflicting interpretations of various project-related issues. So, while equivocality might be reduced by exchange of existing knowledge between people, uncertainty may require acquiring new knowledge from external sources (Daft et al. 1987). At the organizational level, uncertainty is defined as the difference between the amount of information required to perform the task and the amount of information already possessed by the organization (Galbraith 1977). Previous studies suggest that project uncertainty emanates from issues such as a lack of agreement about organizational objectives, the information requirements of those objectives, and the procedures required to fulfill the information requirements (Daft et al. 1986; Ungson et al. 1981). Unlike equivocality, where people are not sure of what questions to ask, uncertainty allows asking questions. Additional information can then be created internally, or acquired from external sources to answer those questions (Kydd 1989).

Software development is a knowledge-intensive activity, and uncertainty in software projects is related to the lack of critical knowledge inputs pertaining to various project areas (Zmud 1980). Previous studies have identified project requirements, project technologies, and project outcomes as three key areas in which knowledge scarcity escalates project uncertainty (Beath 1983; Nidumolu 1995; Nidumolu 1996). Teams working on uncertain projects typically lack a clear understanding of project requirements and are unsure of appropriate technologies required to convert those specifications into software solution. Uncertainty emanating from unclear project requirements and unfamiliar technologies also confounds the ability of teams to predict project outcomes (Thompson 1967). For example, previous studies have discussed project uncertainty in light of the amount of time required before project outcomes can be predicted (Lefton et al. 1966; Van de Ven et al. 1976). Based on the above discussion, uncertainty in software projects is broadly defined as the inadequacy of requirements- and technology-related inputs, which reduces the teams' ability to predict their project outcomes.

As users' desire more and more innovative and unstructured software applications for their businesses, project teams building such applications find it increasingly difficult to acquire a clear understanding of project requirements (Kraut et al. 1995). Uncertainty pertaining to project requirements can be examined in terms of their instability, which refers to how the requirements change over the course of the project, and in terms of their diversity, which highlights the differences in the users' perspectives toward the requirements (Nidumolu 1996). Unstable and diverse requirements can potentially jeopardize software development and its implementation (Turner 1992).

The second key source of uncertainty in software projects is technology. Contemporary projects increasingly involve multiple and state-of-the-art technologies and teams typically need to decide upon the most appropriate set of technologies to convert project specifications into software solution. In the earlier stages of the project, teams need to minimize technological uncertainty to better estimate their resource commitments to the project, and to better predict implementation-related issues (Nidumolu 1995). In the later stages of the project, technological uncertainty can manifest itself when the chosen technology creates unexpected or novel problems while being used (Nidumolu 1995; Perrow 1967). For these reasons, technological uncertainty, if not addressed appropriately, can heighten the risk of project failure (Boehm 1989).

To reduce specifications and technological uncertainty, teams need to fulfill their knowledge scarcity (Daft et al. 1987). Previous research has observed that teams working on more uncertain projects have extensive external networks as compared to those working on less uncertain projects (Kraut et al. 1995). To compensate for their internal knowledge scarcity, such teams may utilize their networks to absorb external knowledge. This and other possible relationships of project uncertainty with teams' knowledge integration are discussed in detail in the hypotheses section of this chapter.

Project Interdependence

Project interdependence represents the degree to which a project requires participating teams to exchange information, skills, and resources with one another to successfully complete the project. Software development is a rapidly evolving field, with knowledge in the field growing quickly. Software teams increasingly find themselves working on the fringe-lines of knowledge about new technologies, business domains, and software development practices. To address this issue, software firms typically assign a project to multiple teams that share a diverse pool of knowledge resources (Kirsch 1996).

Such teams are dependent on one another for successful completion of the project. They typically share two types of interdependence – task and outcome (Andres et al. 2001).

Task interdependence is characterized in terms of project-related inputs of participating teams, and the processes by which they complete the project (Wageman 1995). Previous research in psychology has defined task interdependence in terms of workflow exchanges of materials and objects among organizational units (Thompson 1967; Tushman 1979). Earlier studies in the IS field continued to follow this conception of project interdependence (Kim et al. 1992-93). However, more recent research has argued that teams working on interdependent projects also influence each other's work outcomes by their actions, and has added another dimension - outcome interdependence, to the examination of project interdependence (Andres et al. 2001). Defined as the degree to which the outcomes of a team (e.g., goals, progress, and rewards) depend on the performance of other teams, outcome interdependence, when added to task interdependence, allows a more robust examination of the project interdependence construct (Campion et al. 1993; Wageman 1995). For example, earlier studies have found that the task progress of a team member was constrained when the task was dependent on performance of other members (Saavedra et al. 1993). It was also observed that high interdependence among new product development teams decreased their schedule performance (adherence to project schedule) as well as the budget performance (adherence to project budget) (Hoegl et al. 2004). Based on the discussion above, this study includes both task and outcome interdependence as indicators of project interdependence.

To avoid making costly mistakes, effective coordination is required among teams working on interdependent projects (Kazanjian et al. 2000; Loch et al. 1998). For example, teams need to synchronize various project stages (e.g., requirements specification, design specification, coding, testing, documenting, and implementing) and arrange the linked activities to meet project's schedule and budgetary constraints (Sabbagh 1996). Issues pertaining to inter-team coordination and information exchange may help develop the possible relationships between project interdependence and teams' internal and external knowledge integration. They are discussed in more detail in the hypotheses section of this chapter.

Hypotheses Predicting Main Relationships

This section develops various relationships between the team- and project-related antecedents and software teams' internal and external knowledge integration. Developing hypothesis pertaining to these relationships will help us better understand the role of each of these constructs in the research framework. In the following sections, theoretical arguments supporting each of the hypotheses are presented.

Teams' Knowledge Heterogeneity and Teams' Knowledge Integration

A team's knowledge heterogeneity, which connotes the presence of diverse knowledge resources among the team members, fulfills a fundamental pre-condition for knowledge integration (Moran et al. 1996). More specifically, knowledge distribution among team members helps define the initial structure of the integration of that knowledge (Lewis 2004). This may happen because members of such teams are likely to expect that everyone needs to contribute their unique knowledge to accomplish project goals. Such expectations regarding each other's contributions may provide the groundwork for better integration of team's knowledge resources by making the members accountable for knowledge inputs in their respective domains and to rely on others for complementary inputs (Hollingshead 2001; Lewis 2004).

Additionally, previous research suggests that team members with diverse backgrounds are more likely to have divergent worldviews and motivations (Lawrence et al. 1986). Members of such teams are likely to have differing interpretations of various project related issues and may find it difficult to reach an agreement on those issues, and typically experience higher cognitive dissonance than homogeneous teams (Jehn et al. 2001; Maruping et al. 2004; Nemeth 1992). To reduce this dissonance, they may need to stimulate information sharing and task-related debates to create an overlapping understanding of each other's worldview. In other words, they may need to develop a shared context. Once a shared context is created, teams can actively integrate members' perspectives and ideas, and create novel approaches to conceptualize and execute the project.

Regarding the influence of knowledge heterogeneity on external knowledge integration, heterogeneous teams with members from diverse backgrounds, may have interpersonal networks in diverse domains, which can be utilized to acquire external knowledge inputs. Furthermore, heterogeneous teams have experts in multiple domains, and experts as compared to novices, have a better understanding of the knowledge inconsistencies in the project, a clear idea of what knowledge to obtain from external sources, and a more elaborate schema of how to apply their knowledge to remove the inconsistencies (Fiske et al. 1983; Lord et al. 1990). Heterogeneous teams will thus have better capacity to integrate external knowledge in multiple domains (Cohen et al. 1990), as compared to homogeneous teams which have experts in the same domain, and thus have a high capacity to integrate external knowledge only in that domain (Anand et al. 2003). Additionally, previous research suggests that people tend to acquire and learn more information in their own domains if they believe that others have different rather than similar expertise, and when task outcomes are contingent on members contributing different but complementary information (Hollingshead 2000; Hollingshead 2001; Wittenbaum et al. 1998).

Therefore, heterogeneous teams, as compared to homogeneous teams, may not only have better access to the external knowledge sources in multiple domains but also have higher capacity and motivation to integrate knowledge in those domains.

Based on the discussion above, the respective relationships between software teams' knowledge heterogeneity and their internal and external knowledge integration are hypothesized below:

Hypothesis 1a: A software team's knowledge heterogeneity positively influences its internal knowledge integration.

Hypothesis 1b: A software team's knowledge heterogeneity positively influences its external knowledge integration.

Teams' Relational Capital and Teams' Knowledge Integration

A software team's relational capital improves its inherent transparency and openness, which may create a suitable environment for knowledge integration (Nahapiet et al. 1998). Specifically, a team with higher levels of relational capital will have a higher level of mutual trust among its members. Trust improves members' confidence in each other by reducing their mutual suspicion regarding opportunistic behavior, and providing a context for more cooperative interaction among them (Tsai et al. 1998). Trust also improves interpersonal communication and dialogue in the team (Misztal 1996). Thus, we can say that trust may alleviate mutual suspicion about misuse of members' unique knowledge (Davenport et al. 1998). Also, by improving interpersonal communication within the team, trust may also create a climate conducive to internal knowledge integration. High level of mutual trust also motivates team members to take risks in exchanging their uniquely held knowledge (Nahapiet 1996; Ring et al. 1994), which may improve team's efforts to try new combinations of integrating its internal knowledge resources, and their integration with the knowledge acquired from external sources (Nahapiet et al. 1998).

Team's relational capital also includes closeness of working relationships between its members. Team members with close working relationships may gradually develop strong interpersonal ties, and thus, be able to interact more frequently and communicate more effectively. Frequent interactions and effective communication among members help develop a robust mutual understanding about issues like what are project's knowledge requirements and who possesses relevant technical or functional knowledge internally, or has access to appropriate knowledge outside the team (Ko et al. 2005). Close and intense interactions between the members also improve the exchange and subsequent integration of more sticky and tacit knowledge across individual interface (Marsden 1990; Tiwana et al. 2005).

Additionally, members sharing close, trustworthy relationships freely exchange their knowledge with each other, as they are confident of being reciprocated (Kale et al. 2000). Reciprocity may improve team's internal knowledge exchange situation, as members are more likely to share their uniquely held knowledge with each other. This may reinforce team's efforts to combine these knowledge resources in innovative ways to create project level knowledge. Additionally, reciprocity may even motivate team members to acquire knowledge requested by their colleagues from their external sources.

In light of the above discussion, the relationships between a software team's relational capital and its internal and external knowledge integration is hypothesized as follows:

Hypothesis 2a: A software team's relational capital positively influences its internal knowledge integration.

Hypothesis 2b: A software team's relational capital positively influences its external knowledge integration.

Boundary Buffering Processes and Teams' Knowledge Integration

Previous literature on work unit boundaries forwards an interesting perspective that treats the environment beyond the unit's (e.g., a team) boundaries as a source of disturbance (Miller et al. 1967; Thompson 1967; Yan et al. 1999). A software team has multiple dependencies with external environment inside as well as outside the organization. Team's boundaries are thus overwhelmed with the constellation of various external sources, and team members have to perform boundary-buffering processes (especially sentry and guard processes) to protect themselves from disturbances emanating from such sources. By keeping out external interference, the two boundary-buffering processes enhance the team's ability to effectively manage its time, effort, and knowledge to produce project outcomes (Yan et al. 1999).

Organizational adoption of advanced IT has increased the likelihood of team members being exposed to huge amount of information from sources inside and outside the organization. Sentry processes help monitor and control this flow of information from external entities into the team (Pawlowski et al. 2004). Software teams performing sentry processes close their boundaries to undesired inputs from external sources, thereby preventing their information-processing capacities from being overloaded (Jemison 1984), and making them available for internal knowledge integration instead. Also, teams performing sentry processes are able to work on their projects with minimal external interference. This helps them focus on their project, and thus develop a better understanding of various team and project-related issues. With this understanding, such teams are more likely to better integrate their internal knowledge resources in to achieve project goals. Thus, it is hypothesized:

Hypothesis 3a: A software team's sentry processes positively influence its internal knowledge integration.

Guard processes, on the other hand, help teams manage the movement of their resources to external entities. Software teams usually have limited resources (e.g., members with a particular expertise or skill) (Walz et al. 1993), and given the projects' deadlines and budgetary constraints, teams are hard pressed to utilize their resources in the most effective manner. Under these circumstances, software teams that perform guard processes take selective decisions about fulfilling external requests for their resources

(Jemison 1984). Such teams are able to better utilize their internal resources (e.g., knowledge) for their own purposes. Therefore, it is hypothesized:

Hypothesis 4a: A software team's guard processes positively influence its internal knowledge integration.

On the other hand, organizational units (such as software teams) policing their boundaries though sentry and guard processes may have to incur certain costs (Aldrich et al. 1977). For example, members of teams performing sentry processes may not have an open access to external knowledge sources and thus may not be aware of project-relevant knowledge held by those sources. This may depress teams' knowledge acquisition from external sources (Awazu 2004), and therefore have a negative influence on their knowledge integration from those sources. Teams performing guard processes are also more likely to restrict their internal knowledge from crossing team boundaries. Such teams may earn a negative reputation of being non-cooperative, and may elicit a similar response if they required knowledge inputs from external sources. A dearth of external knowledge inputs may reduce teams' external knowledge integration. In view of these arguments, it is predicted:

Hypothesis 3b: A software team's sentry processes negatively influence its external knowledge integration.

Hypothesis 4b: A software team's guard processes negatively influence its external knowledge integration.

Project Uncertainty and Teams' Knowledge Integration

Zmud (1980) identified software project uncertainty with the lack of critical knowledge inputs in various project-related areas. For the purpose of this study, project uncertainty is defined as the inadequacy of information inputs regarding requirements specifications and technological issues, which reduces the ability of software teams to predict project outcomes. No wonder project uncertainty is identified as a key practical problem in effective management of software projects (McFarlan 1981).

Regarding inadequacy of information for requirements specifications, Zmud (1980) proposes that contemporary software teams need to keep up with constantly changing software requirements, and anticipate, plan, and control their development efforts accordingly. To carry out their activities, teams need to regularly absorb specifications-related knowledge from the external environment (Curtis et al. 1988; Kraut et al. 1995). Previous research suggests that sourcing these inputs through vertical coordination between the project manager and the users (Van de Ven et al. 1976), can reduce project uncertainty (Nidumolu 1995). Teams working on uncertain projects also need to coordinate vertically with other groups, such as top management, to obtain slack knowledge resources (Galbraith 1977). Both, the requirements-related inputs and the slack resources can then be integrated with the teams' internal knowledge resources to buffer the project from hazards of “requirements creep” (Nidumolu 1995). Based on these arguments, it can be proposed that teams will engage in more knowledge integration to reduce specifications-related project uncertainty.

To reduce technological uncertainty, teams initiate informal horizontal communication with external sources (Andres et al. 2001; Galeghar et al. 1994). It has been observed that teams working on less certain projects typically have extensive interpersonal networks as compared to those working on more certain projects (Kraut et al. 1995). Teams compensate for their knowledge scarcity by frequently seeking and integrating knowledge inputs from these networks (Anand et al. 2003; Hoegl et al. 2004). For example, in the early stages of the project, teams may seek inputs from external sources regarding alternative technologies, and integrate that advice with their internal knowledge (e.g., project specifications) to make an appropriate decision. External knowledge inputs can also be integrated with teams' internal expertise to reduce technological unpredictability in the later stages of uncertain projects.

Thus, it can be proposed that teams working on uncertain projects fulfill their knowledge scarcity by integrating knowledge from both internal and external sources. Therefore, it is hypothesized:

Hypothesis 5a: Project uncertainty positively influences a software team's internal knowledge integration.

Hypothesis 5b: Project uncertainty positively influences a software team's external knowledge integration.

Project Interdependence and Teams' Knowledge Integration

Project interdependence among software teams refers to the extent to which teams working on a common project need to share information, skills, and resources among themselves to successfully complete the project. For the purpose of this study, project interdependence is indicated by two components – task interdependence and outcome interdependence.

Both, task and outcome interdependence may have a negative influence on teams' internal knowledge integration. Previous research has reported that people working on tasks of high interdependence feel that they are not making a distinct contribution to the task (Wong et al. 1991). This lack of intrinsic motivation lowers their productivity and satisfaction, and makes them lose interest in activities that would typically improve team performance (such as internal knowledge integration) (Andres et al. 2001). The negative influence of project interdependence on internal knowledge integration may be compounded by the fact that increased external coordination and communication may not leave members of interdependent teams with enough time to develop a shared understanding critical for internal knowledge integration. For example, members of such teams may not be aware of their colleagues' unique expertise, skills, and abilities, which may hamper the teams' efforts to integrate these internal knowledge resources. This argument suggests:

Hypothesis 6a: Project interdependence negatively influences a software team's internal knowledge integration.

Interdependent projects require extensive inter-team coordination (Kazanjian et al. 2000; Loch et al. 1998). Coordination is required to avoid costly mistakes such as performing redundant activities, and to sequence, schedule, and synchronize interdependent tasks, as delays and mistakes in completing one task may jeopardize the completion of another (Andres et al. 2001). Increased coordination substantially increases communication between the teams (Struas et al. 1994). Previous research has observed that teams working on interdependent projects shared more information than independent teams (Jarvenpaa et al. 2000). Thus, through coordination, communication, and information-sharing, interdependent teams exchange knowledge inputs, which they integrate to perform project-related tasks. Additionally, the expertise and skills required for interdependent projects are usually distributed, and need to be integrated to achieve project outcomes. For example, software may be developed in the form of separate yet modules assigned to different teams (Zmud 1980). To avoid typical software errors that crop up at the interface of such modules, teams will need to create specialized knowledge to smoothly combine the modules into a coherent software system (Koushik et al. 1995; Kraut et al. 1995), and to create this systemic knowledge, teams may need to integrate module-specific knowledge from various teams. Thus, software teams working on interdependent projects may actively absorb knowledge inputs from each other, and integrate them to achieve project outcomes. Thus, it is predicted:

Hypothesis 6b: Project interdependence positively influences a software team's external knowledge integration.

IT-Usage

What influence does IT have on knowledge processes? This question has provoked an emerging body of research in the past few years. Among some of the early conceptual discussions on the topic, Alavi and Leidner proposed that "the application of information technologies can create an infrastructure and environment that contribute to organizational knowledge management by actualizing, supporting, augmenting, and reinforcing knowledge processes at a deep level through enhancing their underlying scope, timing, and overall synergy" (2001: 124). Five years later, the field is still coping with absence of insights into this proposition. Improvements in the status quo necessitate better understanding about the organizational processes through which IT-based systems influence knowledge processes such as knowledge integration (Sambamurthy et al. 2005), which is one of the objectives of this study.

To develop robust insights into the influence of IT on knowledge processes, this study examines the usage of two categories of IT-based systems - collaborative systems (e.g., corporate intranets, e-mail, telephone, list serves, and group support systems) (Hinds et al. 1995; Jarvenpaa et al. 2000; Sher et al. 2003) and KM systems (e.g., electronic knowledge repositories, expert directories, and electronic forum software) (Kankanhalli et al. 2005; Sher et al. 2003). This classification of IT-based systems is guided by Huber's distinction between communications and computing technologies (1984). He defines communications technology as including technological infrastructure for interpersonal information exchange, and computing technology as a combination of MIS, knowledge management systems, and DSS (Huber 1984; Lee et al. 1999/2000). Both, collaborative and KM systems are discussed in the next few sections.

Usage of Collaborative Systems

For the purpose of this study, the use of collaborative systems refers to "the use of IT -based systems to accomplish information activities such as accessing, searching, sharing, storing, and publishing information in a computer network within a person's work unit/department/organization (i.e., internal information activities) as well as external to the person's organization (i.e., external organization)" (Jarvenpaa et al. 2000: 130). Two key characteristics that differentiate among various collaborative systems and also define their respective importance towards knowledge integration are the bandwidth and synchrony of the medium (Hinds et al. 1995; Zmud et al. 1990). Bandwidth defines the ability to exchange information from multiple human senses, and synchrony is defined as the ability to allow two-way communication at the same time (Nohria et al. 1992). Thus, telephone has more bandwidth than e-mail, and it is also synchronous as compared to e-mail. A group support system (GSS) is synchronous and has less bandwidth than a telephone, but more bandwidth than e-mail.

Collaborative systems have been included in this study because individuals and teams use them to exchange knowledge (e.g., sharing ideas through e-mail or discussing project-related issues over telephone) (Jarvenpaa et al. 2000). Firms are increasingly investing in collaborative technologies to promote information and knowledge exchange within organizational units (e.g., teams) and across the organizational boundaries (Alavi et al. 1999; Fulk et al. 1995). Although their investment is based on the assumption that increasing the quantity of communication channels and improving computer-mediated collaboration among the team members would improve the chances of exchange and integration of knowledge (Kogut et al. 1992; Teigland et al. 2003), the results of implementing collaborative systems have been mixed. One stream of research suggest the benefits of collaborative systems. For example, it has been observed that as compared to the face-to-face groups, teams using collaborative systems are better on tasks such as brainstorming and decision-making (Maruping et al. 2004; Sambamurthy et al. 1993), and that collaborative systems enable team members to access knowledge beyond the team and even beyond the organizational setting (Teigland et al. 2003).

The second research stream proclaims that because of lack of bandwidth and synchrony, collaborative systems like e-mail and groupware are less effective in conveying ambiguous information as compared to a face-to-face meeting (Galeghar et al. 1994). For example, it was observed that teams using GSS were confused about the role of the system and thus were unable to use it to their benefit (Zigers et al. 1994).

Usage of KM Systems

The use of KM systems refers to the use of a class of “IT-based systems developed to support and enhance the organizational processes of knowledge creation, storage/retrieval, transfer, and application”(Alavi et al. 2001: 114). Previous research on KM systems has identified two models of such systems – the repository model and the network model (Bowman 2002). The repository model stresses on the codification and storage of knowledge to facilitate its reuse (Alavi 2000). A key technological application representing this model is an electronic knowledge repository (EKR), which includes searchable document databases along with the mechanisms for capture, storage, and publication of explicit knowledge (Kankanhalli et al. 2005). Previous studies have reported that people use EKRs to contribute (Jarvenpaa et al. 2000; Wasko et al. 2000) as well as to seek knowledge (Goodman et al. 1998).

The network model underlines facilitating interpersonal connections to improve the likelihood of knowledge exchange (Hansen et al. 1999). Expert directories and electronic forum software are two KM applications guided by this model. Expert directories allow people to search for knowledge sources beyond team boundaries (Sambamurthy et al. 2005), while electronic forum software allows members of communities of practice to interact among each other (Brown et al. 1991).

A typical KMS setup includes an enterprise knowledge portal front-end with an EKR back-end (Ryu et al. 2005). The other KM applications are also linked to the portal. By facilitating quick collection, storage, and exchange of knowledge on the organizational scale, a well-developed KM system helps integrate fragmented stocks and flows of knowledge at the individual, team, and organizational level (Gold et al. 2001; Lee et al. 2003).

Benefits of Using IT-based systems

Previous research has proposed that using IT-based systems accrues two types of benefits. The first category of benefits include the efficiencies related to information aggregation, which are gained by using IT-based systems to reduce the number of communication contacts between various entities (e.g., teams) (Schultze et al. 2004). In context of this study, efficiencies related to information aggregation will influence software teams' knowledge integration by (1) improving the availability of internal and external knowledge inputs that can be integrated, thereby (2) improving the quality of inputs integrated, and (3) decreasing teams' cost (time and effort) of integration (Malone et al. 1987). Teams can accrue these benefits by effectively using the collaborative systems (specifically list serves and e-mail) and KM systems (specifically EKR and expert directories) to streamline their communication and knowledge search process within and outside the team.

Software teams accrue the second type of benefits of IT-based systems when team members use both collaborative and KM systems to jointly interpret and assimilate various internal and external knowledge resources (Malone et al. 1987; Schultze et al.

2004). For example, this effect will be exemplified when team members use telephone, e-mail, GSS, or electronic discussion forums to develop a joint understanding of various project-related issues, and to simultaneously integrate their respective knowledge inputs to develop project-level knowledge.

This study uses these two categories of benefits of IT-based systems to discuss how usage of collaborative systems and KM systems moderates the relationships between the team- and project-related antecedents and teams' integration of internal as well as external knowledge.

Hypotheses Predicting Moderating Influence of IT Usage

In the ensuing sections, hypotheses predicting the moderating influence of IT usage are developed.

IT Usage, Teams' Knowledge Heterogeneity, and Teams' Knowledge Integration

Heterogeneous teams can gain multiple benefits by using IT-based systems for the purpose of internal knowledge integration. Earlier it was proposed that heterogeneous teams are more likely to develop an initial structure for future integration of their knowledge resources. This may require interactive and intensive communication, and collaborative systems may aid this process by streamlining intra-team communications. Collaborative systems also improve the within-team information sharing, thus supporting teams' efforts to create a shared context among the members. Teams can buttress their efforts by using KM systems to interpret and assimilate internal and external knowledge inputs (Schultze et al. 2004). Once the shared context has been created and a structure for knowledge integration has been developed, IT-based systems can still be used by members to submit relevant inputs to teams' internal knowledge integration (Smith et al. 2005).

Regarding the influence of IT-usage on teams' external knowledge integration, the earlier discussion proposed that members of heterogeneous teams have interpersonal networks in diverse domains, which they might utilize as sources of external knowledge. Members can use IT-based systems to improve their efficiencies of knowledge aggregation from these sources. For example, members can use collaborative systems like e-mail and telephone, and KM systems like electronic forums to improve their access to external networks (Malone et al. 1987).

Furthermore, it was proposed earlier that as compared to homogeneous teams, heterogeneous teams have more diverse expertise, and thus have a higher capacity to integrate external knowledge in multiple domains (Cohen et al. 1990). Experts in such teams can use IT-based systems not just to collaborate and combine their expertise but also to absorb more appropriate knowledge inputs from external sources and integrate them with their combined expertise to develop a more robust body of project-level knowledge.

Thus, in light of the above discussion, it can be proposed that IT-usage in heterogeneous teams will further improve teams' access to external knowledge resources in multiple domains, and increase teams' capacity to absorb and integrate these knowledge resources. Additionally, IT-usage will enhance internal collaboration among teams' experts thereby improving their ability to combine diverse knowledge assets into project-level knowledge. It is thus proposed:

Hypothesis 7a: IT-usage moderates the influence of a software team's knowledge heterogeneity on its internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.

Hypothesis 7b: IT-usage moderates the influence of a software team's knowledge heterogeneity on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

IT Usage, Teams' Relational Capital, and Teams' Knowledge Integration

The earlier discussion on teams' relational capital proposes that mutual trust among the team members alleviates suspicion among them about the misuse of their uniquely held knowledge (Davenport et al. 1998). This creates a climate conducive to internal knowledge exchange, which reinforces teams' efforts to integrate the uniquely held knowledge resources of their members (Nahapiet 1996). Teams' usage of collaborative systems may help develop a knowledge sharing climate, while the usage of KM systems may facilitate members' efforts to integrate their knowledge with teams' other knowledge resources (Nahapiet et al. 1998).

On a separate note, members of teams with high levels of relational capital will also share close working relationships, which may help develop a better understanding among the members about project's knowledge requirements and how best to use teams' internal and external knowledge resources to fulfill those requirements (Ko et al. 2005). Teams can use IT-based systems to further improve their awareness of internal and external knowledge resources, and to improve the subsequent integration of these resources.

Mutually trusting team members who share close working relationships also tend to reciprocate each other’s knowledge sharing behavior (Kale et al. 2000), thereby facilitating team’s efforts to integrate its internal knowledge resources. Usage of IT-based systems may enable the team members to reciprocate more effectively. Reciprocity may also motivate them to absorb knowledge inputs requested by their colleagues from external sources, and they can use IT-based systems to search for, and acquire, better quality external knowledge inputs.

In light of the above discussion, the moderating influence of IT-usage on the relationship between a software team’s relational capital and its internal and external knowledge integration is hypothesized as follows:

Hypothesis 8a: IT-usage moderates the influence of a software team’s relational capital on its internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.

Hypothesis 8b: IT-usage moderates the influence of a software team’s relational capital on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

IT Usage, Boundary-Buffering Processes, and Teams’ Knowledge Integration

Teams performing sentry and guard processes may discourage their members to share knowledge with external sources, which may depress teams’ external knowledge integration. Using IT-based systems may compensate this situation, for example, by allowing individual team members to acquire knowledge inputs from external sources otherwise inaccessible because of the sentry and guard processes. Therefore, it is hypothesized:

Hypothesis 9: IT-usage moderates the influence of a software team's sentry processes on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

Hypothesis 10: IT-usage moderates the influence of a software team's guard processes on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

IT Usage, Project Uncertainty, and Teams' Knowledge Integration

Teams regularly need to absorb requirements-related inputs from the users (Nidumolu 1996; Van de Ven et al. 1976). These external inputs need to be integrated with teams' internal knowledge resources to reduce requirements-related uncertainty (Nidumolu 1995). In the absence of frequent face-to-face interactions with the users, teams' may coordinate with them through collaborative systems such as telephone and e-mail (Lee et al. 1999/2000). Telephone is a synchronous medium with high bandwidth, and can facilitate interactive communication, and e-mail can be used to exchange large amounts of factual content. Both systems may improve teams' communication with the users, and help them better integrate knowledge inputs obtained from the users. Therefore, teams working on uncertain projects may facilitate their coordination efforts to integrate external knowledge inputs by using IT-based systems.

Teams working on uncertain projects also need to initiate informal horizontal communication with external sources such as members' interpersonal networks (Andres et al. 2001; Kraut et al. 1992). Teams may absorb knowledge inputs from these networks (Hoegl et al. 2004), and integrate them with the internally available knowledge to reduce technological unpredictability. Using both collaborative and KM systems can facilitate this process. For example, team members can use KM systems like EKR (to search for documented knowledge), expert directories (to search for external experts in the field), and electronic discussion forums (to seek feedback in their communities of practice). Members can then use collaborative systems like e-mail and telephone to share the external knowledge inputs absorbed with the help of the KM systems, thus facilitating the team level assimilation of those inputs. On a separate note, collaborative systems like GSS can also enable the team members to develop and share a common perspective towards project uncertainty, and its possible reasons, thereby improving team's ability to cope with it (Lee et al. 1999/2000). Therefore, it is hypothesized:

Hypothesis 11a: IT-usage moderates the influence of project uncertainty on a software team's internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.

Hypothesis 11b: IT-usage moderates the influence of project uncertainty on a software team's external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

IT Usage, Project Interdependence, and Teams' Knowledge Integration

In the earlier sections, it has been proposed that that increased external coordination and communication among interdependent teams may not leave team members with enough time to develop a shared understanding, which may leave them unaware of their colleagues' unique expertise, skills, and abilities. This may hamper the teams' efforts to integrate these internal knowledge resources. This argument suggests a negative relationship between project interdependence and a participating team’s internal knowledge integration.

Teams’ IT-usage can offset this relationship. Although using IT-based systems may not be a good option to develop within-team shared understanding, team members can use the systems to at least gain communication efficiencies. In the absence of enough time for person-to-person interactions, members can use collaborative systems to develop a preliminary level of mutual awareness about the teams’ internal knowledge resources. This may improve the likelihood of integrating these resources for project purposes. Thus, it is hypothesized:

<u>Hypothesis 12a:</u> IT-usage moderates the influence of project interdependence on a software team’s internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.

In the earlier sections, it has also been proposed that interdependent teams require extensive coordination to sequence, schedule, and synchronize their respective tasks. Close coordination and frequent communication is also required to integrate the technical and functional expertise distributed across multiple teams. Interdependent teams have a choice of two types of coordination and communication mechanisms. They can use human-intensive mechanisms such as cross-unit groups and direct contacts among project managers (Brown 1999). But these mechanisms have limited coordination capabilities (Tanriverdi 2005). Alternatively, teams can use IT-based systems, which, according to task-technology fit theory, are better cross-unit coordination and communication mechanisms for teams working on interdependent projects (Goodhue et al. 1995). Using IT-based systems in interdependent projects reduces the cognitive information processing costs of participating teams (Jarvenpaa et al. 2000). This argument is closely related to the discussion in earlier sections about the benefits of using IT-based systems. As person-to-person interactions are difficult among members of interdependent teams, they increasingly rely on collaborative systems for exchanging knowledge inputs. Collaborative systems improve the efficiencies of interdependent teams, gained by reducing the time and effort spent to communicate with each other (Schultze et al. 2004). Additionally, KM systems such as electronic forums can be utilized to share knowledge inputs with each other, thereby facilitating external knowledge integration among interdependent teams (Kankanhalli et al. 2001). Therefore, it can be argued that the interdependent teams' usage of IT-based systems will positively influence their external knowledge integration efforts. In view of these arguments, it is predicted:

Hypothesis 12b: IT-usage moderates the influence of project interdependence on a software team's external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

All research hypotheses are summarized in Table 1.

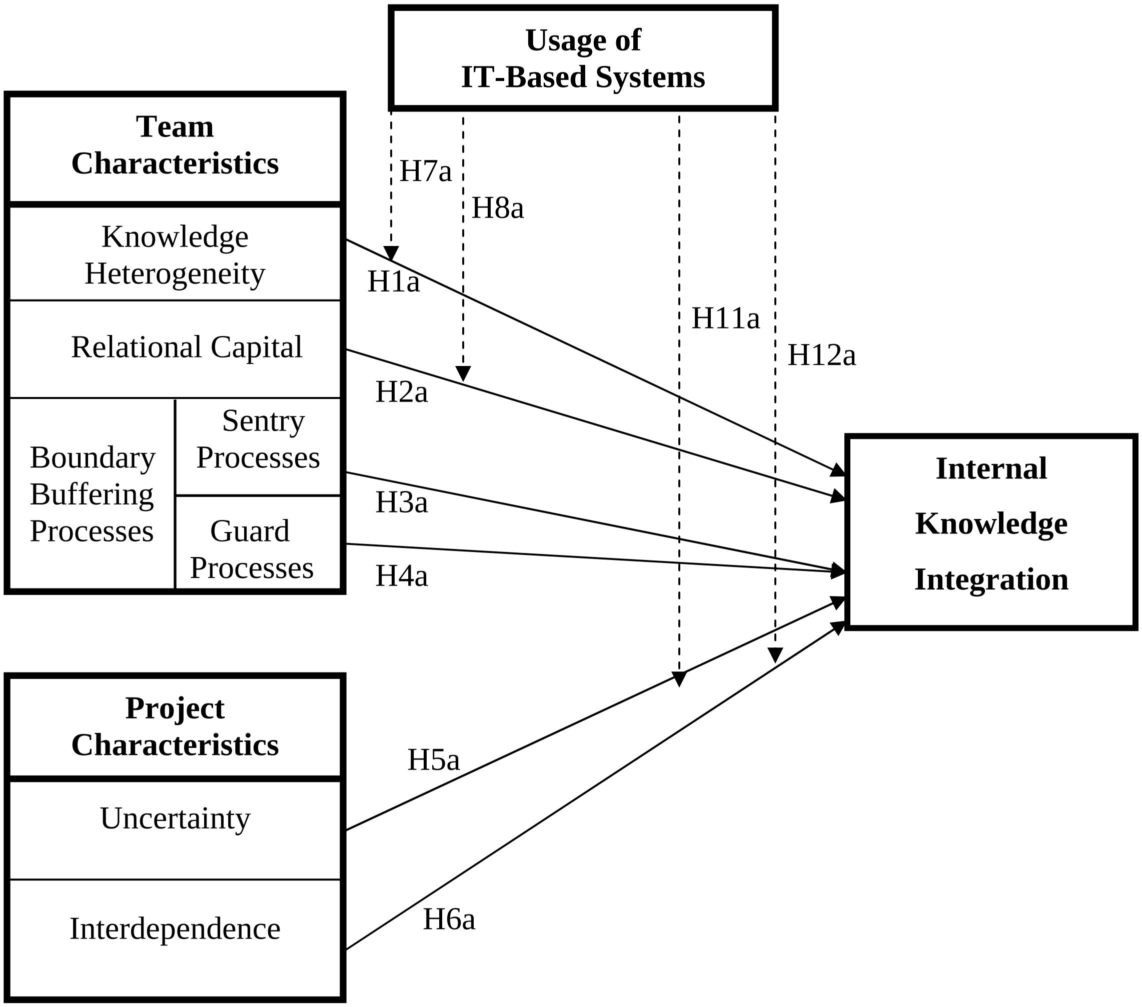


Figure 2.1 Research model indicating proposed hypotheses relating the independent and moderating variables to internal knowledge integration

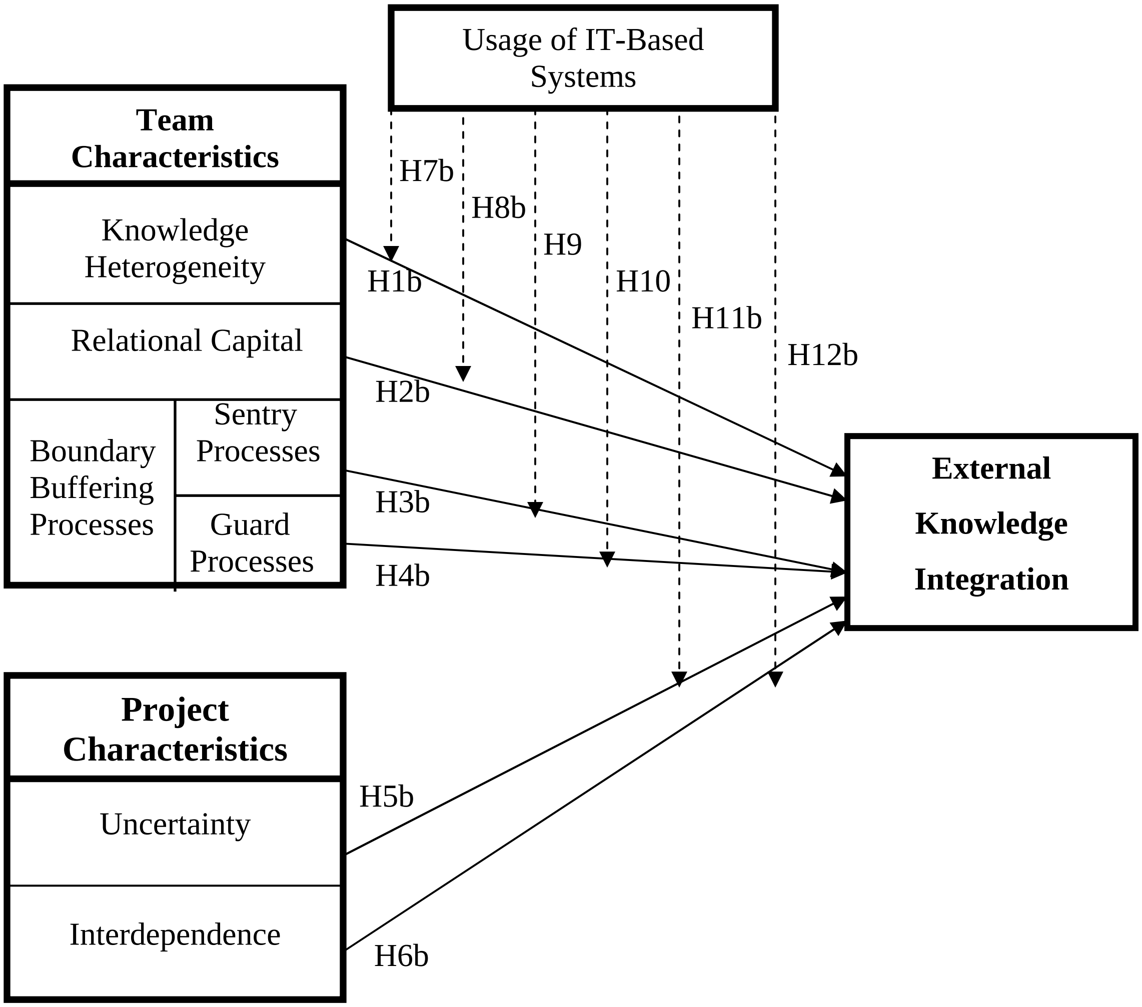


Figure 2.2 Research model indicating proposed hypotheses relating the independent and moderating variables to external knowledge integration

Table 2.1: Research Hypotheses

H1a	A software team’s knowledge heterogeneity positively influences its internal knowledge integration.
H1b	A software team’s knowledge heterogeneity positively influences its external knowledge integration.
H2a	A software team’s relational capital positively influences its internal knowledge integration.
H2b	A software team’s relational capital positively influences its external knowledge integration.
H3a	A software team’s sentry processes positively influence its internal knowledge integration.
H3b	A software team’s sentry processes negatively influence its external knowledge integration.
H4a	A software team’s guard processes positively influence its internal knowledge integration.
H4b	A software team’s guard processes negatively influence its external knowledge integration.
H5a	Project uncertainty positively influences a software team’s internal knowledge integration.
H5b	Project uncertainty positively influences a software team’s external knowledge integration.
H6a	Project interdependence negatively influences a software team’s internal knowledge integration.
H6b	Project interdependence positively influences a software team’s external knowledge integration.
H7a	IT-usage moderates the influence of a software team’s knowledge heterogeneity on its internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.
H7b	IT-usage moderates the influence of a software team’s knowledge heterogeneity on its external knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.
H8a	IT-usage moderates the influence of a software team’s relational capital on its internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.
H8b	IT-usage moderates the influence of a software team’s relational capital on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.
H9	IT-usage moderates the influence of a software team’s sentry processes on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.
H10	IT-usage moderates the influence of a software team’s guard processes on its external knowledge integration: External knowledge integration improves at higher levels of IT-usage.
H11a	IT-usage moderates the influence of project uncertainty on a software team’s internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.
H11b	IT-usage moderates the influence of project uncertainty on a software team’s external knowledge integration: External knowledge integration improves at higher levels of IT-usage.
H12a	IT-usage moderates the influence of project interdependence on a software team’s internal knowledge integration: Internal knowledge integration improves at higher levels of IT-usage.
H12b	IT-usage moderates the influence of project interdependence on a software team’s external knowledge integration: External knowledge integration improves at higher levels of IT-usage.

CHAPTER 3: METHOD

Context

The discussion of research hypotheses in the previous chapter identifies two key themes for this study. First, to examine how software teams' internal and external knowledge integration are influenced by team-related antecedents (knowledge heterogeneity, relational capital, and boundary-buffering processes) and project-related antecedents (project uncertainty and interdependence). And second, to examine how IT-usage (specifically usage of collaborative systems and KM systems) moderates these influences.

Developing the Measurement Instruments

To enhance validity, the constructs were measured using existing scales where available (Stone 1978). Where existing scales were absent, new items were developed from the previous literature. A total of 62 items were thus accumulated for various constructs and sub-constructs. They are discussed briefly in the following sections.

Items to Measure Teams' Knowledge Integration

Two sets of questions related to knowledge integration are included in this study. One set relates to the team's internal knowledge integration (IKI), and the other concerns its external knowledge integration (EKI). A team's internal knowledge integration refers to the synthesis and application of individually-held knowledge into team-level systemic knowledge to accomplish project objectives (Alavi et al. 2002; Tiwana et al. Forthcoming). The six items for IKI were modified from the previous studies in the area, as well as in the areas of knowledge transfer and organizational learning (Mukherji 2002; Tempelton et al. 2002; Tiwana et al. 2003; Tiwana et al. Forthcoming). A high score on internal knowledge integration indicates that the team vigorously combines and assimilates its internal knowledge resources to create systemic project-level knowledge. An example of the items measuring team's internal knowledge integration is "Team members combined their individual perspectives to develop a shared understanding of the project objectives."

External knowledge integration refers to the extent to which the teams absorb knowledge from external sources and integrate it with internally available knowledge to bear on project outcomes (Tiwana et al. 2003). A high score on external knowledge integration indicates the team actively utilizes knowledge from external sources. Previous studies in knowledge acquisition, knowledge transfer, and organizational learning contributed six items for the EKI scale (Ko et al. 2005; Norman 2004; Tempelton et al. 2002). A sample item for external knowledge integration is "If the required knowledge was not available within the team, members used knowledge acquired from external sources.

Items to Measure Teams' Knowledge Heterogeneity

Previous studies on team's knowledge heterogeneity (KH) have examined the construct in light of two factors – the diversity of functional backgrounds of team members, and the diversity of their expertise and skill sets (Campion et al. 1993;

Campion et al. 1996). These factors fit with the nature of knowledge heterogeneity in software teams, thus a three-item scale pertaining to these factors was utilized in this study to measure knowledge heterogeneity. A sample item is “Members of the team had a variety of different background and experiences.”

Items to Measure Teams’ Relational Capital

Kale, Singh, and Perlmutter (2000) identified level of mutual trust among the team members, closeness of their working relationships, and their level of reciprocal behavior as three indicators of team’s relational capital (RC). In their study, they used a five-item scale pertaining to these indicators. A high score on mutual trust indicates an absence of mutual suspicion of opportunistic behavior among the team members. A sample of mutual trust items is “The team was characterized by mutual trust among members at multiple levels.”

A high score on closeness of working relationships indicates a higher quality of interactions and better friendships among the team members. A sample item for closeness of relationships is “There was a closer, personal interaction among members of this team at multiple levels.”

A high score on reciprocal behavior indicates that team members are more open towards responding positively to each other’s acts of sharing. The item measuring team’s reciprocity is “The team was characterized by high level of reciprocal behavior among members at multiple levels.”

Items to Measure Teams’ Boundary-Buffering Processes

Boundary-buffering processes refer to the external processes teams perform to protect their operations and resources from external disturbances. Two types of boundary-buffering processes are examined in this study. Sentry processes (SP) help teams monitor and control the inflow of information and resources from external entities. A high score on sentry processes indicates that the team avoids accepting undesired external inputs, and filters the desired ones in terms of what portion to accept, and whom to send it to (Ancona et al. 1988; Jemison 1984). A five-item scale was developed from previous research in the field. A sample item for sentry processes is “The team actively monitored information coming from external sources such as other teams, individuals, and departments.”

Teams perform guard processes (GP) to evaluate the external requests for information and resources. A high score on guard processes indicates that the team actively decides which external requests to deny and which ones to fulfill. A five-item scale was developed from previous research on guard processes (Ancona et al. 1992a; Jemison 1984). A sample item is “The team avoided releasing information to others in the company.”

Items to Measure Teams’ IT-Usage

For the purpose of this study, IT-usage refers to the use of collaborative and knowledge management (KM) systems. While collaborative systems are primarily used to facilitate information transfer and knowledge exchange, KM systems assist in the creation, storage/retrieval, and application of knowledge.

Two sets of eight items each were developed for assessing the *nature of IT-usage* (ITU) and the *frequency of IT-usage* (ITF). Example of an ITU item is “Team members used collaborative systems to coordinate project-related tasks among each other.”

The ITF items focus on a comparative assessment of how often the team uses IT-based systems for: (1) internal coordination and communication; (2) collaboration with external entities; (3) creation, search, and retrieval of knowledge. Items were developed primarily from previous studies (Gold et al. 2001; Kankanhalli et al. 2005; Mukherji 2002; Sher et al. 2003). Sample items include “Compared with other teams you have led, the team usesd IT-based systems more to internally coordinate project-related tasks,” and “Compared with other teams you have led, the team used IT-based systems more to retrieve project-related knowledge (e.g., by downloading relevant documents).”

Items to Measure Project Uncertainty

Project uncertainty in software projects is related to the lack of critical knowledge inputs regarding project requirements, project technologies, and project outcomes (Beath 1983; Nidumolu 1995; Nidumolu 1996a). Based on previous studies, three sets of items were developed to assess uncertainty emanating from each of the three areas mentioned above. Requirements uncertainty (RU) was measured in terms of the instability and diversity of project requirements (Nidumolu 1995). The RU scale had 3 items. A sample item includes “Compared to other projects you have worked on, requirements for that project fluctuated quite a bit.” Technological uncertainty (TU) was assessed in terms of the uncertainty in deciding appropriate technologies in the beginning of the project, and the extent to which unexpected or novel technological problems occur in the later stages of the project (Nidumolu 1995). The TU scale had 5 items. A sample items is “Compared to other projects you have worked on, that project had more unpredictable problems related to software platforms.” Finally, outcome uncertainty (OU) was measured in terms of the unpredictability of project outcomes (Van de Ven et al. 1976). The single item OU scale included “Compared to other projects you have worked on, the outcomes of that project were more unpredictable.”

Items to Measure Project Interdependence

Task interdependence (TI) and outcome interdependence (OI) are the two sub-constructs for project interdependence. To measure these two sub-constructs, a seven-item scale was developed from previous studies (Andres et al. 2001; Campion et al. 1993; Pearce et al. 1991). A sample TI item is “Your team had to complete programming tasks that were utilized by other teams to complete the project.” A sample OI item is “Your team’s progress on the project was very much dependent on the progress of other teams.”

Preliminary Test

Before embarking on the data collection exercise, the 62 items were subjected to a conceptual validation exercise based on recommendations by Moore and Benbasat (1991). Four sets of 62 items, printed on separate cards, were prepared. Each set was mixed up and given to an IS doctoral student. The students were also provided names and definitions of the constructs, and were asked to sort the items by assigning them to various construct categories or an “other” (no fit) category. This process helped identify items that were ambiguously worded or did not fit with other questions.

The four sorters correctly assigned 95.2 percent of items to intended constructs (see Table 3.1). The inter-rater reliability was 0.98. Based on the feedback from this exercise, ten items were dropped. These included two items each for internal knowledge integration (IKI3 and IKI6), guard processes (GP4 and GP5), and nature of IT-usage (ITU6 and ITU8); and one item each for sentry processes (SP5), IT-usage frequency (ITF2), task uncertainty (TU5), and task interdependence (TI1).

Table 3.1: Results of Conceptual Validation Exercise

Target Category	Actual Category														Total Questions	Hit Rate (%)
	IKI	EKI	SP	GP	KH	RC	ITU	ITF	RU	TU	OU	TI	OI	Other		
IKI	22					1								1	24	92.86
EKI		24													24	100
SP		1	18	1											20	90
GP		1	4	15											20	75
KH					12										12	100
RC						20									20	100
ITU							30	1						1	32	93.75
ITF							1	31							32	96.87
RU									12						12	100
TU									1	19					20	95
OU											4				4	100
TI												15	1		16	93.75
OI													12		12	100
Average																95.17

In the next stage, a group of four IS faculty members were requested to verify the 52 remaining items and their grouping to measure each construct. Based on their feedback, and to keep only 3-4 items per construct, 17 items were deleted for various constructs. The deleted items included three items each for external knowledge integration (EKI3, EKI5, EKI6), nature of IT-usage (ITU4, ITU5, ITU7) and IT-usage frequency (ITF6, ITF7, ITF8); two items each for relational capital (RC1 & RC2) and outcome interdependence (OI2 & OI3); and one item each for sentry processes (SP3), guard processes (GP3), technological uncertainty (TU1), and task interdependence (TI4). Additionally, one new item was added for outcome uncertainty (OU2) and four items were reworded, which included two questions each for guard processes (GP1 and GP2) and task interdependence (TI2 and TI3).

A preliminary test of the resulting 36 items was then conducted to further examine their content validity, construct validity, and reliability. A questionnaire was designed to collect data in light of the most successful project as well as the least successful project in the experience of a project leader. To do that, the questionnaire was divided in two sections (Section 1 and Section 2). Section 1 was titled “Most Successful Project” and section 2 was titled “Least Successful Project.” Both sections contained the same set of 36 items.

Scales included 7-point Likert anchors ranging from <u>strongly disagree</u> (=1) to <u>strongly agree</u> (=7). The questionnaire was administered to 50 project leaders in a CMM Level 5 software company. A total of 38 responses out of 50 surveys administered resulted in a response rate of 76%. 2 responses were invalid, as they were not completed properly. Discarding these responses left 36 useable surveys. 85.8% respondents were male and 14.2% were female. Respondents had an average of 7.3 years of experience in the software industry. Table 3.2 summarizes various project demographics for both “most successful” and “least successful” projects.

Table 3.2. Project Demographics for Preliminary Test

Project Demographics		**Most Successful Projects**	**Least Successful Projects**
Average Team Size		13	15
Project Duration		8.3 months	8.8 months
Project Type	Developing Customized Solution	86.12%	13.88%
	Product Development	75%	25%

To test construct validity, exploratory factor analysis was conducted on the pre-test data using Principal Components Analysis (PCA) extraction method with Varimax rotation. Reliability was calculated for each group of items using Cronbach’s alpha coefficient (Cronbach 1951). The following paragraphs present the preliminary test results. They are discussed separately for section 1 (most successful project) and section 2 (least successful project).

Knowledge Integration Scales

The eigenvalue results shown in Tables 3.3a & 3.3b propose a two-factor solution for both sections (here onwards discussed as section 1 and section 2 respectively). For section 1, the eigenvalues are 4.728 and 1.742, and for section 2, the eigenvalues are 4.327 and 1.244 respectively. The two-factor solution explains 71.887 percent of the total variance for section 1 and 79.579 percent variance for section 2.

Table 3.3a: Eigenvalues for Knowledge Integration Instrument (Section 1)

Total Variance Explained

Component	Initial Eigenvalues			Extraction Sums of Squared Loadings		
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	4.728	52.530	52.530	4.728	52.530	52.530
2	1.742	19.357	71.887	1.742	19.357	71.887
3	.670	7.446	79.333			
4	.611	6.786	86.119			
5	.600	6.672	92.790			
6	.272	3.019	95.810			
7	.173	1.921	97.731			
8	.139	1.543	99.274			
9	6.534E-02	.726	100.000			

Extraction Method: Principal Component Analysis.

Table 3.3b: Eigenvalues for Knowledge Integration Instrument (Section 2)

Total Variance Explained

Component	Initial Eigenvalues			Extraction Sums of Squared Loadings		
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	4.327	61.813	61.813	4.327	61.813	61.813
2	1.244	17.766	79.579	1.244	17.766	79.579
3	.731	10.444	90.023			
4	.349	4.989	95.012			
5	.188	2.683	97.695			
6	9.522E-02	1.360	99.055			
7	6.613E-02	.945	100.000			

Extraction Method: Principal Component Analysis.

The two factors were rotated, resulting in clean loadings for their respective indicators in each section (Tables 3.4a & 3.4b). For section 1, the four IKI indicators loaded together with loadings of .798, .767, .825, and .819, while the three EKI indicators loaded together with loadings of .881, .946, and .915. There were no cross-loadings.

Table 3.4a: Rotated Factor Loadings of Knowledge Integration Indicators (Section 1)

Rotated Component Matrix[a]

	Component	
	1	2
IKI1	.798	.165
IKI2	.767	.278
IKI4	.825	.186
IKI5	.819	.156
EKI1	.293	.881
EKI2	.166	.946
EKI4	.206	.915

Extraction Method: Principal Component Analy
Rotation Method: Varimax with Kaiser Normali

a. Rotation converged in 3 iterations.

For section 2, the four IKI indicators loaded together with loadings of .924, .893, .716, and .856, while the three EKI indicators loaded together with loadings of .858, .826, and .822. There were no cross-loadings.

Table 3.4b: Rotated Factor Loadings of Knowledge Integration Indicators (Section 2)

Rotated Component Matrix[a]

	Component	
	1	2
IKI1	.924	.144
IKI2	.893	.188
IKI4	.716	.467
IKI5	.856	.307
EKI1	.179	.858
EKI2	.389	.826
EKI4	.162	.822

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 3 iterations.

Reliability was then calculated for each group of indicators for both the sections. Reliability measures the consistency among indicators for a given construct. An alpha coefficient of .60 or above is considered acceptable in social science research (Nunnally 1967; Robinson et al. 1991). Table 3.6 presents the reliabilities for both the sections.

Table 3.5: Reliabilities for Knowledge Integration Scales

Construct	**Section 1**	**Section 2**
Internal Knowledge Integration	.8436	.9137
External Knowledge Integration	.9236	.8473

Team Characteristics Scales (Knowledge Heterogeneity and Relational Capital)

The eigenvalue results shown in Tables 3.6a & 3.6b propose a two-factor solution for the team characteristics instrument for both sections. For section 1, the eigenvalues are 2.642 and 2.151, and for section 2, the eigenvalues are 2.622 and 1.849 respectively. The two-factor solution explains 79.890 percent of the total variance for section 1 and 74.519 percent variance for section 2.

Table 3.6a: Eigenvalues for Team Characteristics Instrument (Section 1)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.642	44.040	44.040	2.642	44.040	44.040
2	2.151	35.850	79.890	2.151	35.850	79.890
3	.608	10.132	90.021			
4	.268	4.467	94.488			
5	.250	4.171	98.659			
6	8.045E-02	1.341	100.000			

Extraction Method: Principal Component Analysis.

Table 3.6b: Eigenvalues for Team Characteristics Instrument (Section 2)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.622	43.699	43.699	2.622	43.699	43.699
2	1.849	30.820	74.519	1.849	30.820	74.519
3	.494	8.228	82.747			
4	.441	7.347	90.094			
5	.340	5.666	95.760			
6	.254	4.240	100.000			

Extraction Method: Principal Component Analysis.

The two factors had clean loadings for their respective indicators in each section (Tables 3.7a & 3.7b). For section 1, the three knowledge heterogeneity indicators had loadings of .897, .917, and .882, while the three relational capital indicators had loadings of .906, .894, and .745.

Table 3.7a: Rotated Factor Loadings of Team Characteristics Indicators (Section 1)

Rotated Component Matrix[a]

	Component	
	1	2
KH1	.897	-4.79E-02
KH2	.917	7.608E-02
KH3	.882	.140
RC3	7.672E-02	.906
RC4	-7.85E-02	.894
RC5	.148	.745

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

For section 2, the three knowledge heterogeneity indicators had loadings of .883, .893, and .766, while the three relational capital indicators had loadings of .850, .771, and .880. There were no cross-loadings.

Table 3.7b: Rotated Factor Loadings of Team Characteristics Indicators (Section 2)

Rotated Component Matrix[a]

	Component	
	1	2
KH1	.883	-.172
KH2	.893	.135
KH3	.766	.310
RC3	8.904E-02	.850
RC4	.238	.771
RC5	-8.97E-02	.880

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

Reliability was then calculated for each group of indicators for both the sections. Table 3.8 presents the reliabilities for both the sections.

Table 3.8: Reliabilities for Team Characteristics Scales

Construct	**Section 1**	**Section 2**
Knowledge Heterogeneity	.8773	.8063
Relational Capital	.8069	.7932

Boundary-Buffering Processes Scales

The eigenvalue results shown in Tables 3.9a & 3.9b propose a two-factor solution for the team boundary-buffering processes instrument for both sections. For section 1, the eigenvalues are 2.343 and 1.541, and for section 2, the eigenvalues are 2.896 and 1.182 respectively. The two-factor solution explains 77.68 percent of the total variance for section 1 and 81.57 percent variance for section 2.

Table 3.9a: Eigenvalues for Boundary-Buffering Processes Instrument (Section 1)

Total Variance Explained

	Initial Eigenvalues			Rotation Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.343	46.859	46.859	2.319	46.386	46.386
2	1.541	30.824	77.682	1.565	31.296	77.682
3	.589	11.784	89.466			
4	.318	6.368	95.834			
5	.208	4.166	100.000			

Extraction Method: Principal Component Analysis.

Table 3.9b: Eigenvalues for Team External Processes Instrument (Section 2)

Total Variance Explained

	Initial Eigenvalues			Rotation Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.896	57.924	57.924	2.581	51.630	51.630
2	1.182	23.643	81.568	1.497	29.938	81.568
3	.499	9.985	91.552			
4	.294	5.876	97.428			
5	.129	2.572	100.000			

Extraction Method: Principal Component Analysis.

The two factors were rotated, resulting in clean loadings for their respective indicators in each section (Tables 3.10a & 3.10b). For section 1, the three indicators for sentry processes had loadings of .843, .910, and .881, while the two indicators for guard processes had loadings of .886, .872.

Table 3.10a: Rotated Factor Loadings of Team External Processes Indicators (Section 1)

Rotated Component Matrix[a]

	Component	
	1	2
SP1	.843	-4.54E-02
SP2	.910	-.112
SP3	.881	6.511E-02
GP1	-6.16E-02	.886
GP2	2.179E-03	.872

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

For section 2, the three indicators for sentry processes had loadings of .924, .923, and .889, while the two indicators for guard processes had loadings of .893 and .791. There were no cross-loadings.

Table 3.10b: Rotated Factor Loadings of Team External Processes Indicators (Section 2)

Rotated Component Matrix[a]

	Component	
	1	2
SP1	.924	.175
SP2	.923	.184
SP3	.889	9.191E-02
GP1	2.112E-02	.893
GP2	.292	.791

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

Reliability was then calculated for each group of indicators for both the sections. Table 3.11 presents the reliabilities for both the sections.

Table 3.11: Reliabilities for Team External Processes Scales

Construct	**Section 1**	**Section 2**
Sentry Processes	.8498	.9193
Guard Processes	.8169	.7303

IT-Usage Scales

Team IT usage instrument had 5 indicators each for measuring nature of IT-usage and IT-usage frequency. The eigenvalue results shown in Tables 3.12a & 3.12b propose a two-factor solution for this instrument for both sections. For section 1, the eigenvalues are 4.244 and 1.098, and for section 2, the eigenvalues are 4.884 and 1.180 respectively. The two-factor solution explains 76.306 percent of the total variance for section 1 and 86.631 percent variance for section 2.

Table 3.12a: Eigenvalues for Team IT Usage Instrument (Section 1)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	4.244	60.622	60.622	4.244	60.622	60.622
2	1.098	15.684	76.306	1.098	15.684	76.306
3	.662	9.464	85.769			
4	.327	4.669	90.439			
5	.299	4.268	94.707			
6	.207	2.960	97.666			
7	.163	2.334	100.000			

Extraction Method: Principal Component Analysis.

Table 3.12b: Eigenvalues for Team IT Usage Instrument (Section 2)

Total Variance Explained

Component	Initial Eigenvalues			Extraction Sums of Squared Loadings		
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	4.884	69.772	69.772	4.884	69.772	69.772
2	1.180	16.860	86.631	1.180	16.860	86.631
3	.443	6.329	92.961			
4	.265	3.780	96.741			
5	.106	1.510	98.251			
6	7.569E-02	1.081	99.332			
7	4.676E-02	.668	100.000			

Extraction Method: Principal Component Analysis.

The two factors were rotated, resulting in clean loadings for their respective indicators in each section (Tables 3.13a & 3.13b). For section 1, the three indicators for ITU had loadings of .833, .909, and .696, while the four indicators for ITF had loadings of .675, .856, .860, and .874. There were no cross-loadings.

Table 3.13a: Rotated Factor Loadings of IT-Usage Indicators (Section 1)

Rotated Component Matrix[a]

	Component	
	1	2
ITU1	.321	.833
ITU2	.107	.909
ITU3	.493	.696
ITF1	.675	.293
ITF3	.856	.347
ITF4	.860	.260
ITF5	.874	.115

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

For section 2, the three indicators for ITU had loadings of .841, .914, and .818, while the four indicators for ITF had loadings of .856, .918, .903, and .920. There were no cross-loadings.

Table 3.13b: Rotated Factor Loadings of IT-Usage Indicators (Section 2)

Rotated Component Matrix[a]

	Component	
	1	2
ITU1	.278	.841
ITU2	.201	.914
ITU3	.399	.818
ITF1	.856	.256
ITF3	.918	.288
ITF4	.903	.318
ITF5	.920	.296

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 3 iterations.

Reliability was then calculated for each group of indicators for both the sections. Table 3.14 presents the reliabilities for both the sections.

Table 3.14: Reliabilities for IT-Usage Scales

Construct	**Section 1**	**Section 2**
IT Usage	.8192	.8901
IT Usage Frequency	.8769	.9605

Project Uncertainty Scales

Project uncertainty instrument had 8 indicators for measuring three sub-constructs – requirements uncertainty, outcome uncertainty, and technological uncertainty. The eigenvalue results shown in Tables 3.15a & 3.15b propose a three-factor solution for both sections. The eigenvalues for section 1 are 3.055, 1.692, and 1.175, while those for section 2 are 2.890, 1.713, and 1.178. The three-factor solution explains 74.035 percent of total variance for section 1 and 72.271 percent of variance for section 2.

Table 3.15a: Eigenvalues for Project Uncertainty Instrument (Section 1)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	3.055	38.190	38.190	3.055	38.190	38.190
2	1.692	21.152	59.342	1.692	21.152	59.342
3	1.175	14.693	74.035	1.175	14.693	74.035
4	.644	8.047	82.082			
5	.632	7.903	89.985			
6	.352	4.406	94.391			
7	.305	3.808	98.199			
8	.144	1.801	100.000			

Extraction Method: Principal Component Analysis.

Table 3.15b: Eigenvalues for Project Uncertainty Instrument (Section 2)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.890	36.130	36.130	2.890	36.130	36.130
2	1.713	21.418	57.548	1.713	21.418	57.548
3	1.178	14.723	72.271	1.178	14.723	72.271
4	.768	9.604	81.875			
5	.640	8.003	89.878			
6	.380	4.748	94.626			
7	.330	4.127	98.754			
8	9.971E-02	1.246	100.000			

Extraction Method: Principal Component Analysis.

The three factors extracted were rotated, resulting in clean loadings for their respective indicators (Tables 3.16a & 3.16b). For section 1, the indicators for requirements uncertainty had loadings of .864, .865, and .714; outcome uncertainty indicators had loadings of .826 and .828; and technological uncertainty indicators had loadings of .798, .777, and .897.

Table 3.16a: Rotated Factor Loadings of Project Uncertainty Indicators (Section 1)

Rotated Component Matrix[a]

	Component		
	1	2	3
RU1	-6.98E-02	.864	-.131
RU2	-3.53E-02	.865	.235
RU3	-.212	.714	.290
OU1	-.295	-2.42E-02	.826
OU2	5.677E-03	.300	.828
TU2	.798	-.165	-.119
TU3	.777	-1.57E-02	-.261
TU4	.897	-9.82E-02	2.915E-02

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 5 iterations.

For section 2, the indicators for requirements uncertainty loaded together with loadings of .852, .669, and .802; outcome uncertainty indicators loaded together with loadings of .912 and .936; and technological uncertainty indicators loaded together with loadings of .669, .770, and .761. There were no cross-loadings.

Table 3.16b: Factor Loadings of Project Uncertainty Indicators (Section 2)

Rotated Component Matrix[a]

	Component		
	1	2	3
RU1	4.044E-02	.852	.159
RU2	7.523E-02	.669	-.331
RU3	.266	.802	-2.72E-02
OU1	.912	.204	-.220
OU2	.936	.144	-7.85E-02
TU2	.141	-.209	.699
TU3	-.352	.146	.770
TU4	-.244	-2.52E-03	.761

Extraction Method: Principal Component Analysis.
Rotation Method: Varimax with Kaiser Normalization.

[a] Rotation converged in 6 iterations.

Reliabilities were then calculated for each group of indicators for both the sections (Table 3.17). As expected from the previous results, reliabilities for technological uncertainty scale for section 2 was low. Unexpectedly, the reliability for outcome uncertainty scale was low for section 1. To correct these anomalies, the items for these two scales were reworded. Additionally, one new item was added to the outcome uncertainty scale.

Table 3.17: Reliabilities for Project Uncertainty Scales

Construct	**Section 1**	**Section 2**
Requirements Uncertainty	.7775	.7083
Outcome Uncertainty	.6705	.9386
Technological Uncertainty	.7729	.6578

Project Interdependence Scales

Project interdependence scale had 3 indicators. The eigenvalue results shown in Tables 3.18a & 3.18b propose a single-factor solution for this instrument for both sections. For section 1, the eigenvalue is 2.286, and for section 2, the eigenvalue is 2.155. The single factor solution explains 76.195 percent of the total variance for section 1 and 71.842 percent variance for section 2.

Table 3.18a: Eigenvalues for Project Interdependence Instrument (Section 1)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.286	76.195	76.195	2.286	76.195	76.195
2	.406	13.537	89.733			
3	.308	10.267	100.000			

Extraction Method: Principal Component Analysis.

Table 3.18b: Eigenvalues for Project Interdependence Instrument (Section 2)

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	2.155	71.842	71.842	2.155	71.842	71.842
2	.622	20.738	92.580			
3	.223	7.420	100.000			

Extraction Method: Principal Component Analysis.

Tables 3.19a & 3.19b present the loadings for indicators in each section. For section 1, the indicators had loadings of .894, .858, and .866, while for section 2, the indicators had loadings of .850, .926, and .758. There were no cross-loadings.

Table 3.19a: Factor Loadings of Project Interdependence Indicators (Section 1)

Component Matrix[a]

	Component
	1
PI1	.894
PI2	.858
PI5	.866

Extraction Method: Principal Component Analysis.

[a] 1 components extracted.

Table 3.19b: Factor Loadings of Project Interdependence Indicators (Section 2)

Component Matrix[a]

	Component
	1
PI1	.850
PI2	.926
PI5	.758

Extraction Method: Principal Component Analysis.

[a] 1 components extracted.

Reliability of the project interdependence instrument was then calculated for both the sections. Table 3.20 presents the reliabilities for both the sections.

Table 3.20: Reliabilities for Project Interdependence Scales

Construct	**Section 1**	**Section 2**
Project Interdependence	.8436	.8019

Based on the results of preliminary test, 37 items were retained for the final questionnaire. Table 3.21 lists the constructs and the corresponding number of items.

Table 3.21 Research Constructs and Sub-Constructs Included in the Questionnaire

Constructs and Sub-Constructs	**Items**
Knowledge Integration	**(7)**
• Internal integration (IKI)	4
• External integration (EKI)	3
Team Antecedents	**(11)**
• Team's knowledge heterogeneity (KH)	3
• Team's relational capital (RC)	3
• Boundary-Buffering Processes	
• Sentry processes (SP)	3
• Guard processes (GP)	2
IT-Usage	**(7)**
• Nature of IT-usage (ITU)	3
• Frequency of IT-usage (ITF)	4
Project Antecedents	**(12)**
• Project uncertainty (RU, TU, OU)	9
• Project interdependence (TI & OI)	3

Procedures

Two questionnaires (Q1 & Q2) were developed for the purpose of data collection. Q1 was designed in the same manner as the questionnaire used in the preliminary test, i.e., it had two sections (1& 2) titled “most successful project” and “least successful project” respectively. In Q2, these two sections were flipped. The questionnaires are presented in the Appendix.

Data was collected from 150 project leaders in 9 mid- to large-size software services firms. Project leaders were chosen as respondents as they have an overall understanding of most issues pertaining to the team and the project. To secure a firm's participation, the chief knowledge officer (CKO) of each firm was contacted by phone or e-mail. The questionnaires were administered through the World Wide Web. Other than the 38 items, the questionnaire included questions regarding team size, project type, project duration, and individual demographics such as project leader's gender and overall experience,

After obtaining necessary approvals from the firms, link to the questionnaires was forwarded to the CKOs, who subsequently e-mailed it to multiple project leaders in their respective organizations. The link, which was common for both Q1 and Q2, first opened a letter explaining the intent of the study, and the facts that the participation to the study is voluntary and the responses to the study will be anonymous. Project leaders agreeing to participate in the study proceeded to the questionnaire by clicking on a second link at the end of the letter. To ensure that same number of Q1 and Q2 were filled, the second link alternatively routed the respondents to Q1 and Q2.

Statistical Analyses

Partial least squares (PLS) technique was used to validate the measurement model and to test the hypothesized relationships. PLS is a second-generation structural equation modeling technique that utilizes a correlational, principal component-based approach to estimation (Majchrzak et al. Forthcoming). PLS is a favorable technique for causal-predictive analysis in situations characterized by early stages of theory development

(Kankanhalli et al. 2001). As this study is an early attempt to develop a theoretical framework of teams' knowledge integration, PLS is an appropriate technique for this study.

PLS is also recommended above ANOVA and regression especially in research situations involving moderator analysis (Chin et al. 2003). Regression typically utilizes interaction terms to conduct moderator analysis. Moderated regression involving multiple item measures (such as this study) holds two key assumptions. The first assumption, called "equal item reliability," implies equal contribution of all items towards estimating the interaction effect. The second assumption, referred to as the "unchanging scale reliability," presumes no change in the reliability of the summated multiple item scale when it is applied in the theoretical model. But, as Chin et al. profess, "Unfortunately, by virtue of the summation process, we have no opportunity to assess the validity of these two assumptions…"(2003: 190). Thus moderator's measurement error should be considered both in the initial reliability assessment, and the subsequent analysis of the theoretical model. The latent variable modeling approach within PLS allows the subsequent assessment of this error, thereby providing more accurate estimates of the interaction effects (Chin et al. 2003).

Summary

This chapter discusses the procedures and methodology used in the research. Also discussed are the details of preliminary tests conducted on various instruments to develop the final questionnaires. The following chapter presents the results of the final data collection.

CHAPTER 4: ANALYSES AND RESULTS

Data Collection

Data was collected from 150 project leaders in nine mid- to large-size software services firms. The nine firms provide custom-made software solutions to Fortune 1000 clients. They were chosen because of similarity in their nature of operations. Additionally, all the firms are CMM Level 5 companies, which ensured consistency of their software development processes.

Links to the questionnaires were forwarded to 225 project leaders in these organizations, of which 161 completed the questionnaire, resulting in a 71.56 percent response rate. Of the 161 responses, eleven were incomplete, and were excluded from subsequent analyses. 83 of the remaining 150 questionnaires were Q1 while 67 were Q2. Tables 4.1(a & b) present the demographics of respondents for both versions of the questionnaire.

For Q1, 18 percent of respondents were female while 82 percent were male. They had an average industry experience of 8.2 years indicating that the respondents had a good understanding of various project management processes. In terms of project type, 88 percent of both most as well as least successful projects involved developing customized software solution for clients. The rest 12 percent were product development projects. The average project duration for most successful projects was 12 months as compared to 11 months for the least successful ones. Most successful project teams had an average of 18 members, while the least successful ones had 15.

Table 4.1(a): Demographics of Respondents (Q1)

Demographic		**Overall**	**Most Successful Projects**	**Least Successful Projects**
Gender	Female	18%		
	Male	82%		
Average Industry Experience (In years)		8.2		
Project Type	Developing Customized Solution	88%	88%	88%
	Product Development	12%	12%	12%
Average Project Duration (In months)		11.5	12	11
Average Team Size (Number of Team Members)		16	18	15

For Q2, 26 percent of respondents were female while 74 percent were male. Compared to Q1 respondents, they had a slightly higher average industry experience of 9.4 years. In terms of project type, 80 percent of most successful projects and 82 percent of least successful projects involved developing customized software solution for clients. 20 percent of most successful projects and 18 percent of least successful projects included product development. The average project duration for most successful projects was 16 months as compared to 13 months for the least successful ones. Most successful project teams had 12 members, while the least successful ones had 17.

Table 4.1(b): Demographics of Respondents (Q2)

Demographic		**Overall**	**Most Successful Projects**	**Least Successful Projects**
Gender	Female	26%		
	Male	74%		
Average Industry Experience (In years)		9.4		
Project Type	Developing Customized Solution	81%	80%	82%
	Product Development	19%	20%	18%
Average Project Duration (In months)		14.5	16	13
Average Team Size (Number of Team Members)		15	12	17

Data Analyses

Each of the 150 valid questionnaires included data concerning the most successful project as well as the least successful project in the experience of the responding project leader. Thus, a combined dataset of 300 projects was available for further analyses. Partial least squares (PLS)[1] latent modeling technique was employed to develop and test the measurement and structural model, which further aided the testing of research hypotheses.

---

[1] I used PLS-GRAPH version 3.0 build 1126 to run PLS.

Two kinds of analyses were conducted for this study. The first type of analysis included examining the overall dataset of 300 projects for testing the research model. For the second type of analysis, data were separated into two files, containing responses from the most successful and the least successful projects respectively. The two data sets were then analyzed separately using PLS technique. Measurement and structural models were developed and tested for following four analyses:

- EKI – Most: Analysis for external knowledge integration (EKI) in most successful projects
- EKI – Least: Analysis for external knowledge integration (EKI) in least successful projects
- IKI – Most: Analysis for internal knowledge integration (IKI) in most successful projects
- IKI – Least: Analysis for internal knowledge integration (EKI) in least successful projects

Details of these analyses are presented in the following sections. I begin the discussion with the combined analysis of all projects.

Combined Analysis

In light of the hypotheses, separate analyses were conducted for external knowledge integration and internal knowledge integration. Main effects and moderation effects were tested with separate models.

External Knowledge Integration (EKI): Measurement Model

Tables 4.2 and 4.3 present the results of PLS component-based analysis conducted to examine (1) individual item reliability, (2) internal consistency, and (3) discriminant validity. Individual item reliability was assessed by examining the loading and the cross loadings of each item (Table 4.2). Boldface numbers are loadings of indicators on their own construct, the rest are cross-loadings. To calculate cross-loadings, a factor score for each construct was calculated based on the weighted sum (provided by PLS-Graph) of the construct's indicators. These scores were correlated with individual indicators to obtain cross-loadings.

Although some cross loadings are noticeable, all items have a higher loading on their own construct than on other constructs. One item on the relational capital scale (RC_3) has a less than prescribed loading of 0.7. Previous studies have observed that well-established scales sometimes show poor factor loadings when they are used in causal modeling (Barclay et al. 1995; Yoo et al. 2001). Given that the relational capital scale is a standard scale from the literature, and given the importance to retain items from the original scale to maintain the comparability of my results with other studies using the same scales (Barclay et al. 1995), this item was included in the final analysis.

Table 4.2: Loadings and Cross-Loadings of Items

Item	**RU**	**OU**	**TU**	**EKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**IT Usage**
RU_1	**0.80**	0.13	0.07	0.02	0.10	-0.03	-0.02	0.01	-0.10	-0.02
RU_2	**0.76**	0.38	0.09	-0.13	0.00	-0.05	-0.07	-0.03	0.04	-0.09
RU_3	**0.74**	0.26	0.16	-0.03	0.10	0.00	-0.04	-0.03	0.01	-0.03
OU_1	0.29	**0.82**	0.06	-0.06	-0.02	-0.07	-0.03	0.01	-0.04	-0.05
OU_2	0.18	**0.90**	0.12	-0.09	0.04	-0.02	-0.08	0.02	-0.05	-0.09
OU_3	0.22	**0.89**	0.14	-0.07	0.01	-0.07	-0.06	-0.06	-0.03	-0.06
TU_1	0.01	0.15	**0.80**	-0.12	0.12	0.08	0.07	-0.06	0.03	-0.06
TU_2	0.09	0.04	**0.92**	-0.02	0.00	0.03	-0.02	0.01	-0.02	-0.03
TU_3	0.20	0.11	**0.85**	0.05	-0.09	-0.03	-0.08	0.03	-0.01	-0.01
EKI_1	0.00	-0.11	-0.11	**0.72**	0.15	0.16	0.20	-0.08	0.17	0.10
EKI_2	-0.08	-0.08	0.01	**0.75**	0.04	0.05	0.13	0.06	0.08	0.07
EKI_3	-0.06	-0.10	0.00	**0.71**	0.24	0.18	0.18	-0.07	0.19	0.11
KH_1	0.07	0.10	0.01	0.07	**0.81**	0.10	0.11	0.04	0.15	-0.01
KH_2	0.03	0.03	0.05	0.15	**0.87**	0.07	0.09	0.08	0.03	-0.03
KH_3	0.12	-0.09	-0.02	0.11	**0.82**	0.09	0.16	0.10	0.11	0.11
RC_1	-0.01	-0.04	0.08	0.10	0.11	**0.84**	0.17	0.12	-0.07	0.12
RC_2	-0.06	-0.02	0.01	0.12	0.14	**0.87**	0.06	0.01	0.08	-0.07
RC_3	-0.01	-0.19	0.00	0.14	0.00	**0.63**	0.26	-0.04	0.34	0.10
SP_1	-0.09	-0.06	-0.05	0.14	0.24	0.13	**0.76**	0.05	0.30	0.08
SP_2	-0.05	-0.07	-0.07	0.22	0.17	0.13	**0.78**	0.08	0.23	0.07
SP_3	-0.03	-0.07	0.08	0.16	0.06	0.17	**0.79**	0.13	-0.09	0.02
GP_1	0.06	-0.05	0.01	-0.02	0.06	-0.01	0.16	**0.87**	0.08	-0.07
GP_2	0.01	-0.17	-0.11	-0.09	0.08	0.14	0.12	**0.72**	0.18	0.11
PI_1	-0.05	-0.04	0.00	0.22	0.13	0.08	0.20	0.12	**0.82**	0.07
PI_2	-0.03	-0.07	0.00	0.18	0.20	0.06	0.09	0.10	**0.75**	0.04
PI_3	-0.04	0.01	0.02	-0.02	0.18	0.06	0.16	0.09	**0.73**	0.09
ITU_1	-0.02	-0.09	-0.01	0.09	0.07	0.00	0.09	0.02	0.02	**0.80**
ITU_2	-0.10	-0.10	-0.07	0.09	0.02	0.07	0.07	0.03	0.05	**0.83**
ITU_3	-0.05	-0.06	-0.08	0.14	-0.01	0.06	0.01	0.01	0.10	**0.73**
ITF_1	-0.04	-0.02	-0.06	0.16	0.17	-0.01	0.10	0.01	0.04	**0.84**
ITF_2	-0.03	0.03	-0.02	0.11	0.16	-0.01	0.13	0.02	0.07	**0.88**
ITF_3	-0.06	-0.06	-0.01	0.13	0.08	0.06	0.12	0.00	0.11	**0.87**
ITF_4	-0.08	-0.04	-0.03	0.07	0.11	0.06	0.16	0.01	0.07	**0.89**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **EKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Internal consistency was examined using the alpha coefficients for each scale (Table 4.3) used in this analysis (alpha coefficient of the IKI scale is reported later in a separate section). They are all greater than the recommended value of 0.7 (Nunally 1978). Composite reliabilities (ρ_c), which are a more accurate measure of internal consistency as they avoid the assumption of equal weighting of items, are even higher. Another conservative criterion is average variance extracted (AVE), which measures the amount of variance that a latent variable captures from its indicators (Fornell et al. 1981). All AVE values are higher than the recommended value of 0.5 (Chin 1998). AVE values can also be used to examine the discriminant validity. Comparing the square root of each AVE value (bold figures on the diagonal in Table 4.3, representing the average association of each construct to its measures), with the correlations among constructs (the off-diagonal figures) points out the closeness of association of each construct to its measures than to the measures of other constructs. A more conservative estimate is to compare the AVE values themselves (square roots of AVE values are higher than the values themselves) to the correlations. This comparison also supports the discriminant validity of the constructs included in this study.

Table 4.3: Inter Construct Correlations - Consistency and Reliability Tests

Construct (# of Items)	**Cronbach's Alpha**	**Composite Reliability (ρ_c)**	**AVE**	**RU**	**OU**	**TU**	**EKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**ITU**	**ITF**
RU (3)	0.774	0.869	0.690	**0.831**										
OU (3)	0.908	0.943	0.847	0.551	**0.920**									
TU (3)	0.839	0.905	0.761	0.279	0.257	**0.872**								
EKI (3)	0.750	0.864	0.680	-0.171	-0.238	-0.095	**0.824**							
KH (3)	0.858	0.914	0.780	0.115	0.014	0.017	0.358	**0.883**						
RC (3)	0.777	0.867	0.685	-0.120	-0.200	0.018	0.401	0.259	**0.827**					
SP (3)	0.836	0.901	0.754	-0.155	-0.195	-0.070	0.502	0.394	0.436	**0.868**				
GP (2)	0.733	0.850	0.743	-0.072	-0.157	-0.090	0.060	0.175	0.193	0.210	**0.862**			
PI (3)	0.742	0.849	0.741	-0.018	-0.129	-0.024	0.431	0.386	0.301	0.468	0.210	**0.861**		
ITU (3)	0.915	0.947	0.856	-0.188	-0.201	-0.128	0.346	0.157	0.193	0.271	0.120	0.258	**0.925**	
ITF (4)	0.953	0.966	0.876	-0.151	-0.106	-0.094	0.407	0.320	0.197	0.401	0.078	0.349	0.689	**0.936**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **EKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

External Knowledge Integration (EKI): Structural Model

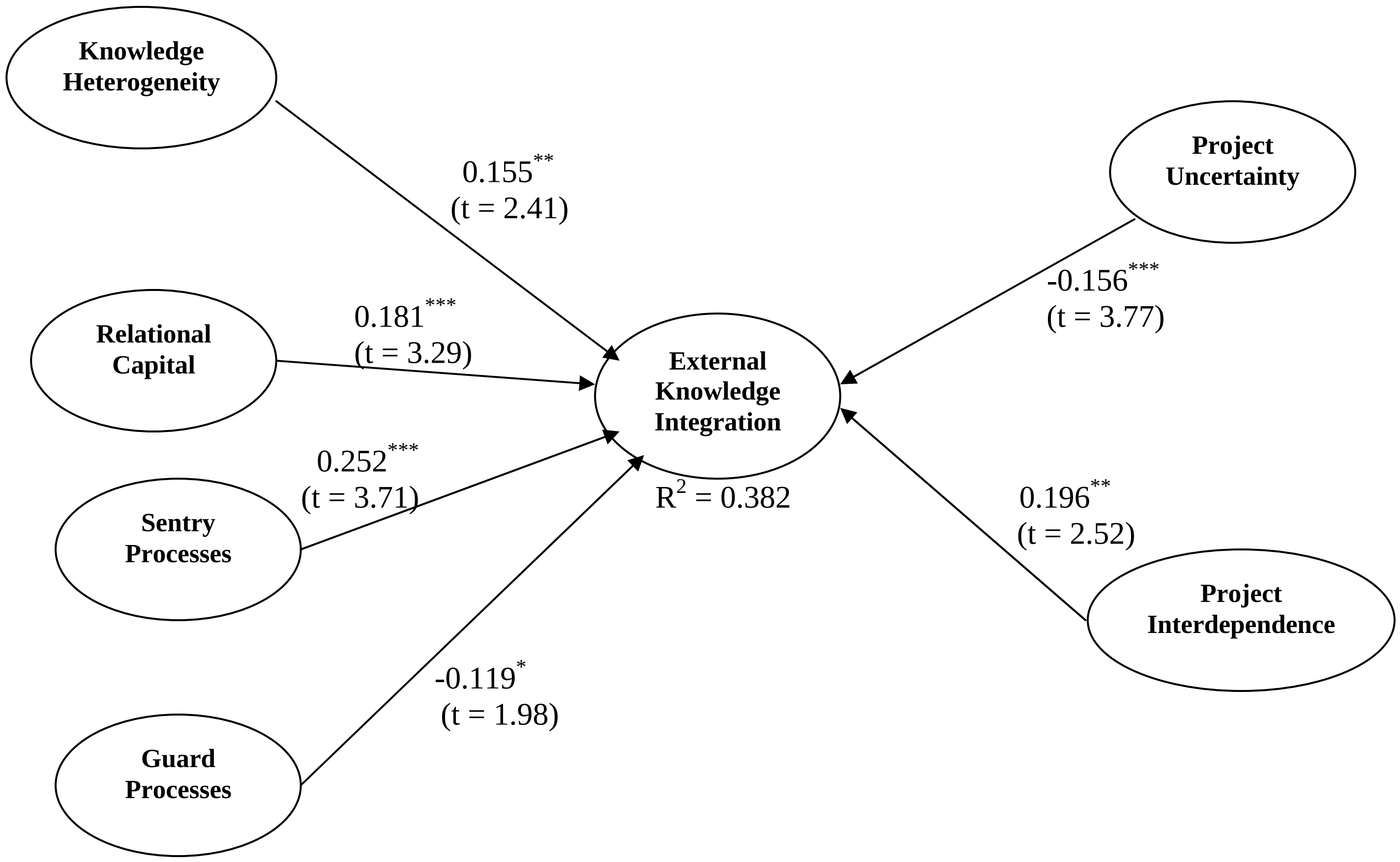
Table 4.4 presents the outer model loadings of the items on each construct, which represent the convergent validity of various scales. Results show convergent validity as the t-values of outer model loadings are higher than 1.96 (Gefen et al. 2005).

Figure 4.1 presents the PLS results of the EKI main effects model in a graphical form. All paths are significant with the model accounting for 38.2 percent of the variance in external knowledge integration. PLS-Graph provides Q^2 as another measure of predictive relevance of the structural model (Wold 1982). It is calculated using a blindfolding procedure that excludes a part of the data for a particular block of indicators during parameter estimations, and then tries to estimate the omitted part using the estimated parameter (Chin 1998). $Q^2 > 0$ means that the model has predictive relevance, whereas $Q^2 < 0$ suggests a lack of it. A Q^2 of 0.15 was obtained, which suggests that the model has predictive relevance.

Table 4.4: Outer Model Loadings for EKI Analysis

Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
External Knowledge Integration (EKI)				
EKI_1	0.8358	0.8362	0.0269	31.0606
EKI_2	0.7446	0.7430	0.0388	19.1847
EKI_3	0.8863	0.8876	0.0153	57.9669
Knowledge Heterogeneity (KH)				
KH_1	0.8635	0.8636	0.0224	38.5843
KH_2	0.9058	0.9059	0.0173	52.4535
KH_3	0.8791	0.8813	0.0180	48.8265
Relational Capital (RC)				
RC_1	0.7999	0.7821	0.0514	15.5517
RC_2	0.8362	0.8234	0.0549	15.2367
RC_3	0.8462	0.8527	0.0321	26.3693
Sentry Processes (SP)				
SP_1	0.8975	0.8966	0.0191	46.9166
SP_2	0.9225	0.9220	0.0146	63.1228
SP_3	0.7779	0.7725	0.0447	17.3864
Guard Processes (GP)				
GP_1	0.7261	0.7117	0.3300	2.2004
GP_2	0.9773	0.7960	0.2573	3.7979
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.5947	0.5962	0.0428	13.8971
RU_2	0.7477	0.7481	0.0292	25.6393
RU_3	0.6746	0.6725	0.0367	18.3597
Outcome Uncertainty (OU)				
OU_1	0.7620	0.7664	0.0313	24.3181
OU_2	0.7841	0.7872	0.0247	31.7190
OU_3	0.8104	0.8115	0.0214	37.8883
Technological Uncertainty (TU)				
TU_1	0.4821	0.4629	0.0692	6.9691
TU_2	0.4911	0.4710	0.0720	6.8240
TU_3	0.5637	0.5459	0.0653	8.6343
Project Interdependence (PI)				
PI_1	0.8743	0.8661	0.0264	33.1230
PI_2	0.9055	0.9052	0.0190	47.6072
PI_3	0.6333	0.6408	0.0666	9.5068
IT Usage (ITU)				
ITU_1	0.9006	0.9007	0.0195	46.0765
ITU_2	0.9478	0.9444	0.0105	90.3686
ITU_3	0.9264	0.9245	0.0143	64.8485
IT Usage Frequency (ITF)				
ITF_1	0.9236	0.9226	0.0136	67.9885
ITF_2	0.9433	0.9422	0.0127	74.2280
ITF_3	0.9331	0.9318	0.0119	78.4879
ITF_4	0.9444	0.9420	0.0116	81.5076

Figure 4.1: EKI Main Effects Analysis Results



* Significant at .05 level; ** Significant at .01 level; *** Significant at .001 level

External Knowledge Integration (EKI): Main Effects Analysis

In hypothesis 1b, it was proposed that knowledge heterogeneity improves software teams' external knowledge integration. Empirical evidence supports the hypothesis (t = 2.413, p < .01).

In hypothesis 2b, it was predicted that relational capital improves software teams' external knowledge integration. Empirical evidence supports the hypothesis (t = 3.2891, p < .001).

In hypothesis 3b, it was suggested that external knowledge integration deteriorates in software teams performing sentry processes. Surprisingly, empirical evidence suggests the contrary - that sentry processes had a highly significant positive influence on external knowledge integration (t = 3.7144, p < .001).

In hypothesis 4b, it was proposed that external knowledge integration deteriorates in software teams performing guard processes. Empirical evidence supports the hypothesis (t = 1.976, p < .05).

In hypothesis 5b, it was projected that project uncertainty improves software teams' external knowledge integration. Empirical evidence suggests the contrary. Project uncertainty had a highly significant negative influence on external knowledge integration (t = 3.772, p < .001).

Lastly, in hypothesis 6b, it was predicted that high project interdependence increases software teams' external knowledge integration. Empirical evidence supports this hypothesis (t = 2.524, p < .01).

Table 4.5 summarizes these results. Implications of these results are discussed in greater detail in the next chapter.

Table 4.5: Summary of Main Effects Analysis for EKI

Hypothesis: Path	**Beta**	**T-stat**
H1b: Knowledge Heterogeneity → External Knowledge Integration	**0.1550****	**2.4131**
H2b: Relational Capital → External Knowledge Integration	**0.1810*****	**3.2891**
H3b: Sentry Processes → External Knowledge Integration	**0.2520*****	**3.7144**
H4b: Guard Processes → External Knowledge Integration	**-0.1190***	**1.9759**
H5b: Project Uncertainty → External Knowledge Integration	**-0.1560*****	**3.7723**
H6b: Project Interdependence → External Knowledge Integration	**0.1960****	**2.5236**

$^{*}p < 0.05$; $^{**}p < 0.01$; $^{***}p < 0.001$

External Knowledge Integration (EKI): Moderation Effects Analysis

Table 4.6 depicts a summary of results of moderation effects analysis for external knowledge integration. The analysis was conducted using the procedure outlined by Chin et al. (2003), as per which moderation is examined by introducing an interaction term in the main effects model.

In this study, IT-usage is hypothesized to moderate the influence of each predictor variable on external knowledge integration. To examine these effects, which are represented by hypotheses H7b - H12b, six separate models were run to test the interaction of IT-usage with knowledge heterogeneity, relational capital, sentry processes, guard processes, project uncertainty, and project interdependence respectively. Results are presented in Table 4.6. All R^2 values are in excess of 40 percent. All Q^2 values are higher than 0.1 suggesting predictive relevance of the models. Also presented in Table 4.6 are f^2 values, which represent the effect size for interaction. f^2 can be calculated as:

$$\frac{(R^2_{\text{included}} - R^2_{\text{excluded}})}{(1- R^2_{\text{included}})}$$

Where R^2_{included} and R^2_{excluded} are the R-squares for the dependent latent variable when the interaction term is included and omitted in the main effects model respectively. Values of 0.02, 0.15, and 0.35 are recommended as small, moderate, and large effects respectively (Cohen 1988). Most of the effects are between small and medium, but are larger than found in most past IS studies. It is crucial to understand that a small f^2 does not necessarily connote an unimportant effect. As Chin et al. (2003) explain, “Even a small interaction can be significant under extreme moderating conditions, if the resulting beta changes are meaningful, then it is important to take these conditions into account” (p. 211). Beta values presented in Table 4.6 suggest that all the hypothesized interaction paths are significant.

Table 4.6: Summary of Interaction Effects Analysis for EKI

Hypothesis: Path	R^2	Q^2	f^2	**Beta**	**T-stat**
H7b: IT Usage* KH	42.9%	.13	.08	$\mathbf{0.377}^{**}$	**2.9903**
H8b: IT Usage* RC	40.5%	.17	.04	$\mathbf{0.239}^{**}$	**2.3624**
H9: IT Usage* SP	41.7%	.12	.06	$\mathbf{0.320}^{**}$	**2.7746**
H10: IT Usage* GP	44.6%	.14	.10	$\mathbf{0.400}^{***}$	**3.0906**
H11b: IT Usage* PU	42%	.13	.06	$\mathbf{0.363}^{***}$	**3.1826**
H12b: IT Usage* PI	40.6%	.17	.04	$\mathbf{0.202}^{**}$	**2.9851**

$^{*}p < 0.05$; $^{**}p < 0.01$; $^{***}p < 0.001$

In hypothesis 7b, it was predicted that IT-usage moderates the influence of a software team’s knowledge heterogeneity on its external knowledge integration. Evidence strongly supports the hypothesis ($t = 2.990$; $p < .01$). IT-usage increased the positive influence of knowledge heterogeneity on external knowledge integration.

In hypothesis 8b, it was suggested that IT-usage moderates the influence of a software team’s relational capital on its external knowledge integration. Evidence supports this moderation ($t = 2.362$, $p < .01$). Thus, IT-usage increased the positive influence of relational capital on external knowledge integration.

In hypothesis 9, it was proposed that IT-usage moderates the influence of sentry processes on software teams’ external knowledge integration. Results support this moderation ($t = 2.775$, $p < .01$). IT-usage strengthened the positive influence of sentry processes on external knowledge integration.

In hypothesis 10, it was proposed that IT-usage moderates the influence of guard processes on software teams' external knowledge integration. Results support a strong moderation (t = 3.091, p < .001). IT-usage nullified the negative influence of guard processes on external knowledge integration.

In hypothesis 11b, it was suggested that IT-usage moderates the influence of project uncertainty on software teams' external knowledge integration. Empirical evidence strongly supports this moderation (t = 3.183, p < .001). IT-usage nullified the negative influence of project uncertainty on external knowledge integration.

Lastly, in hypothesis 12b, it was predicted that IT-usage moderates the influence of project interdependence on software teams' external knowledge integration. The results are as predicted, and empirical evidence supports a strong positive moderation (t = 2.985, p < .01). Thus, IT-usage increased the positive influence of project interdependence on external knowledge integration.

Internal Knowledge Integration (IKI): Measurement Model

Tables 4.7 and 4.8 present the results of PLS component-based analysis conducted to examine (1) individual item reliability, (2) internal consistency, and (3) discriminant validity. Similar to the EKI analysis, individual item reliability was assessed by examining the loading and the cross loading of each item (Table 4.7). Although there are some cross loadings, all items have a higher loading on their own construct than on other constructs.

Table 4.7: Loadings and Cross-Loadings of Items

Item	RU	OU	TU	IKI	KH	RC	SP	GP	PI	IT Usage
RU_1	**0.74**	0.19	0.09	-0.01	0.13	-0.05	-0.02	0.01	-0.08	-0.14
RU_2	**0.70**	0.44	0.10	-0.06	0.02	-0.08	-0.06	-0.03	-0.02	-0.10
RU_3	**0.69**	0.32	0.18	0.00	0.12	-0.03	-0.04	-0.04	-0.01	-0.01
OU_1	0.29	**0.79**	0.05	-0.18	-0.02	-0.02	-0.02	0.01	0.01	-0.08
OU_2	0.19	**0.87**	0.11	-0.13	0.03	0.01	-0.07	0.02	-0.05	-0.07
OU_3	0.24	**0.86**	0.13	-0.15	0.01	-0.02	-0.05	-0.06	-0.02	-0.05
TU_1	0.05	0.12	**0.79**	-0.04	0.08	0.09	0.07	-0.04	-0.03	-0.08
TU_2	0.06	0.05	**0.93**	-0.07	0.01	0.03	-0.02	0.01	0.03	-0.03
TU_3	0.20	0.09	**0.84**	-0.06	-0.08	-0.02	-0.06	0.02	0.02	-0.04
IKI_1	0.02	-0.13	-0.11	**0.86**	0.10	0.12	0.12	0.07	0.10	0.07
IKI_2	-0.05	-0.11	-0.07	**0.84**	0.18	0.09	0.13	0.02	0.07	0.15
IKI_3	0.01	-0.11	-0.01	**0.84**	0.11	0.18	0.10	0.04	0.08	0.09
IKI_4	-0.08	-0.10	-0.03	**0.82**	0.08	0.15	0.22	-0.05	0.08	0.18
KH_1	0.07	0.11	0.01	0.19	**0.79**	0.06	0.09	0.04	0.16	0.16
KH_2	0.03	0.03	0.05	0.11	**0.88**	0.07	0.10	0.07	0.13	0.09
KH_3	0.19	-0.13	-0.04	0.11	**0.79**	0.12	0.16	0.11	0.14	0.13
RC_1	-0.01	-0.04	0.09	0.27	0.09	**0.82**	0.13	0.13	-0.04	-0.01
RC_2	-0.14	0.05	0.03	0.24	0.17	**0.80**	0.09	0.00	0.11	0.04
RC_3	0.02	-0.16	0.00	0.50	-0.06	**0.53**	0.21	-0.03	0.17	0.19
SP_1	-0.01	-0.10	-0.07	0.27	0.19	0.11	**0.74**	0.05	0.28	0.17
SP_2	-0.02	-0.07	-0.07	0.32	0.14	0.07	**0.76**	0.08	0.20	0.21
SP_3	-0.09	-0.03	0.10	0.14	0.10	0.14	**0.79**	0.12	0.03	0.15
GP_1	0.00	0.00	0.03	0.05	0.08	-0.06	0.16	**0.87**	0.11	0.01
GP_2	0.09	-0.19	-0.12	0.13	0.02	0.12	0.10	**0.74**	0.02	0.06
PI_1	0.05	-0.09	-0.03	0.30	0.05	0.05	0.18	0.11	**0.77**	0.10
PI_2	-0.03	-0.06	-0.01	0.11	0.19	0.05	0.10	0.07	**0.86**	0.17
PI_3	-0.17	0.09	0.06	-0.02	0.23	0.03	0.12	0.10	**0.61**	0.21
ITU_1	0.10	-0.20	-0.02	0.01	0.01	0.10	0.03	0.05	0.09	**0.81**
ITU_2	0.06	-0.24	-0.10	-0.04	-0.06	0.20	0.04	0.06	0.07	**0.84**
ITU_3	0.07	-0.14	-0.10	0.08	-0.07	0.13	-0.03	0.03	0.09	**0.86**
ITF_1	-0.16	0.08	-0.03	0.12	0.24	-0.09	0.12	-0.01	0.12	**0.82**
ITF_2	-0.16	0.15	0.01	0.13	0.23	-0.11	0.16	0.00	0.14	**0.82**
ITF_3	-0.14	0.03	0.00	0.21	0.11	-0.05	0.15	-0.01	0.06	**0.84**
ITF_4	-0.19	0.07	-0.01	0.19	0.16	-0.06	0.18	-0.01	0.06	**0.84**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **IKI**: Internal Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Table 4.8 presents the alpha coefficients for each scale used in this analysis. They are all greater than the recommended value of 0.7 (Nunally 1978). Composite reliabilities (ρ_c) are even higher. All AVE values are also higher than the recommended value of 0.5 (Chin 1998). A comparison of square root of each AVE value (bold figures on the diagonal in Table 4.8), with the correlations among constructs (the off-diagonal figures) supports the discriminant validity of constructs.

Table 4.8: Inter Construct Correlations - Consistency and Reliability Tests

Construct (# of Items)	**Cronbach’s Alpha**	**Composite Reliability (ρ_c)**	**AVE**	**RU**	**OU**	**TU**	**IKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**ITU**	**ITF**
RU (3)	0.774	0.869	0.690	**0.831**										
OU (3)	0.908	0.943	0.847	0.551	**0.920**									
TU (3)	0.839	0.905	0.761	0.279	0.257	**0.872**								
IKI (4)	0.924	0.946	0.814	-0.140	-0.296	-0.140	**0.973**							
KH (3)	0.858	0.914	0.780	0.117	0.016	0.017	0.331	**0.883**						
RC (3)	0.777	0.867	0.685	-0.120	-0.203	0.019	0.582	0.257	**0.827**					
SP (3)	0.836	0.901	0.754	-0.156	-0.195	-0.071	0.518	0.396	0.439	**0.868**				
GP (2)	0.733	0.850	0.743	-0.065	-0.146	-0.079	0.170	0.181	0.188	0.259	**0.862**			
PI (3)	0.742	0.849	0.741	-0.108	-0.131	-0.026	0.362	0.381	0.305	0.474	0.226	**0.861**		
ITU (3)	0.915	0.947	0.856	-0.184	-0.199	-0.126	0.234	0.160	0.195	0.271	0.109	0.257	**0.925**	
ITF (4)	0.953	0.966	0.876	-0.154	-0.109	-0.094	0.304	0.319	0.197	0.402	0.083	0.340	0.691	**0.936**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **IKI**: Internal Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Internal Knowledge Integration (IKI): Structural Model

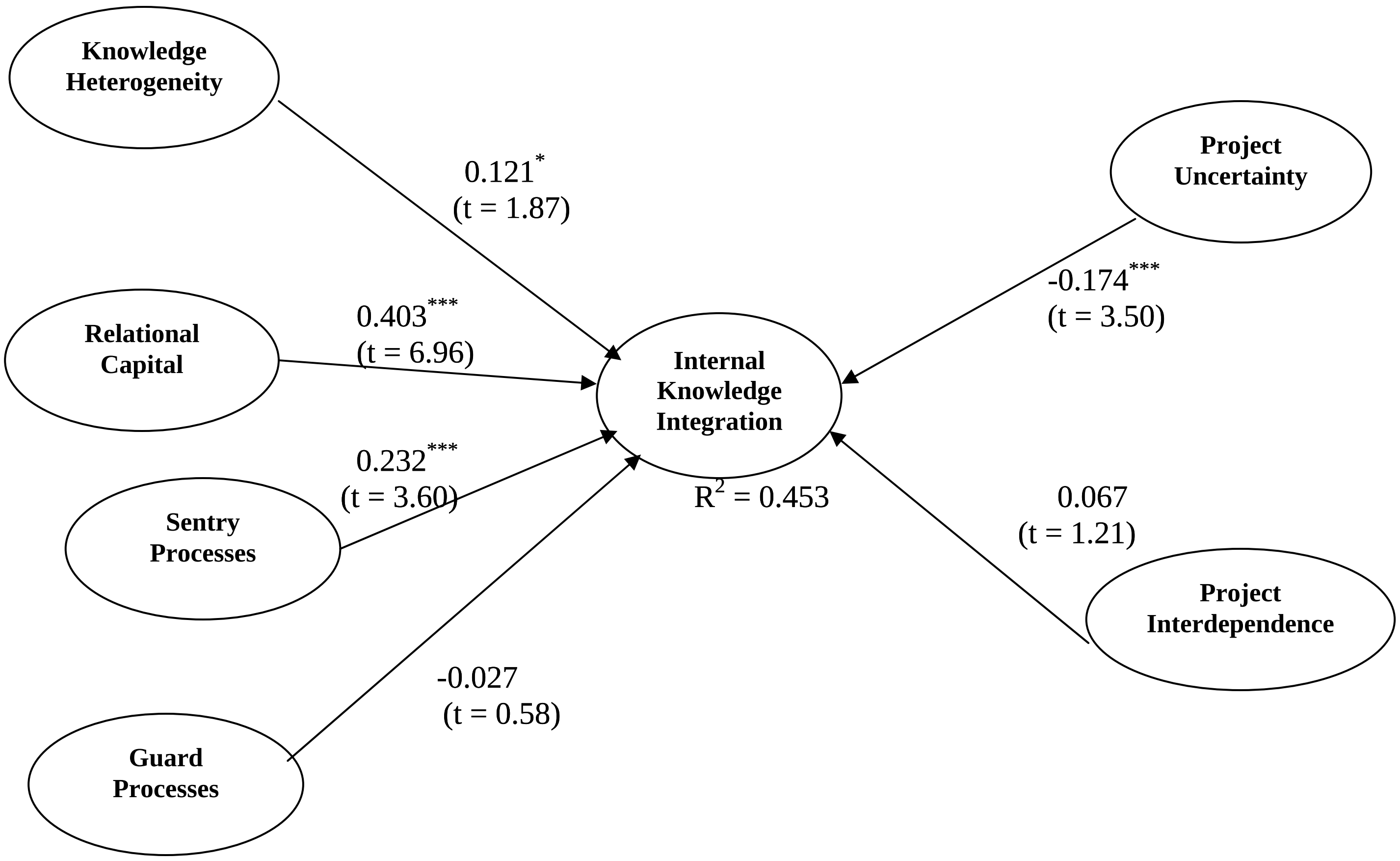
Table 4.9 presents the outer model loadings of the items on each construct, which represent the convergent validity of various scales. Results suggest convergent validity.

Figure 4.2 presents the PLS results of main effects model in a graphical form. The model accounts for 45.3 percent of the variance in internal knowledge integration. Q^2 of 0.28 suggests that the model has predictive relevance.

Table 4.9: Outer Model Loadings for IKI Analysis

Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
Internal Knowledge Integration (EKI)				
IKI_1	0.9043	0.9025	0.0160	56.6360
IKI_2	0.9081	0.9027	0.0166	54.6721
IKI_3	0.8922	0.8897	0.0168	53.1458
IKI_4	0.9050	0.9006	0.0151	59.9329
Knowledge Heterogeneity (KH)				
KH_1	0.8970	0.9000	0.0202	44.3445
KH_2	0.8743	0.8771	0.0238	36.7958
KH_3	0.8758	0.8745	0.0222	39.3774
Relational Capital (RC)				
RC_1	0.8137	0.8062	0.0390	15.5517
RC_2	0.8743	0.8061	0.0526	15.2367
RC_3	0.8758	0.8608	0.0213	26.3693
Sentry Processes (SP)				
SP_1	0.9024	0.9015	0.0171	52.9197
SP_2	0.9221	0.9224	0.0156	58.9631
SP_3	0.7720	0.7968	0.0459	16.8131
Guard Processes (GP)				
GP_1	0.7933	0.7406	0.1740	4.5583
GP_2	0.9502	0.9388	0.0800	11.8771
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.7866	0.7866	0.0297	26.4552
RU_2	0.8793	0.8788	0.0129	67.9743
RU_3	0.8229	0.8199	0.0239	34.3992
Outcome Uncertainty (OU)				
OU_1	0.8854	0.8870	0.0180	49.2672
OU_2	0.9352	0.9360	0.0098	95.8759
OU_3	0.9386	0.9393	0.0082	115.1080
Technological Uncertainty (TU)				
TU_1	0.8143	0.8122	0.0287	28.4008
TU_2	0.9188	0.9181	0.0116	79.4917
TU_3	0.8812	0.8824	0.0160	55.0249
Project Interdependence (PI)				
PI_1	0.9038	0.8966	0.0231	39.1335
PI_2	0.8821	0.8809	0.0307	28.7498
PI_3	0.6074	0.6126	0.0731	8.3146
IT Usage (ITU)				
ITU_1	0.9046	0.9035	0.0234	38.6202
ITU_2	0.9393	0.9340	0.0191	49.2582
ITU_3	0.9297	0.9279	0.0188	49.3385
IT Usage Frequency (ITF)				
ITF_1	0.9148	0.9133	0.0184	49.6915
ITF_2	0.9389	0.9366	0.0160	58.7924
ITF_3	0.9387	0.9386	0.0121	77.4870
ITF_4	0.9512	0.9496	0.0090	105.1636

Figure 4.2: IKI Main Effects Analysis Results



* Significant at .05 level; *** Significant at .001 level

Internal Knowledge Integration (IKI): Main Effects Analysis

In hypothesis 1a, it was proposed that knowledge heterogeneity improves software teams' internal knowledge integration. Evidence supports the hypothesis (t = 1.867, p < .05).

In hypothesis 2a, it was predicted that relational capital improves software teams' internal knowledge integration. Evidence suggests that relational capital had a highly significant positive influence on internal knowledge integration (t = 6.963, p < .001).

In hypothesis 3a, it was suggested that internal knowledge integration improves in software teams performing sentry processes. Evidence supports the hypothesis (t = 3.5968, p < .001).

In hypothesis 4a, it was proposed that internal knowledge integration improves in software teams performing guard processes. This hypothesis was not supported. Empirical evidence suggests that guard processes had no influence on internal knowledge integration (t = 0.5836).

In hypothesis 5a, a positive relationship was predicted between project uncertainty and software teams' internal knowledge integration. Empirical evidence suggests the contrary. Project uncertainty had a highly significant negative influence on internal knowledge integration (t = 3.498, p < .001).

Lastly, in hypothesis 6a, it was predicted that internal knowledge integration will deteriorate in software teams working on interdependent project. Empirical evidence does not support this hypothesis. Results suggest that project interdependence did not influence software teams' internal knowledge integration (t = 1.207).

Table 4.10 summarizes these results. Implications of these results are discussed in greater detail in the next chapter.

Table 4.10: Summary of Main Effects Analysis for IKI

Hypothesis: Path	**Beta**	**T-stat**
H1a: Knowledge Heterogeneity → Internal Knowledge Integration	**0.1210***	**1.8673**
H2a: Relational Capital → Internal Knowledge Integration	**0.4030*****	**6.9630**
H3a: Sentry Processes → Internal Knowledge Integration	**0.2320*****	**3.5968**
H4a: Guard Processes → Internal Knowledge Integration	-0.0270	0.5836
H5a: Project Uncertainty → Internal Knowledge Integration	**-0.1740*****	**3.4976**
H6a: Project Interdependence → Internal Knowledge Integration	0.0670	1.2068

$^{*}p < 0.05$; $^{***}p < 0.001$

Internal Knowledge Integration (IKI): Moderation Effects Analysis

Table 4.11 depicts a summary of results of the IT moderation effects analysis for internal knowledge integration. Four separate models were run to test the moderation hypotheses. These results are presented as bold elements in Table 4.11. All R^2 values are in excess of 46 percent. All Q^2 values are higher than 0.2 suggesting predictive relevance of the models.

Hypotheses H7a, H8a, H11a, and H12a proposed that IT usage will significantly moderate the respective influence of knowledge heterogeneity (KH), relational capital (RC), project uncertainty (PU), and project interdependence (PI) on internal knowledge integration (IKI). Results suggest that none of the hypothesized interaction paths are significant. Thus, IT usage did not moderate the influence of any of the predictor variables on internal knowledge integration. Extremely small f^2 values also support the results.

Table 4.11: Summary of Interaction Effects Analysis for IKI

Hypothesis: Path	$\mathbf{R^2}$	$\mathbf{Q^2}$	$\boldsymbol{f^2}$	**Beta**	**T-stat**
H7a: IT Usage* KH	46.6%	.29	.02	0.057	.6671
H8a: IT Usage* RC	46.5%	.29	.02	0.019	.0748
H11a: IT Usage* PU	46.5%	.29	.02	0.025	.2887
H12a: IT Usage* PI	47%	.30	.03	0.079	.6918

Combined Analysis: Overall Summary of Results

Results of this study suggest that among team-related antecedents, software teams' knowledge heterogeneity, relational capital, and sentry processes improved internal knowledge integration. Guard processes did not have a significant influence. Among project-related antecedents, project uncertainty lowered software teams' internal knowledge integration, but project interdependence had no significant influence. Regarding IT-related antecedents, team's usage of IT-based systems did not significantly moderate any of the above-mentioned relationships.

Results also suggest that most of the team-related antecedents considered in this study (knowledge heterogeneity, relational capital, and sentry processes) improved software teams' external knowledge integration, while one of them (guard processes) reduced it. Among project-related antecedents, project uncertainty reduced software teams' external knowledge integration while project interdependence improved it. Regarding IT-related antecedents, team's usage of IT-based systems positively (and very significantly) moderated all of the above-mentioned relationships. Table 4.12 summarizes these findings.

Table 4.12: Summary of Research Findings

Antecedent	**Internal Knowledge Integration**	**External Knowledge Integration**
Knowledge Heterogeneity	Positive	Positive
Relational Capital	Positive	Positive
Sentry Processes	Positive	Positive
Guard Processes	No Influence	Negative
Project Uncertainty	Negative	Negative
Project Interdependence	No Influence	Positive
IT-Usage	No Influence	Positive

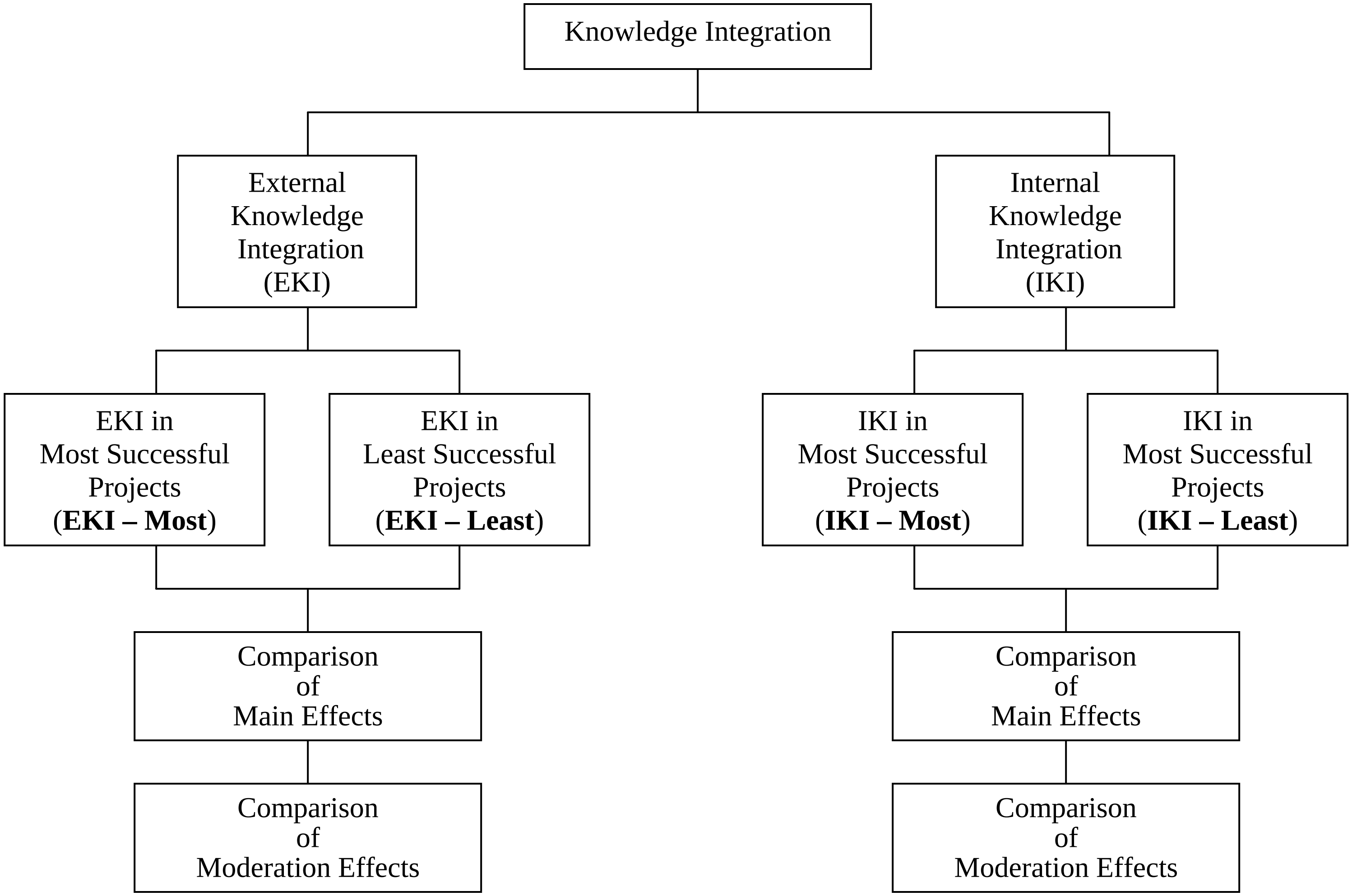
Individual Analyses: Most Successful vs. Least Successful Projects

This section discusses the results of following four analyses:

- EKI – Most: External knowledge integration (EKI) in most successful projects
- EKI – Least: External knowledge integration (EKI) in least successful projects
- IKI – Most: Internal knowledge integration (IKI) in most successful projects
- IKI – Least: Internal knowledge integration (EKI) in least successful projects

These four analyses were conducted separately as per the procedure outlined in figure 4.3. It was expected that the results of these analyses would provide interesting insights regarding knowledge integration in successful and not so successful projects.

Figure 4.3: Individual Data Analysis Procedure



Individual Analysis: EKI – Most & EKI – Least (Measurement Models)

Table 4.12 presents the composite reliabilities (ρ_c) and AVE values for the most successful projects dataset. Cronbach alphas are not presented as they were found to be the same as reported for combined analysis. Composite reliabilities are all higher than 0.8. AVE values are also higher than the recommended value of 0.5 (Chin 1998). A comparison of square root of each AVE value (bold figures on the diagonal in Table 4.12), with the correlations among constructs (the off-diagonal figures) supports the discriminant validity of constructs.

Table 4.13 presents the composite reliabilities (ρ_c) and AVE values for the least successful projects dataset. Cronbach alphas are not presented as they were found to be the same as reported for combined analysis. Most of the composite reliabilities are higher than 0.9. All AVE values are higher than 0.6, and many are higher than 0.8. Discriminant validity of constructs is supported by a comparison of the square root of each AVE value (bold figures on the diagonal in Table 4.13), with the correlations among constructs (the off-diagonal numbers).

Table 4.12: Inter Construct Correlations: Consistency and Reliability Tests (Most Successful Projects)

Construct (# of Items)	**Composite Reliability (ρ_c)**	**AVE**	**RU**	**OU**	**TU**	**EKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**ITU**	**ITF**
RU (3)	0.872	0.694	**0.831**										
OU (3)	0.944	0.848	0.476	**0.921**									
TU (3)	0.894	0.739	0.232	0.206	**0.859**								
EKI (3)	0.852	0.659	-0.231	-0.242	-0.094	**0.812**							
KH (3)	0.888	0.727	0.056	0.076	0.090	0.276	**0.853**						
RC (3)	0.861	0.674	-0.139	-0.206	0.058	0.384	0.232	**0.821**					
SP (3)	0.881	0.713	-0.205	-0.198	0.007	0.421	0.372	0.434	**0.844**				
GP (2)	0.867	0.768	-0.019	-0.088	0.043	-0.062	0.241	0.081	0.272	**0.876**			
PI (2)	0.850	0.660	-0.099	0.050	0.035	0.416	0.406	0.289	0.431	0.154	**0.812**		
ITU (3)	0.943	0.846	-0.170	-0.171	-0.164	0.482	0.191	0.298	0.295	-0.100	0.221	**0.919**	
ITF (4)	0.959	0.855	-0.171	-0.049	-0.192	0.467	0.251	0.251	0.451	-0.085	0.290	0.708	**0.925**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **EKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Table 4.13: Inter Construct Correlations: Consistency and Reliability Tests (Least Successful Projects)

Construct (# of Items)	Composite Reliability (ρ_c)	AVE	RU	OU	TU	EKI	KH	RC	SP	GP	PI	ITU	ITF
RU (3)	0.867	0.686	**0.828**										
OU (3)	0.941	0.843	0.622	**0.918**									
TU (3)	0.909	0.770	0.315	0.282	**0.877**								
EKI (3)	0.869	0.690	-0.103	-0.211	-0.099	**0.830**							
KH (3)	0.932	0.821	0.179	-0.013	-0.040	0.401	**0.906**						
RC (3)	0.873	0.696	-0.097	-0.183	0.022	0.450	0.295	**0.834**					
SP (3)	0.924	0.803	-0.089	-0.165	-0.095	0.586	0.422	0.415	**0.896**				
GP (2)	0.860	0.756	-0.107	-0.171	-0.181	0.209	0.134	0.175	0.245	**0.869**			
PI (2)	0.904	0.824	-0.115	0.335	-0.102	0.473	0.316	0.329	0.479	0.322	**0.908**		
ITU (3)	0.943	0.846	-0.208	-0.228	-0.102	0.218	0.125	0.082	0.249	0.261	0.264	**0.919**	
ITF (4)	0.959	0.855	-0.123	-0.145	0.012	0.360	0.389	0.124	0.402	0.222	0.334	0.679	**0.925**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **EKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Individual Analysis: EKI – Most & EKI – Least (Structural Models)

Table 4.14 and 4.15 present the outer model loadings of items for the two datasets. These loadings can be used to examine the convergent validity of scales. Convergent validity is shown when the t-values of the outer model loadings are above 1.96. Results suggest convergent validity of scales for both datasets.

Figures 4.4 and 4.5 present the PLS results of EKI – Most and EKI – Least analyses in a graphical form. The EKI – Most model accounts for 38.8 percent of the variance in external knowledge integration. A Q^2 of 0.1 was obtained, which suggests that the model has predictive relevance. The model for EKI – Least analysis accounts for 44.8 percent of the variance in external knowledge integration. A Q^2 of 0.21 was obtained, which suggests that the model has predictive relevance.

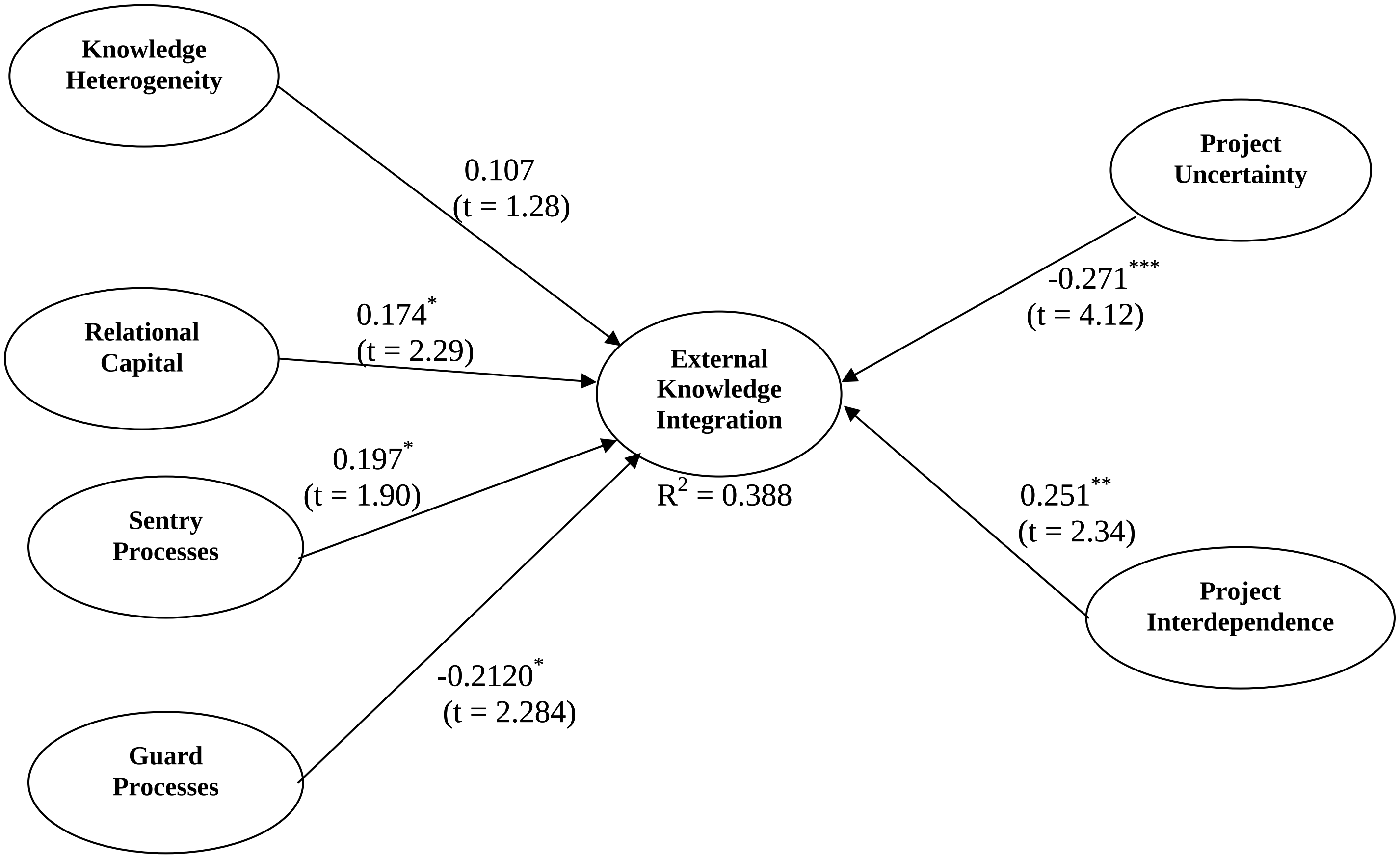
Table 4.14: Outer Model Loadings for EKI – Most Analysis

Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
External Knowledge Integration (EKI)				
EKI_1	0.8011	0.7947	0.0513	15.6290
EKI_2	0.7438	0.7511	0.0518	14.3496
EKI_3	0.8839	0.8839	0.0214	41.3782
Knowledge Heterogeneity (KH)				
KH_1	0.9060	0.9010	0.0301	30.1415
KH_2	0.8303	0.8242	0.0475	17.4785
KH_3	0.8186	0.8202	0.0510	16.0550
Relational Capital (RC)				
RC_1	0.7760	0.7575	0.0992	7.8247
RC_2	0.8033	0.7908	0.0981	8.1910
RC_3	0.8804	0.8786	0.0548	16.0550
Sentry Processes (SP)				
SP_1	0.8909	0.8866	0.0423	21.0769
SP_2	0.9003	0.9044	0.0275	32.7589
SP_3	0.7317	0.7193	0.0741	9.8686
Guard Processes (GP)				
GP_1	0.9721	0.8136	0.2691	3.6128
GP_2	0.7689	0.7754	0.2544	3.0229
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.7796	0.7863	0.0411	18.9895
RU_2	0.8847	0.8865	0.0191	46.4085
RU_3	0.8320	0.8307	0.0333	25.0049
Outcome Uncertainty (OU)				
OU_1	0.8846	0.8879	0.0271	32.6936
OU_2	0.9384	0.9418	0.0103	90.8151
OU_3	0.9386	0.9405	0.0109	86.3608
Technological Uncertainty (TU)				
TU_1	0.8043	0.7977	0.0565	14.2307
TU_2	0.8921	0.8862	0.0298	29.9449
TU_3	0.8791	0.8830	0.0264	33.3537
Project Interdependence (PI)				
PI_1	0.9005	0.9005	0.0310	29.0296
PI_2	0.9256	0.9241	0.0262	35.2676
IT Usage (ITU)				
ITU_1	0.8881	0.8845	0.0322	27.5613
ITU_2	0.9543	0.9539	0.0110	87.1466
ITU_3	0.9157	0.9152	0.0249	36.7604
IT Usage Frequency (ITF)				
ITF_1	0.9148	0.9131	0.0192	47.6080
ITF_2	0.9474	0.9450	0.0148	64.1459
ITF_3	0.9023	0.8989	0.0251	35.9153
ITF_4	0.9337	0.9290	0.0198	47.2296

Table 4.15: Outer Model Loadings for EKI – Least Analysis

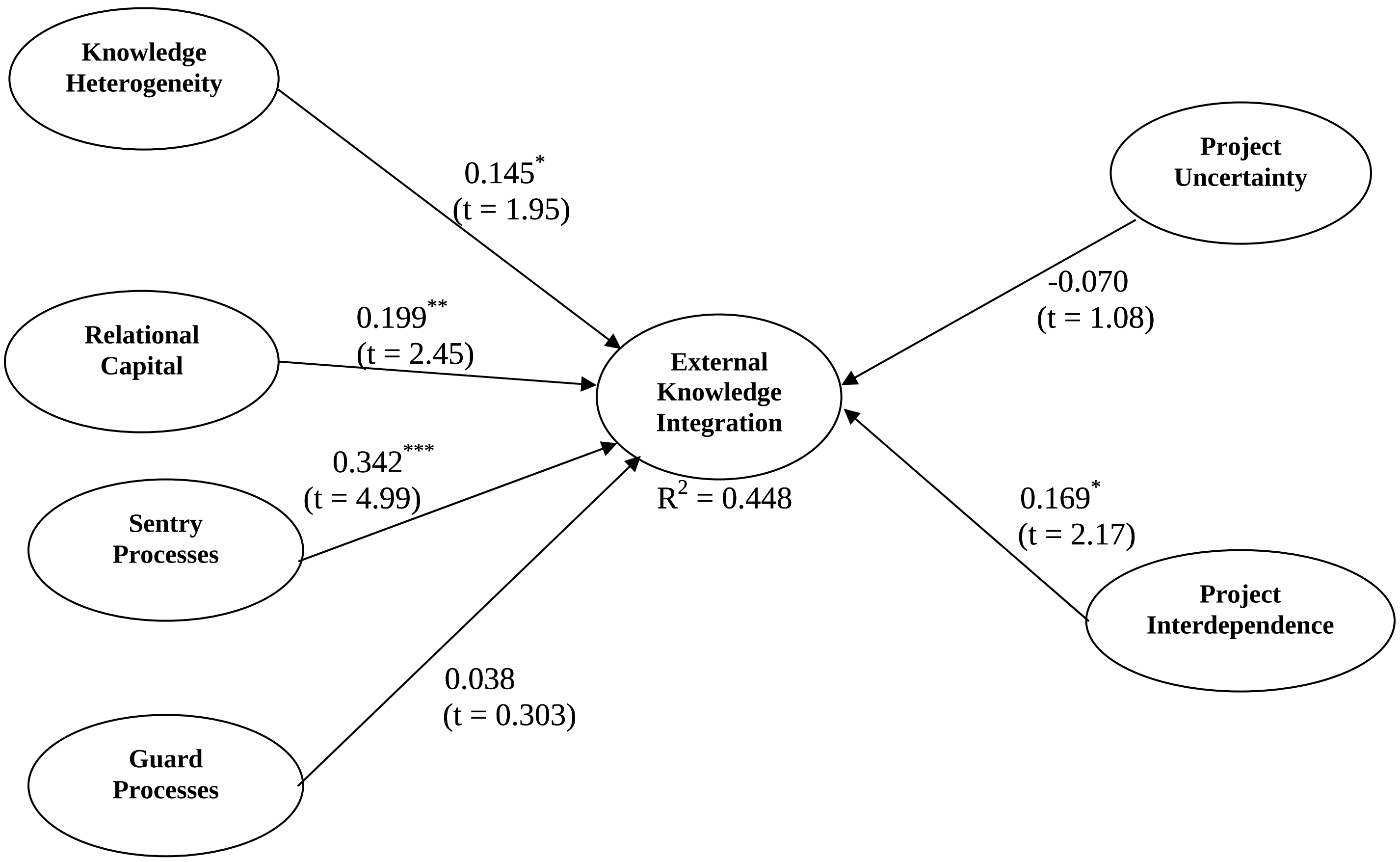
Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
External Knowledge Integration (EKI)				
EKI_1	0.8652	0.8622	0.0295	29.3593
EKI_2	0.7400	0.7250	0.0548	13.5141
EKI_3	0.8791	0.8806	0.0228	38.5430
Knowledge Heterogeneity (KH)				
KH_1	0.8996	0.8977	0.0231	38.9683
KH_2	0.9105	0.9094	0.0271	33.5937
KH_3	0.9079	0.9086	0.0202	45.0438
Relational Capital (RC)				
RC_1	0.8247	0.8094	0.0549	15.0112
RC_2	0.8726	0.8564	0.0493	17.6831
RC_3	0.8048	0.8044	0.0466	17.2555
Sentry Processes (SP)				
SP_1	0.9036	0.9403	0.0212	42.5479
SP_2	0.9410	0.9426	0.0159	59.2395
SP_3	0.8408	0.8283	0.0524	16.0309
Guard Processes (GP)				
GP_1	0.8135	0.7610	0.2216	3.6710
GP_2	0.9219	0.8731	0.1795	5.1372
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.7882	0.7903	0.0459	17.1680
RU_2	0.8760	0.8760	0.0197	44.5431
RU_3	0.8183	0.8132	0.0342	23.9244
Outcome Uncertainty (OU)				
OU_1	0.8873	0.8868	0.0227	39.0627
OU_2	0.9290	0.9301	0.0170	54.5403
OU_3	0.9368	0.9371	0.0119	78.7905
Technological Uncertainty (TU)				
TU_1	0.8150	0.8118	0.0382	21.3150
TU_2	0.9347	0.9334	0.0141	66.1963
TU_3	0.8790	0.8785	0.0239	36.8078
Project Interdependence (PI)				
PI_1	0.9433	0.9427	0.0144	65.3518
PI_2	0.8710	0.8640	0.0470	18.5371
IT Usage (ITU)				
ITU_1	0.9056	0.9006	0.0393	23.0700
ITU_2	0.9423	0.9381	0.0232	40.6662
ITU_3	0.9450	0.9426	0.0209	45.2618
IT Usage Frequency (ITF)				
ITF_1	0.9309	0.9306	0.0190	48.9841
ITF_2	0.9374	0.9375	0.0203	46.1132
ITF_3	0.9609	0.9604	0.0076	126.2547
ITF_4	0.9577	0.9567	0.0139	68.9923

Figure 4.4: Results of EKI – Most Analysis



* Significant at .05 level; ** Significant at .01 level; *** Significant at .001 level

Figure 4.5: Results of EKI – Least Analysis



* Significant at .05 level; ** Significant at .01 level; *** Significant at .001 level

Main Effects Analysis: EKI – Most & EKI – Least

Evidence suggests that knowledge heterogeneity did not have a significant influence on software teams' external knowledge integration in most successful projects ($t = 1.2835$), but it did have a significant positive influence in the least successful projects ($t = 1.946$, $p < .05$). Relational capital significantly influenced software teams' external knowledge integration in both most successful ($t = 2.297$, $p < .05$) and least successful projects ($t = 2.449$, $p < .01$), although the influence was more significant in least successful ones. Sentry processes also had a significant positive influence on external knowledge integration in both most successful ($t = 1.902$, $p < .05$) as well as least successful projects ($t = 4.991$, $p < .001$), but guard processes had a significant negative influence on external knowledge integration only in most successful projects ($t = 2.2836$, $p < .05$).

Project uncertainty was significantly related to external knowledge integration only in most successful projects ($t = 4.119$, $p < .001$). Project interdependence, on the other hand, had a significant positive influence on external knowledge integration in both most successful ($t = 2.341$, $p < .01$) as well as least successful projects ($t = 2.168$, $p < .01$).

Table 4.16 compares the results of both EKI – Most and EKI – Least main effects analyses. These outcomes are discussed in greater detail in the next chapter.

Table 4.16: Summary of Results for EKI – Most & EKI – Least Analyses

Path	EKI – Most		EKI - Least	
	Beta	**T- Statistic**	**Beta**	**T-Statistic**
Knowledge Heterogeneity → External Knowledge Integration	0.1070	1.2835	**0.145***	**1.9462**
Relational Capital → External Knowledge Integration	**0.1740***	**2.2974**	**0.199****	**2.4493**
Sentry Processes → External Knowledge Integration	**0.1970***	**1.9021**	**0.342*****	**4.9906**
Guard Processes → External Knowledge Integration	**-0.2120***	**2.2836**	0.038	0.3030
Project Uncertainty → External Knowledge Integration	**-0.2710*****	**4.1188**	-0.070	1.0822
Project Interdependence → External Knowledge Integration	**0.2510****	**2.3410**	**0.169****	**2.1685**

$^{*}p < 0.05$; $^{**}p < 0.01$; $^{***}p < 0.001$

Moderation Effects Analysis: EKI – Most & EKI – Least

The next stage of analysis involved examining the moderation effect of IT-usage in most successful and least successful projects. Models were run (separately for most successful and least successful projects) to test the interaction of the two components of IT-usage – Nature (ITU) and Frequency (ITF), with knowledge heterogeneity, relational capital, sentry processes, guard processes, project uncertainty, and project interdependence respectively. The results are presented in Table 4.17. R^2 values of all models are in excess of 40 percent. All Q^2 values are higher than 0.1 suggesting predictive relevance of the models. Also presented are the f^2 values. As mentioned earlier, these values represent the overall effect size for interaction, where 0.02, 0.15, and 0.35 are recommended as small, moderate, and large effects respectively (Cohen 1988). Most of the effects are between small and medium, but are larger than found in most past IS studies. As a mentioned earlier, a small f^2 does not necessarily connote an unimportant effect. As Chin et al. (2003) explain, “Even a small interaction can be significant under extreme moderating conditions, if the resulting beta changes are meaningful, then it is important to take these conditions into account” (p. 211).

Results suggest that all the interaction paths are significant in most successful projects. Thus, in most successful projects, IT-usage significantly moderated the influence of each predictor variable on external knowledge integration.

Results also suggest that none of the interaction paths are significant in least successful projects. Thus, in least successful projects, IT-usage did not moderate the influence of any predictor variable on external knowledge integration.

Table 4.17: Summary of Results for EKI – Most & EKI – Least Moderation Analyses

Model (Path)	**R^2**	**Q^2**	**f^2**	**Beta**	**T-stat**
EKI - Most (ITU*KH)	42.3%	.12	.06	**0.333****	**2.3907**
EKI - Most (ITF*KH)	42.1%	.13	.05	**0.309****	**2.6910**
EKI - Least (ITU*KH)	44.3%	.20	.06	0.032	0.2808
EKI - Least (ITF*KH)	44.5%	.20	.07	0.085	0.8008
EKI - Most (ITU*RC)	44%	.16	.12	**.341****	**2.9056**
EKI - Most (ITF*RC)	41.2%	.12	.04	**.245****	**2.4650**
EKI - Least (ITU*RC)	44.5%	.20	.07	.069	.6882
EKI - Least (ITF*RC)	44.8%	.20	.07	.111	1.2296
EKI - Most (ITU*SP)	41.7%	.11	.05	**.283***	**2.0509**
EKI - Most (ITF*SP)	41.2%	.12	.04	**.269****	**2.4027**
EKI - Least (ITU*SP)	44.5%	.20	.07	.088	.1047
EKI - Least (ITF*SP)	44.8%	.20	.07	.130	1.3997
EKI - Most (ITU*GP)	44%	.14	.08	**.358****	**2.7827**
EKI - Most (ITF*GP)	43.3%	.13	.07	**.337****	**3.0492**
EKI - Least (ITU*GP)	44.5%	.20	.07	.106	.7167
EKI - Least (ITF*GP)	44.7%	.20	.07	.115	1.0609
EKI - Most (ITU*PU)	41.1%	.10	.04	**.172***	**2.1302**
EKI - Most (ITF*PU)	41.2%	.11	.04	**.190****	**2.5748**
EKI - Least (ITU*PU)	44.7%	.20	.07	-.087	.6011
EKI - Least (ITF*PU)	45.2%	.21	.08	.129	1.5973
EKI - Most (ITU*PI)	41.3%	.12	.04	**.309****	**2.6820**
EKI - Most (ITF*PI)	41.4%	.12	.04	**.299****	**2.6633**
EKI - Least (ITU*PI)	44.4%	.20	.07	.079	.6953
EKI - Least (ITF*PI)	44.7%	.20	.07	.117	1.1364

$^{*}p < 0.05$; $^{**}p < 0.01$

Individual Analysis: IKI – Most & IKI – Least (Measurement Models)

Table 4.18 presents the composite reliabilities (ρ_c) and AVE values for the most successful projects dataset. Cronbach alphas are not presented as they were found to be the same as reported for combined analysis. Composite reliabilities are all higher than 0.85. All AVE values are also higher than the recommended value of 0.5 (Chin 1998). A comparison of square root of each AVE value (bold figures on the diagonal in Table 4.18), with the correlations among constructs (the off-diagonal figures) supports the discriminant validity of constructs.

Table 4.19 presents the composite reliabilities (ρ_c) and AVE values for the least successful projects dataset. Cronbach alphas are not presented as they were found to be the same as reported for combined analysis. Most of the composite reliabilities are higher than 0.9. All AVE values are higher than 0.6, and many are higher than 0.8. A comparison of square root of each AVE value (bold figures on the diagonal in Table 4.19), with the correlations among constructs (the off-diagonal figures) supports the discriminant validity of constructs.

Table 4.18: Inter Construct Correlations: Consistency and Reliability Tests (Most Successful Projects)

Construct (# of Items)	**Composite Reliability (ρ_c)**	**AVE**	**RU**	**OU**	**TU**	**IKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**ITU**	**ITF**
RU (3)	0.872	0.694	**0.831**										
OU (3)	0.944	0.848	0.476	**0.921**									
TU (3)	0.894	0.739	0.232	0.206	**0.859**								
IKI (4)	0.940	0.796	-0.227	-0.367	-0.072	**0.892**							
KH (3)	0.888	0.727	0.058	0.078	0.087	0.277	**0.853**						
RC (3)	0.861	0.674	-0.134	-0.208	0.071	0.654	0.233	**0.821**					
SP (3)	0.881	0.713	-0.207	-0.198	0.010	0.569	0.370	0.447	**0.844**				
GP (2)	0.867	0.768	-0.021	-0.134	0.034	0.145	0.223	0.184	0.270	**0.876**			
PI (2)	0.850	0.660	-0.084	0.049	0.052	0.266	0.404	0.267	0.426	0.131	**0.812**		
ITU (3)	0.943	0.846	-0.171	-0.172	-0.163	0.309	0.191	0.295	0.295	-0.049	0.204	**0.919**	
ITF (4)	0.959	0.855	-0.175	-0.051	-0.188	0.328	0.242	0.241	0.382	-0.078	0.269	0.703	**0.925**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **IKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Table 4.19: Inter Construct Correlations: Consistency and Reliability Tests (Least Successful Projects)

Construct (# of Items)	**Composite Reliability (ρ_c)**	**AVE**	**RU**	**OU**	**TU**	**IKI**	**KH**	**RC**	**SP**	**GP**	**PI**	**ITU**	**ITF**
RU (3)	0.867	0.686	**0.828**										
OU (3)	0.941	0.843	0.622	**0.918**									
TU (3)	0.909	0.770	0.315	0.282	**0.877**								
IKI (4)	0.951	0.828	-0.044	-0.203	-0.164	**0.910**							
KH (3)	0.932	0.821	0.182	-0.009	-0.038	0.353	**0.906**						
RC (3)	0.871	0.693	-0.099	-0.188	0.018	0.532	0.295	**0.832**					
SP (3)	0.923	0.801	-0.090	-0.167	-0.101	0.470	0.427	0.424	**0.895**				
GP (2)	0.854	0.746	-0.104	-0.152	-0.160	0.190	0.153	0.179	0.235	**0.864**			
PI (2)	0.903	0.823	-0.118	-0.335	-0.104	0.463	0.314	0.342	0.484	0.318	**0.907**		
ITU (3)	0.948	0.859	-0.190	-0.218	-0.089	0.187	0.120	0.090	0.249	0.275	0.268	**0.927**	
ITF (4)	0.972	0.896	-0.124	-0.146	0.020	0.274	0.392	0.134	0.405	0.230	0.333	0.683	**0.947**

Legend: **RU**: Requirements Uncertainty; **OU**: Outcome Uncertainty; **TU**: Technological Uncertainty; **IKI**: External Knowledge Integration; **KH**: Knowledge Heterogeneity; **RC**: Relational Capital; **SP**: Sentry Processes; **GP**: Guard Processes; **PI**: Project Interdependence

Individual Analysis: IKI – Most & IKI – Least (Structural Models)

Table 4.20 and 4.21 present the outer model loadings of items for the two datasets. These loadings can be used to support the examination of convergent validity of scales. Convergent validity is shown when the t-values of the outer model loadings are above 1.96. Results suggest convergent validity of scales for both datasets.

Figures 4.6 and 4.7 present the PLS results of IKI – Most and IKI – Least analyses in a graphical form. The IKI – Most model accounts for 57.2 percent of the variance in external knowledge integration. A Q^2 of 0.4 was obtained, which suggests that the model has high predictive relevance. The model for IKI – Least analysis accounts for 41.3 percent of the variance in external knowledge integration. A Q^2 of 0.2 was obtained, which suggests that the model has predictive relevance.

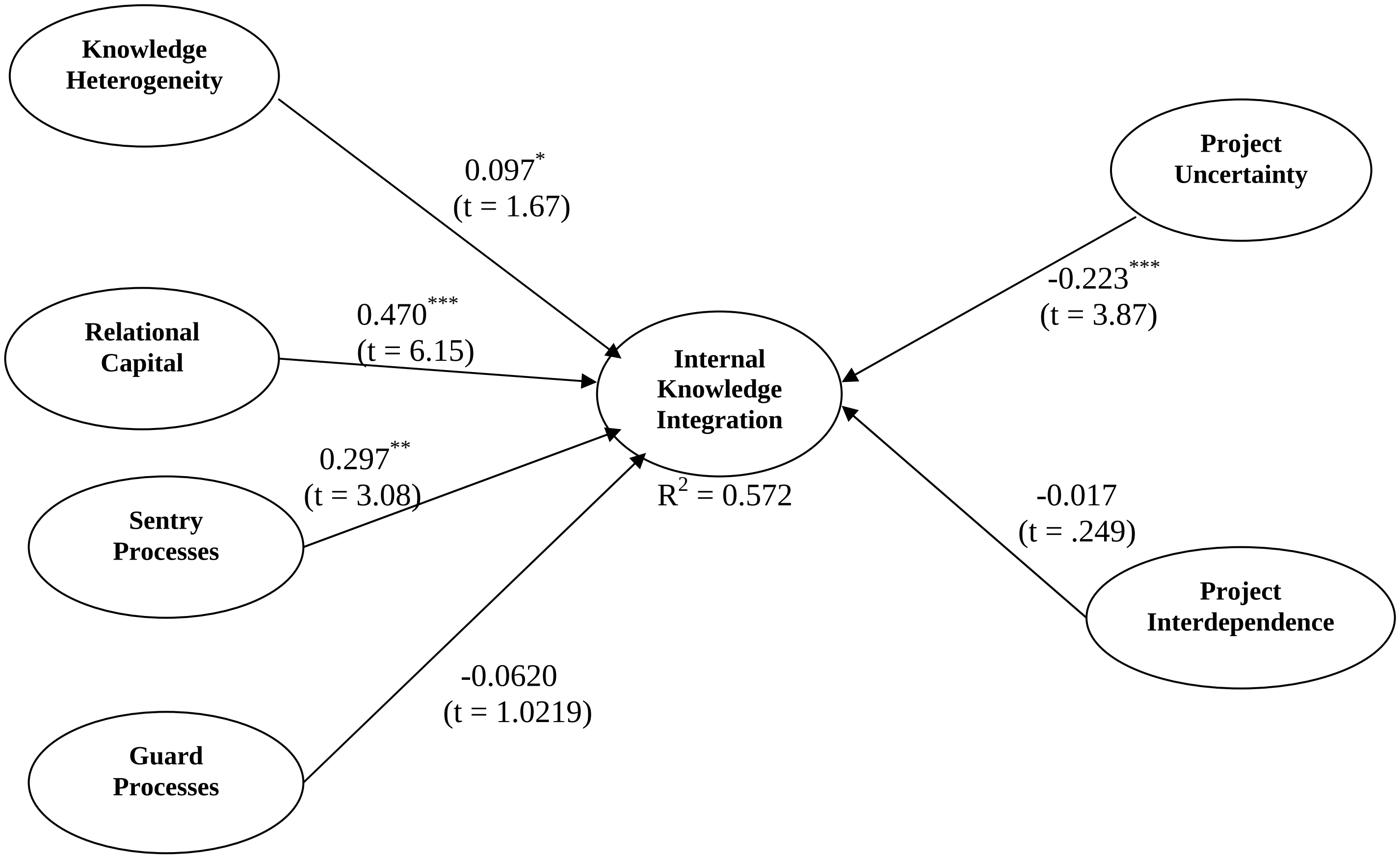
Table 4.20: Outer Model Loadings for IKI – Most Analysis

Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
External Knowledge Integration (EKI)				
IKI_1	0.8671	0.8668	0.0422	20.5391
IKI_2	0.9026	0.9003	0.0309	20.1835
IKI_3	0.8875	0.8912	0.0266	33.3428
IKI_4	0.9104	0.9132	0.0182	49.9280
Knowledge Heterogeneity (KH)				
KH_1	0.8959	0.8907	0.0326	27.5222
KH_2	0.8256	0.8148	0.0655	12.6034
KH_3	0.8304	0.8370	0.0513	16.2527
Relational Capital (RC)				
RC_1	0.8239	0.8303	0.0423	19.4796
RC_2	0.7655	0.7616	0.0857	8.9333
RC_3	0.8779	0.8813	0.0203	43.3392
Sentry Processes (SP)				
SP_1	0.8937	0.8922	0.0268	33.3081
SP_2	0.8884	0.8885	0.0358	24.8259
SP_3	0.7436	0.7394	0.0610	12.1802
Guard Processes (GP)				
GP_1	0.7971	0.7408	0.2530	3.1512
GP_2	0.9604	0.8747	0.2187	4.3907
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.7787	0.7836	0.0436	17.8428
RU_2	0.8851	0.8871	0.0189	46.8138
RU_3	0.8323	0.8283	0.0307	27.0922
Outcome Uncertainty (OU)				
OU_1	0.8848	0.8858	0.0297	29.7975
OU_2	0.9383	0.9411	0.0107	87.6701
OU_3	0.9385	0.9410	0.0117	80.5059
Technological Uncertainty (TU)				
TU_1	0.8035	0.7969	0.0638	12.5854
TU_2	0.8918	0.8879	0.0381	23.4086
TU_3	0.8800	0.8788	0.0355	24.7820
Project Interdependence (PI)				
PI_1	0.9427	0.9433	0.0314	30.0336
PI_2	0.8786	0.8657	0.0848	10.3666
IT Usage (ITU)				
ITU_1	0.8890	0.8800	0.0588	15.1063
ITU_2	0.9570	0.9533	0.0333	28.7364
ITU_3	0.9119	0.9061	0.0459	19.8737
IT Usage Frequency (ITF)				
ITF_1	0.8934	0.8849	0.0471	18.9608
ITF_2	0.9390	0.9334	0.0424	22.1432
ITF_3	0.9155	0.9099	0.0578	15.8286
ITF_4	0.9476	0.9443	0.0483	19.6133

Table 4.21: Outer Model Loadings for IKI – Least Analysis

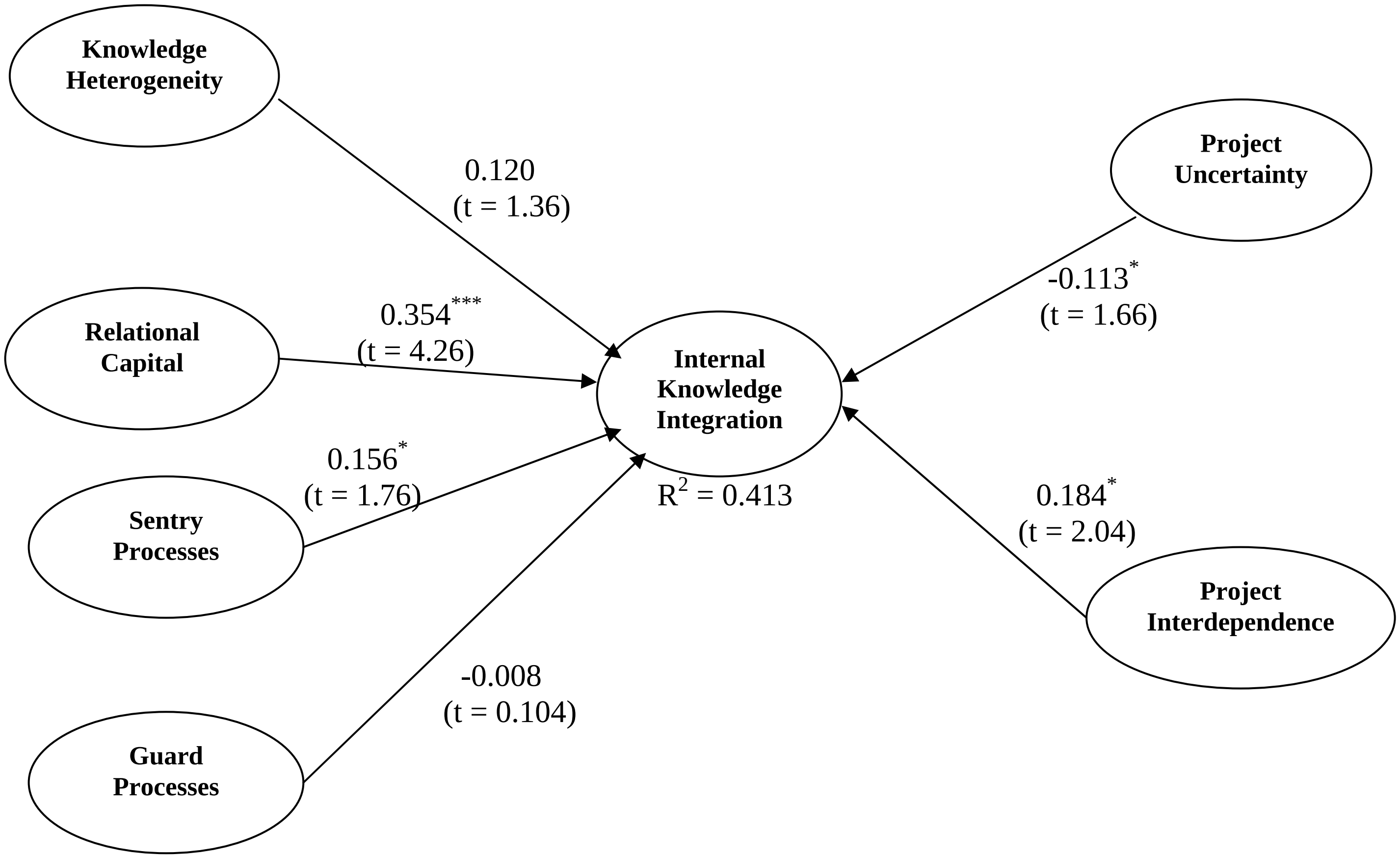
Construct Items	**Entire Sample Estimate**	**Mean of Sub-samples**	**Standard T-Statistic Error**	**T-Statistic**
External Knowledge Integration (EKI)				
IKI_1	0.9280	0.9241	0.0137	67.7660
IKI_2	0.9122	0.9083	0.0191	47.7069
IKI_3	0.8942	0.8838	0.0255	35.0983
IKI_4	0.9053	0.8984	0.0224	40.3945
Knowledge Heterogeneity (KH)				
KH_1	0.9103	0.9079	0.0235	38.8073
KH_2	0.9048	0.8988	0.0332	27.2817
KH_3	0.9024	0.9019	0.0260	34.6451
Relational Capital (RC)				
RC_1	0.8107	0.7889	0.0608	13.3263
RC_2	0.8586	0.8425	0.0584	14.7079
RC_3	0.8273	0.8365	0.0437	18.9155
Sentry Processes (SP)				
SP_1	0.9074	0.9018	0.0261	34.8179
SP_2	0.9478	0.9498	0.0110	85.8444
SP_3	0.8250	0.8116	0.0628	13.1279
Guard Processes (GP)				
GP_1	0.7746	0.7157	0.2369	3.2692
GP_2	0.9448	0.8983	0.1678	5.6316
Project Uncertainty (PU)				
Requirements Uncertainty (RU)				
RU_1	0.7871	0.7868	0.0427	18.4386
RU_2	0.8768	0.8806	0.0188	46.6718
RU_3	0.8183	0.8152	0.0338	24.2034
Outcome Uncertainty (OU)				
OU_1	0.8873	0.8874	0.0216	41.1509
OU_2	0.9290	0.9286	0.0170	54.6066
OU_3	0.9368	0.9368	0.0115	81.7673
Technological Uncertainty (TU)				
TU_1	0.8154	0.8142	0.0340	24.0118
TU_2	0.9348	0.9356	0.0127	73.4100
TU_3	0.8786	0.8815	0.0218	40.2781
Project Interdependence (PI)				
PI_1	0.9475	0.9475	0.0164	57.7672
PI_2	0.8647	0.8605	0.0475	18.1967
IT Usage (ITU)				
ITU_1	0.9145	0.8869	0.1168	7.8270
ITU_2	0.9070	0.8722	0.1204	7.5333
ITU_3	0.9576	0.9323	0.1226	7.8083
IT Usage Frequency (ITF)				
ITF_1	0.9324	0.9310	0.0188	49.4755
ITF_2	0.9357	0.9317	0.0276	33.9147
ITF_3	0.9606	0.9607	0.0083	116.3802
ITF_4	0.9581	0.9569	0.0127	75.4415

Figure 4.6: Results of IKI – Most Analysis



* Significant at .05 level; ** Significant at .01 level; *** Significant at .001 level

Figure 4.7: Results of IKI – Least Analysis



* Significant at .05 level; ** Significant at .01 level; *** Significant at .001 level

Main Effects Analysis: IKI – Most & IKI - Least

Evidence suggests that knowledge heterogeneity had a significant influence on internal knowledge integration in most successful projects (t = 1.6710, p < .05), but not in least successful projects (t = 0.0931). Relational capital significantly influenced internal knowledge integration in both most successful (t = 6.1595, p < .001) and least successful projects (t = 4.2589, p < .001). Sentry processes also had a significant positive influence on internal knowledge integration in both most successful (t = 3.0819; p < .01) as well as the least successful projects (t = 1.7598, p < .05), but guard processes did not have a significant influence on internal knowledge integration in either most successful (t = 1.0219) or least successful projects (t = 0.1040).

Project uncertainty had a significant negative influence on internal knowledge integration in both most successful (t = 3.8713, p < .001) and least successful projects (t = 1.6585, p < .05). Project interdependence, on the other hand, had a significant positive influence on internal knowledge integration only in least successful projects (t = 2.0370, p < .05).

Table 4.22 summarizes the results of the IKI – Most and IKI – Least main effects analyses. These results are discussed in greater detail in the next chapter.

Table 4.22: Summary of Results for IKI – Most & IKI – Least Analyses

Path	**IKI – Most**		**IKI - Least**	
	Beta	**T- Statistic**	**Beta**	**T-Statistic**
Knowledge Heterogeneity → Internal Knowledge Integration	**0.0970***	**1.6710**	0.1200	0.0931
Relational Capital → Internal Knowledge Integration	**0.4700*****	**6.1595**	**0.3540*****	**4.2589**
Sentry Processes → Internal Knowledge Integration	**0.2970****	**3.0819**	**0.1560***	**1.7598**
Guard Processes → Internal Knowledge Integration	-0.0620	1.0219	-0.0080	0.1040
Project Uncertainty → Internal Knowledge Integration	**-0.2230*****	**3.8713**	**-0.1130***	**1.6585**
Project Interdependence → Internal Knowledge Integration	-0.0170	0.2491	**0.1840***	**2.0370**

$^{*}p < 0.05$; $^{**}p < 0.01$; $^{***}p < 0.001$

Moderation Effects Analysis: IKI – Most & IKI - Least

The next stage of analysis involved examining the moderation effect of IT-usage in most successful and least successful projects. Models were run (separately for most successful and least successful projects) to test the interaction of the two components of IT-usage – Nature (ITU) and Frequency (ITF), with knowledge heterogeneity, relational capital, project uncertainty, and project interdependence respectively. The results are presented in Table 4.23. R^2 values of all models for most successful projects are in excess of 57 percent. Additionally, Q^2 values for these models are higher than 0.37, implying high predictive relevance of these models. R^2 values of models for least successful projects are in excess of 40 percent. Q^2 values in excess of 0.2 substantiate the predictive relevance of these models.

Also presented are the f^2 values. As mentioned earlier, these values can be used to examine the effect of the moderator, where 0.02, 0.15, and 0.35 are recommended as a gauge for small, moderate, and large effect respectively (Cohen 1988). Most of the models exhibit a zero effect of moderator. The rest have abysmally small effect size. These effects agree with the results, which suggest that none of the interaction paths are significant for either most successful or least successful projects. Thus, IT-usage does not moderate the influence of any of the predictor variables on internal knowledge integration in either most successful or least successful projects.

Table 4.23: Summary of Results for IKI – Most & IKI – Least Moderation Analyses

Model (Path)	**R^2**	**Q^2**	***f^2***	**Beta**	**T-stat**
IKI - Most (ITU*KH)	57.3%	.3789	0.002	0.072	.5898
IKI - Most (ITF*KH)	57.6%	.3865	0.009	0.112	1.2549
IKI - Least (ITU*KH)	41.3%	.2225	0	-0.019	.1766
IKI - Least (ITF*KH)	41.3%	.2199	0	0.004	.0360
IKI - Most (ITU*RC)	57.2%	.3765	0	-0.028	.2494
IKI - Most (ITF*RC)	57.2%	.3796	0	0.000	0.000
IKI - Least (ITU*RC)	41.3%	.2226	0	-0.018	.1906
IKI - Least (ITF*RC)	41.5%	.2239	0.003	0.064	.6566
IKI - Most (ITU*PU)	57.2%	.3796	0	0.021	.2470
IKI - Most (ITF*PU)	57.5%	.3830	0.007	0.065	.7533
IKI - Least (ITU*PU)	43.9%	.2520	0.04	-0.240	1.032
IKI - Least (ITF*PU)	41.7%	.2235	0.007	0.085	.5315
IKI - Most (ITU*PI)	57.2%	.3778	0	-0.040	.2930
IKI - Most (ITF*PI)	57.2%	.3811	0	0.036	.3814
IKI - Least (ITU*PI)	41.3%	.2222	0	-0.021	.2088
IKI - Least (ITF*PI)	41.3%	.2223	0	0.044	.4139

Summary

In this chapter, results were presented from two different analyses of the data. First analysis involved examining the combined dataset of 300 projects (150 most successful and 150 least successful) to test various research hypotheses. Results from this analysis supported a number of hypotheses.

The second analysis involved testing the predicted relationships separately in most successful and least successful projects. Some interesting results were obtained from this analysis. These results, the results of the combined analysis, and their implications are discussed in the next chapter.

CHAPTER 5: DISCUSSION & CONCLUSIONS

In order to perform well, project teams, such as software development teams, must be able to leverage knowledge from multiple sources. This includes integrating the specialized knowledge of members (Walz et al. 1993), with knowledge from external sources (Nonaka et al. 1995). There has been little research on what factors influence the internal and external knowledge integration in software teams as they are engaged in achieving project goals.

In this study, it was argued that knowledge integration in software teams is influenced by three categories of antecedents. The first category includes team-related issues, such as the heterogeneity of team's internal knowledge resources, team's relational capital, and its boundary-buffering processes. The second category of antecedents includes project characteristics such as uncertainty and interdependence. And, the third category of antecedents pertains to the usage of various information technology (IT) based systems. The research framework for this study was designed to examine:

1. The influence of team antecedents (knowledge heterogeneity, relational capital, and boundary-buffering processes) on teams' internal and external knowledge integration
2. The influence of project antecedents (uncertainty and interdependence) on teams' internal and external knowledge integration

3. The moderating influence of teams' IT-usage on its internal and external knowledge integration

It was proposed that:

1. Team antecedents improve knowledge integration, except for the boundary-buffering processes, which impair external knowledge integration
2. Among the project antecedents - uncertainty improves both internal and external knowledge integration. Interdependence improves external integration, but, impairs internal knowledge integration
3. IT-usage moderates the influence of team and project antecedents on both internal as well as external knowledge integration. More specifically, both internal and external knowledge integration improve at higher levels of IT-usage.

Empirical results presented in the previous chapter suggest interesting insights into these propositions. In this chapter, the implications of those results are discussed. Also discussed are key theoretical and managerial contributions, and the limitations of this study. The chapter concludes with the discussion of opportunities created by this study for future research.

Discussion of Findings

Knowledge integration is a critical process in the software teams (Walz et al. 1993). This study was conducted to examine the dependencies of various team-, project-, and IT-related antecedents with the integration of internal and external knowledge in software teams.

In light of the research framework, findings are discussed first for the combined analysis of 300 projects. A key insight gained from the results, and that would be helpful in discussing their implications, is that teams develop an infrastructure for integrating knowledge - internal as well as external. At the risk of oversimplification, one may suggest that this infrastructure includes cognitive elements (e.g., shared context between the team members), relational elements (e.g., trust, close interpersonal relationships, and reciprocal behavior), procedural elements (e.g., inter- and intra-team communications, and decision making processes), and technological elements (e.g., IT-based systems). It is in light of this knowledge integration infrastructure that the influence of the three categories of antecedents (team-, project-, and IT-related) to knowledge integration can be explained. In the ensuing sections, this issue is discussed in more granularity.

Knowledge Heterogeneity, Internal Knowledge Integration and IT-Usage

The results of this study suggest that a software teams' knowledge heterogeneity improves its internal knowledge integration. In developing the main effects hypothesis, two conflicting perspectives relating knowledge heterogeneity to the internal knowledge integration were discussed. First perspective suggests a positive relationship between knowledge heterogeneity and internal knowledge integration (Lewis 2004; Nahapiet et al. 1998), while the second proposed no, or even negative, relationship (Rulke et al. 2000; Tiwana et al. 2005). This study argued in favor of a positive relationship, as also supported by the results. Thus, it appears that the internal knowledge integration in heterogeneous software teams gains from the diversity of expertise and skills among their members. Heterogeneous teams have large knowledge bases, which not only improves the quality of inputs to the knowledge integration process (Smith et al. 2005), but also allows teams more variety of approaches to integrate their internal knowledge resources. Diverse members also bring in a better understanding of project's knowledge requirements, and a more elaborate schema of how to apply their knowledge to the project (Fiske et al. 1983; Lord et al. 1990). This might influence some of the procedural elements (e.g., improved decision-making) and cognitive elements (e.g., better shared context) of the team's knowledge integration infrastructure.

The results of IT moderation suggest that the technological elements of the knowledge integration infrastructure will not enhance this influence. It was argued that knowledge integration in heterogeneous teams may require interactive and intensive communication, and collaborative systems may streamline intra-team communications. It was also proposed that KM systems would enhance this process by helping members interpret and assimilate internal as well as external knowledge inputs. The results, by contradicting my arguments, suggest an interesting support for the *contingency theory*, as per which, an interactive multi-person communication medium, such as person-to-person communication, is preferable over IT-based communications in situations involving multiple perspectives, and information that may be liable to multiple interpretations (Galeghar et al. 1994; Lawrence et al. 1986). Teams with diverse knowledge resources do face such situations. Internal knowledge integration in such teams requires building a consensus among the members on project's knowledge requirements and how best to fulfill them. Computer-mediated groups have more difficulty building this consensus, as they are less likely to reach an agreement, compared to groups engaged in person-to-person communications (Hiltz et al. 1986; Struas et al. 1994). Thus, project leaders supervising heterogeneous software teams may benefit from developing a regimen of holding regular meetings of the team members to discuss various project-related issues.

Knowledge Heterogeneity, External Knowledge Integration and IT-Usage

The results imply that the diversity of expertise in heterogeneous teams not only improves their access to the external knowledge sources across multiple domains, but also increases their capacity to integrate valuable knowledge inputs from these sources (Cohen et al. 1990). These characteristics may positively influence some of the procedural elements of the team's knowledge integration infrastructure. Technological elements will enhance this influence. The experts in such teams use IT-based systems not just to collaborate and combine their expertise but also to absorb more appropriate knowledge inputs from external sources and integrate them with their combined expertise to develop a more robust body of project-level knowledge.

Relational Capital, Internal Knowledge Integration and IT-Usage

The results imply that teams with members sharing high levels of mutual trust, close working relationships, and reciprocal behavior will better integrate their internal knowledge resources. This makes sense because high level of relational capital in a team makes the members more conducive to integrating their unique knowledge resources as they:

- Are more open to sharing their unique knowledge resources with one another
- Interact more frequently and communicate more effectively
- Are confident that their acts of open exchange of knowledge will be reciprocated

This may have a positive influence on some of the cognitive and procedural elements of the team's knowledge integration infrastructure. The results of IT moderation imply that technological elements of the team's knowledge integration infrastructure will not enhance this influence. The results, although contradict the proposed hypothesis, support the theories of social capital, which suggest that relational characteristics such as commitment and trust are difficult to develop in computer networks (Nahapiet et al. 1998; Nohria et al. 1992). Thus, teams with high levels of relational capital are less expected to use IT-based systems for inter-personal communications (Daft et al. 1987). Such teams are also less likely to prefer IT-based systems for integrating specialized knowledge inputs of the members to create systemic project-level knowledge (Trevino et al. 1987).

Relational Capital, External Knowledge Integration and IT-Usage

Findings suggest that relational capital in software teams is an important predictor of teams' integration of external knowledge inputs, and IT-usage strongly increases this influence. Two possible explanations can be forwarded to support this conclusion. First, members of teams with high level of relational capital trust each other's expertise, which seems to motivate them to gather and integrate relevant knowledge even from the external sources (Tiwana et al. 2005). Thus, high level of relational capital will have a positive influence on some of the cognitive, relational, and procedural elements of teams' knowledge integration infrastructure. Additionally, as suggested by the results of IT moderation, this influence is enhanced if the teams use the technological elements of their knowledge integration infrastructure to create efficiencies of information aggregation.

Second, frequent interactions and communication among team members, resulting from high levels of mutual trust and close working relationships, influence the cognitive elements of the knowledge integration infrastructure, thus helping team members develop a shared understanding of issues such as: What are project's knowledge requirements and who has access to external sources of relevant project-related knowledge (Ko et al. 2005). A clear understanding of these issues streamlines the teams' acquisition and subsequent integration of both explicit as well as tacit knowledge from external sources (Marsden 1990), which may subsequently influence some of the procedural elements of the knowledge integration infrastructure. Once again, teams can use the technological elements of the knowledge integration infrastructure to support the acquisition of both explicit and tacit knowledge inputs, and subsequent integration of explicit inputs. Tacit knowledge inputs will typically be better integrated in person-to-person settings.

Sentry Processes and Internal Knowledge Integration

Results suggest that sentry processes help the team members better integrate their internal knowledge resources. This may happen for two reasons. First, sentry processes allow the team members to work with minimal external distraction, saving their time and energies, which are better utilized in trying to integrate their unique expertise and skills in innovative ways to achieve project goals (Yan et al. 1999). This may have a positive influence on some of the cognitive elements of the knowledge integration infrastructure. Second, by screening out undesired information inputs from the external sources, sentry processes free up teams' information-processing capacities, which can be employed to better integrate teams’ internal knowledge resources (Jemison 1984). This may positively influence some of the procedural elements of the knowledge integration infrastructure.

Sentry Processes, External Knowledge Integration and IT-Usage

This study’s findings regarding the influence of teams’ sentry processes on their external knowledge integration have interesting implications. Previous literature in this field argues that the organizational units (e.g., teams) buffering their boundaries through sentry processes have to incur certain costs (Aldrich et al. 1977). For example, members of the teams performing sentry processes may not have an open access to external knowledge sources, and thus, may not be aware of the project-relevant knowledge held by those sources. This may depress teams’ knowledge acquisition from external sources (Awazu 2004), and therefore have a negative influence on their knowledge integration from those sources.

On the contrary, the results of this study suggest that the external knowledge integration benefits from sentry processes. This may be possible because teams performing sentry processes typically assign a ‘gatekeeping’ role to one or more of their members. Gate-keeping role involves monitoring and streamlining the flow of information from external sources. Typically, the gatekeepers are strongly connected to the external sources of information (Tushman 1977), and are responsible for bringing-in relevant updated information from those sources. Past studies have even reported of specialized gate-keeping roles that acquire external information in separate domains (Allen et al. 1969; Walsh et al. 1972), which might be the case in software teams, as these teams rarely have all the technical and functional knowledge required to execute a project. By clearly assigning the gatekeeping roles, software teams may streamline their external knowledge acquisition function. In the absence of a particular knowledge resource available internally, the team members will typically ask the gatekeeper to acquire that knowledge from external sources. This clarity has a positive influence on some of the cognitive and procedural elements of the teams' knowledge integration infrastructure. The boundary-buffering roles also streamline teams' communication with other teams, thus influencing some of the procedural elements of their knowledge integration infrastructure.

Results of IT-usage moderation suggest that the technological elements of the software teams' knowledge integration infrastructure, if used appropriately, might have the capability to enhance the positive influence of sentry processes on the external knowledge integration. Gatekeepers using technological elements can create efficiencies related to information aggregation by (1) improving the availability of internal and external knowledge inputs that can be integrated, (2) improving the quality of inputs integrated, and (3) decreasing the teams' cost (time and effort) of integration (Malone et al. 1987).

Guard Processes and Internal Knowledge Integration

Results of this study imply that guard processes did not predict internal knowledge integration in software teams. This result can be discussed as follows: Guard processes include some typical activities. One such activity is classifying external requests for information and resources in terms of their legitimacy and cost to the team (Ancona et al. 1988). A subsequent activity includes controlling the release of information or resources when the request does not seem legitimate, or is expected to incur significant cost to the team. Another simultaneous activity involves delivering the requested information or resources if the external request seems legitimate and reasonable. It seems that one reason why guard processes have no significant influence on the team's internal knowledge integration is because team members may have clear roles and responsibilities (e.g., project managers) to execute various guard activities. Members with these roles may attend to the external requests, thereby not disturbing rest of the team. Thus, team's other functions, such as internal knowledge integration, continue unaffected. In other words, the roles protect the team's internal knowledge integration infrastructure from external disturbance.

Guard Processes, External Knowledge Integration and IT-Usage

Contrary to the results for internal knowledge integration, guard processes had a significant negative influence on the external knowledge integration. Interesting conclusions can be drawn from these findings. Team members are typically less willing to share their knowledge with members of other teams (Blau 1964). This may be because of less frequent interactions among the teams, lower levels of interpersonal trust, or because of a perception that sharing knowledge results in reduced status and low personal worth (Nahapiet et al. 1998; Orlikowski 1996). Whatever be the reason, individuals are less willing to engage in knowledge sharing across the team boundaries, and if they do, they are more likely to expect reciprocal behavior as compared to knowledge sharing within their own team. Guard processes complicate this situation. Teams performing guard processes may earn a negative reputation of being fastidious and non-cooperative, and elicit a similar response when they request knowledge inputs from other sources. This may have a negative influence on some of the procedural elements of teams' knowledge integration infrastructure (e.g., inter-team communications), which, in addition to a dearth of good quality inputs from external knowledge sources, may deteriorate teams' external knowledge integration. However, the technological elements of the teams' knowledge integration infrastructure, if used adequately, have the capability to offset this negative influence. Members of teams performing guard processes can still use electronic knowledge repositories to access external knowledge inputs. Additionally, members can bypass the teams' guard processes by using systems such as bulletin boards, list serves, and chat rooms to connect with to connect with knowledge workers in their field who are globally dispersed and typically strangers (Teigland et al. 2003).

Project Uncertainty, Internal Knowledge Integration and IT-Usage

Results of this study suggest that a software team's capability to integrate its internal knowledge deteriorates as project uncertainty increases. Uncertainty is typically defined as the absence of critical information (Tushman et al. 1978; Zmud 1980). Thus, gaining additional information, and processing it to gain more clarity, is typically said to reduce uncertainty (Kydd 1989). The negative influence of uncertainty on the team's internal knowledge integration can be explained by uncertainty's impact on some of the cognitive and relational elements of the team's knowledge integration infrastructure (e.g., mutual trust, interpersonal relations, and shared context among team members). This is how it seems to happen: the ambiguity inherent in uncertain projects may prevent a clear understanding among the team members about project-related issues, and the team may have to make frequent changes in role allocations, schedules, and priorities (Galbraith 1973; Van de Ven et al. 1976). These changes may prevent the formation of a shared context among the members. The situation may be aggravated by multiple connotations of various project-related issues among the team members. Additionally, differences of opinion among the members may become more pronounced in situations involving uncertainty. As Kraut and Streeter explain: "Software is uncertain because different subgroups involved in its development often have different beliefs about what it should do and how it should do it…..While analysts may try to adopt the point of view of the software's users, designers and programmers often have a more technical focus, with an emphasis on ease of development and efficiency of computation. As more groups become involved in software development, disagreements among them inevitably increase" (1995: 70).

Thus, by interfering with some of the cognitive and relational elements of the team's knowledge integration infrastructure, high project uncertainty challenges the team's capability to synthesize its internal knowledge resources to develop systemic project-level knowledge, and to use that knowledge to achieve project goals.

The results of IT moderation suggest that the technological elements of the team's knowledge integration infrastructure did not moderate the negative influence of project uncertainty on internal knowledge integration. The results, although contrary to my hypothesis, align with the theory of information richness, which argues that situations involving multiple and conflicting viewpoints require communication through information-rich channels (Daft et al. 1986; Daft et al. 1987; Struas et al. 1994). Thus, situations involving high uncertainty may benefit more from frequent person-to-person communications (Kim 1988). Another possibility is that such teams clearly define individual roles right in the beginning of the project, and then refrain from very frequent interactions, whether IT-based or personal. As Zmud discusses among one of his management guidelines to deal with project uncertainty: “If tasks are defined so as to minimize the need for participant interaction, the smaller the likelihood that misunderstanding between participants will occur. Thus *task independence* simplifies the development effort” (1980: 47).

Project Uncertainty, External Knowledge Integration and IT-Usage

Except for a strong positive moderation of IT-usage, which is discussed later, results here are similar to those for the previous section (project uncertainty & internal knowledge integration). The results can be explained by uncertainty’s impact on some of the procedural elements of the teams’ knowledge integration infrastructure. In uncertain projects, the instability of various project-related issues (e.g., requirements specifications) necessitates the software teams to keep themselves informed of the most recent updates on those issues (Zmud 1980). Thus, teams needs to regularly absorb knowledge inputs from the external environment (Curtis et al. 1988; Kraut et al. 1995), which compels them to initiate horizontal coordination with the external sources (Andres et al. 2001; Galeghar et al. 1994). Additionally, in their efforts to reduce the unpredictability in uncertain projects, horizontal coordination may be required with other teams having prior experience of executing similar projects, or to seek inputs from experts in relevant technical and functional domains. The teams working on uncertain projects may also need to coordinate vertically with other groups, such as the top management, to obtain slack knowledge resources (Galbraith 1977). This portfolio of vertical as well as horizontal coordination may overwhelm teams' communication as well as its decision-making processes (Argote 1982). By interfering with these procedural elements of the teams' knowledge integration infrastructure, high project uncertainty may reduce the teams' capability to better integrate external knowledge inputs.

Results of IT moderation suggest that the technological elements of teams' knowledge integration infrastructure, if used adequately, have the potential to assuage the negative influence of project uncertainty on external knowledge integration. These findings are interesting, especially in light of the fact that a similar moderation was not supported for internal knowledge integration. A valid question then is: How do technological elements moderate the influence of project uncertainty on external integration, but not on internal integration? A possible explanation lies in the role played by technological elements in facilitating the teams' horizontal coordination. For example, team members can use KM systems such as electronic knowledge repositories (to search for documented knowledge), expert directories (to search for external experts in the field), and electronic discussion forums (to seek feedback in their communities of practice). Technological elements may also facilitate teams' vertical coordination. In the absence of frequent person-to-person interactions, teams' usage of collaborative systems, such as telephone and e-mail, may assist them in this process (Lee et al. 1999/2000). Telephone is a synchronous medium with high bandwidth that can facilitate interactive communication, and e-mail can be used to exchange large amounts of factual content.

Project Interdependence, Internal Knowledge Integration and IT-Usage

It was hypothesized that high project interdependence, as characterized by task and outcome interdependence, lowers the intrinsic motivation of team members, making them lose interest in activities that improve team performance (such as internal knowledge integration) (Andres et al. 2001). However, results did not support this expectation, as a significant relationship was not observed between project interdependence and internal knowledge integration.

A possible explanation for the absence of any significant relationship between project uncertainty and internal knowledge integration is that interdependence does not influence any of the elements of teams' knowledge integration infrastructure. This is contrary to some of the previous studies, which report that interdependence may have a negative influence on team members' satisfaction (Andres et al. 2001). This point of view was adopted while developing the hypothesis. It was argued that interdependent teams might not get enough time to develop the cognitive and relational elements of teams' knowledge integration infrastructure. Apparently, this is not the case.

An alternative explanation, that also supports the results of this study, is as follows: successful execution of interdependent projects typically requires extensive expertise and skills distributed across multiple cross-functional teams (Andres et al. 2001). Thus, teams working on interdependent projects need to integrate their knowledge more with each other, than inside themselves. This, of course, does not mean that interdependent teams do not engage in internal knowledge integration at all. They probably pursue it as typical software teams do, but project interdependence has no influence on their efforts. Evidence forwarded by the moderation analysis also supports this explanation, as teams' usage of IT-based systems did not moderate the relationship of project interdependence (or rather lack of it) with the internal knowledge integration.

Project Interdependence, External Knowledge Integration and IT-Usage

Results imply that in software teams, project interdependence is a strong predictor of external knowledge integration. This finding can be interpreted in light of the influence of project interdependence on some of the procedural elements of teams' knowledge integration infrastructure. Interdependent teams need to use some of the procedural elements more than the teams working on standalone projects. For example, interdependent teams engage in extensive inter-team coordination (to sequence, schedule, and synchronize their respective tasks) and communication (to share project-related information) (Andres et al. 2001; Struas et al. 1994). Inter-team coordination and communication enables the teams to integrate knowledge inputs, which might improve their external knowledge integration.

Additionally, the expertise and skills required to complete interdependent projects are typically distributed over multiple cross-functional teams, and need to be integrated to achieve project outcomes. Findings of IT moderation suggest that the teams use IT-based systems for this purpose. Using IT-based systems in interdependent projects reduces the cognitive information processing costs of the participating teams (Jarvenpaa et al. 2000). Also, person-to-person interactions may be difficult among the members of interdependent teams, so they rely on IT-based systems for exchanging knowledge inputs.

Interesting Findings from Separate Analyses of Most & Least Successful Projects

In this section, some interesting findings from the separate analyses of the most successful and the least successful projects are discussed. Discussion is in light of the fact that although projects have been categorized as most and least successful, they were both completed.

Knowledge Heterogeneity

In the most successful projects, teams' knowledge heterogeneity had a significant positive influence on internal knowledge integration, but the relationship was not significant in the least successful projects. It appears that the most successful teams, as compared to the least successful ones, were able to benefit from the diversity of their knowledge resources by integrating those resources in a better manner.

Results were different for external knowledge integration. Knowledge heterogeneity, in moderation with IT-usage, improved external knowledge integration in the most successful projects. But surprisingly, knowledge heterogeneity had a significant positive influence on external knowledge integration in the least successful projects too. As a possible explanation, it seems that heterogeneous teams, although working on less successful projects, would still have better access to external knowledge inputs in diverse domains, and better capacity to integrate those inputs, as compared to homogeneous teams. But, as suggested by the non-significance of IT moderation in the least successful projects, these teams did not support their external knowledge integration by using IT-based systems.

Relational Capital

Relational capital had a highly significant positive influence on knowledge integration in both the most successful and the least successful projects. It appears that both internal as well as external knowledge integration in the most as well as least successful teams benefited from mutual trust, close working relations, and reciprocal behavior among members. Findings also suggest that the most successful teams enhanced this benefit by using IT-based systems.

Sentry Processes

Results suggest that sentry processes improve internal knowledge integration even in the least successful projects, although in such projects, they are a weaker predictor of internal knowledge integration, as compared to the most successful projects. This seems appropriate because the teams performing sentry processes, even if they are working on less successful projects, could perform internal knowledge integration better as compared to teams not performing sentry processes. A possible reason, for example, is that such teams have more of their information-processing capacities available for knowledge integration as compared to the teams not performing sentry processes.

Sentry processes also emerged as a key predictor of external knowledge integration in both the least as well as the most successful projects. Surprisingly, the significance is stronger in the least successful projects, which may hint towards the presence of some factors in those projects (e.g., higher complexity or ambiguity) that required the teams to boost up their sentry processes, and thus their external knowledge integration. But, as the results of moderation analysis suggest, these teams did not use IT-based systems significantly enough to improve their external knowledge integration. On the other hand, teams working on the most successful projects did further improve their external knowledge integration by using IT-based systems.

Guard Processes

In the separate analyses of the most successful and the least successful projects, guard processes had a significant negative relationship with external knowledge integration in the most successful projects. IT-usage strongly nullified this negative influence of guard processes.

It may be possible that the teams working on less successful projects didn’t perform guard activities at all, nor did they use IT-based systems to access and integrate knowledge inputs from external sources. On the other hand, teams working on the most successful projects did both.

Project Uncertainty

An interesting finding was observed regarding the influence of uncertainty on external knowledge integration. One would expect project uncertainty to have a negative influence on external knowledge integration in the least successful projects. But surprisingly, project uncertainty had a significant negative influence on external knowledge integration only in the most successful projects.

This can be explained by revisiting our discussion of combined analysis results. It was discussed that project uncertainty reduces external knowledge integration by interfering with some of the procedural elements of teams’ knowledge integration structure. It seems that as compared to the most successful projects, the least successful projects did not possess an elaborate knowledge integration structure in the first place. So uncertainty did not have any influence on external knowledge integration in these projects.

What differentiated the most successful projects from the least successful ones was their usage of IT-based systems. IT-usage nullified the negative influence of project uncertainty on external knowledge integration in the most successful projects.

Project Interdependence

Project interdependence had a significant positive influence on internal knowledge integration in only the least successful projects. It seems that the least successful teams increased their internal knowledge integration as an effort to cope with the requirements of interdependence (although to no avail).

Project interdependence also appears to be a key predictor of external knowledge integration in both the most as well as the least successful projects. However, findings from moderation analysis suggest that only the most successful teams used IT-based systems significantly enough to further improve their external knowledge integration.

Research Contributions

The study makes some novel contributions to the IS and KM literatures. It is one of the first studies to develop and test a model of knowledge integration at the team-level. Compared to earlier studies in the field (e.g., Tiwana et al. 2005), this study includes a more robust conception of knowledge integration by considering both internal and external components. Another difference is the examination of project- and IT-related antecedents to knowledge integration. Thus, in developing and testing a model of knowledge integration that includes team-, project-, and IT-related antecedents, this study *integrates knowledge* from literatures in information systems, management science, psychology, and knowledge management. The results of this study interlink these fields.

Another unique feature of this study includes the multiple levels of data analyses. Data of 300 projects was first analyzed for the proposed main effects and IT moderation effects. Dataset was then split into two categories each of 150 “most successful” and “least successful” projects, and the main effects as well as IT moderation effects were again examined in the light of moderating effect of project success. Interesting findings emerged from these analyses, which add a new level of understanding to our knowledge of how the team-, project-, and IT-related antecedents influence internal and external knowledge integration in software teams. As an illustration, based on the findings of combined analysis of 300 projects, it was expected that project uncertainty would have a negative influence on knowledge integration only in least successful projects. It was assumed that most successful teams would have better managed project uncertainty, and thus the relationship between uncertainty and knowledge integration will be positive, or at least insignificant. Surprisingly, uncertainty had a negative influence on knowledge integration in most successful projects. More surprisingly, uncertainty had no influence on external knowledge integration in *least successful projects*! Such findings demanded an explanation that goes beyond the scope of existing literature. The explanation that was offered is the next contribution of this study. It includes the conception of a “knowledge integration infrastructure” in teams. The study proposes that teams develop an infrastructure to integrate their internal and external knowledge resources. The study also identifies various elements of this structure (cognitive, relational, procedural, and technological). The conception of this structure, which has seldom been reported in previous studies, helps us graduate to the next level in our pursuit of developing a robust framework of knowledge integration.

The last contribution of this study is the examination of role of IT-usage in knowledge integration. Such detailed examination of IT has seldom been conducted in prior studies on knowledge integration. The results of this examination improve our understanding of how IT interacts with other variables to improve only external knowledge integration. Results also suggest that the purpose to study the influence of IT-usage is better served when it is examined as a moderator. For example, the results of moderation analyses bring out a clear difference in the influence of IT-usage on internal versus external knowledge integration. Although, the lack of influence of IT on internal knowledge integration remains an area that warrants future investigation. It may be possible that the absence of a significant relationship is a result of the way in which IT-usage was measured.

Managerial Contributions

Results of this study provide appealing advice for practitioners, especially those in team-based organizations. First, project leaders should aim at developing a knowledge integration infrastructure. A robust infrastructure is characterized by: (1) high level of mutual trust, close working relationships, and knowledge sharing behavior among the team members; and (2) clear roles for intra- and inter-team communications and team decision-making. Thus, the time and effort expended on these two issues is well spent, unless team shares interdependencies with other teams, when the first issue is not that critical.

Second, deploying IT resources, especially collaborative and KM systems, and informing the team members about their respective advantages will buttress the knowledge integration structure. Project leaders may also want to encourage more person-to-person interactions, than IT-based communications, among team members.

Third, project leaders of the teams working on uncertain projects may face relational issues such as low trust levels and conflicts among team members. While addressing these issues, project leaders may want to remember that these issues may be the result of a general confusion emanating from project uncertainty.

Finally, teams should be structured keeping in mind that the diversity of expertise and skills among members will improve the quality of internal as well as external inputs to the project-level knowledge. Diverse teams are also more likely to have a better understanding of their project's knowledge requirements, and they adopt innovative approaches to synthesize available knowledge resources to best achieve project goals.

Limitations of the Study

Certain limitations of this study need to be noted. First, generalizability of results is a possible limitation. The fact that the research was conducted on software development teams may limit the results preventing their generalization beyond that scope. There is also a cultural issue attached to the study, as the data was collected in Indian software companies. However, this study argues that the conceptual model applies to most knowledge-intensive firms with team-based structures, and the methodology used is sound and replicable to another set of organizational characteristics. The global software industry is knowledge-intensive in nature, and it is plausible that the results of this study may apply to other knowledge-intensive industries or firms. Caution must still be exercised while generalizing the results to a range of team-based organizations operating in varied contexts.

Second, the study did not cover all team-, project-, and IT-related antecedents to knowledge integration. Only some of these antecedents were focused upon. Third, the use of only perceptual survey measures for data collection might increase the risk of common-method bias. Additionally, though the survey method is useful for identifying various sets of relationships; it does not inform us why these relationships exist. This limits our ability to draw causal references.

Future Research Opportunities

The limitations of this study open the door to future research. Findings of this study will be more generalizable if future research can replicate the research framework across other settings and over time. Another limitation concerns the inability of survey method to address causality. Future research efforts, especially those including in-depth case studies, would make a valuable contribution in expanding our comprehension of team-level knowledge integration.

This study also integrates previous academic pursuits in information systems, management science, psychology, and knowledge management. It is expected that the findings of this study will motivate diverse minds, providing them a foundation for future inter-disciplinary research, especially for developing a robust knowledge integration framework. The conception of a team-level knowledge integration structure provides a starting point for research in this domain. More exhaustive team-level models need to be developed and tested before we can fully comprehend the true nature of such a structure. Two interesting streams of future research are identified towards this end. One stream of research, as illustrated in Figure 5.1, involves a detailed testing of the validity of knowledge integration infrastructure itself. Another stream of research, as summarized in Figure 5.2, focuses on testing various relationships proposed in earlier sections. For example, future studies can examine the influence of teams’ boundary-buffering roles (e.g., gatekeeper) on the cognitive, relational, as well as procedural elements of teams’ knowledge integration infrastructure. Another interesting relationship would be between project interdependence and the procedural elements (e.g., inter- and intra-team communications, and decision making processes) of the infrastructure.

Figure 5.1: Proposed Framework for Future Research - I

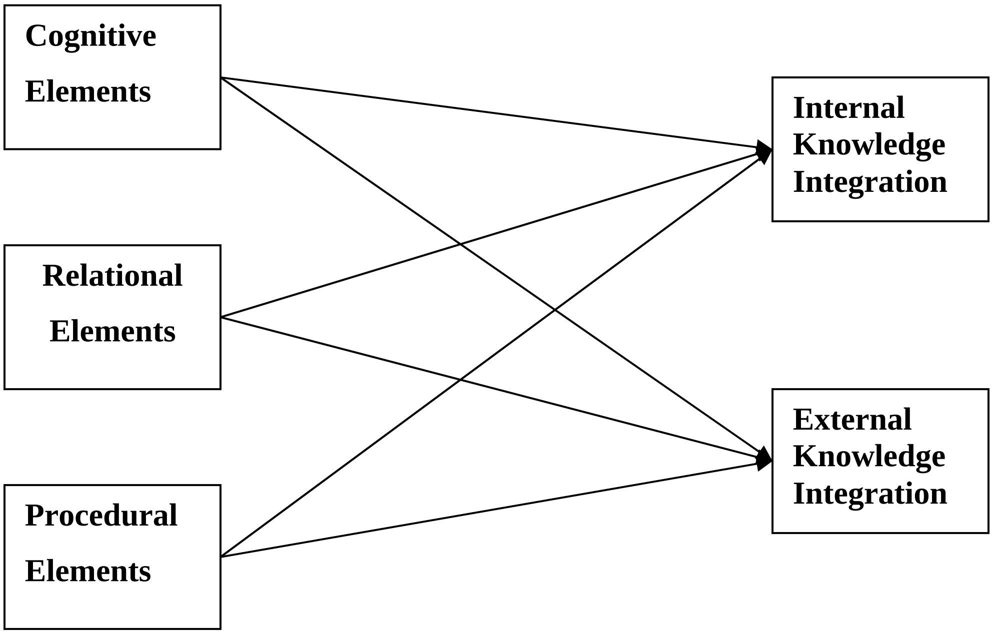
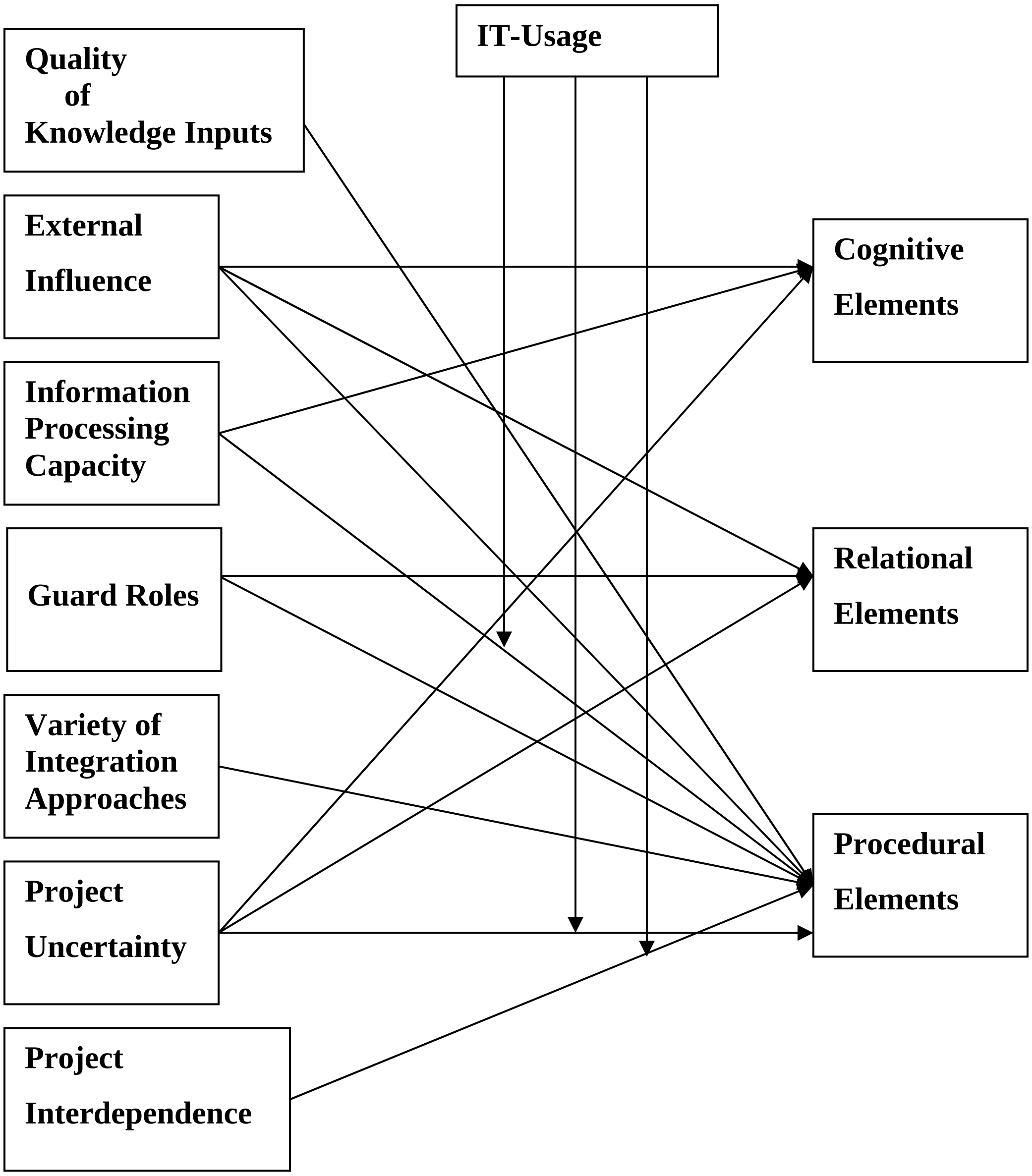


Figure 5.2: Proposed Stream of Future Research - II



BIBLIOGRAPHY

Alavi, M. "Managing Organizational Knowledge," in *Framing the Domains of IT Management: Projecting the Future Through the Past,* R.W. Zmud (Ed.), Pinnaflex Edu. Resources Inc., Cincinnati, OH, 2000, pp. 15-28.

Alavi, M., and Leidner, D. "Knowledge Management and Knowledge Management Systems: Conceptual Foundations And Research Issues," *MIS Quarterly* (25:1), 2001, pp. 107-136.

Alavi, M., and Tiwana, A. "Knowledge Integration in Virtual Teams: The Potential Role of KMS," *Journal of The American Society for Information Science and Technology* (53:12), 2002, pp. 1029-1037.

Aldrich, H., and Herker, D. "Boundary Spanning Roles and Organizational Structures," *Academy of Management Review* (2), 1977, pp. 217-230.

Allen, T. J. *Managing the Flow of Technology* MIT Press, Cambridge, MA, 1977.

Allen, T. J., and Cohen, S. "Information Flow in R&D Labs," *Administrative Science Quarterly* (14), 1969, pp. 12-19.

Anand, V., Clark, M. A., And Zellmer-Bruhn, M. "Team Knowledge Structures: Matching Task to Information Environment," *Journal Of Managerial Issues* (15:1), 2003, pp. 15-31.

Ancona, D. G., and Caldwell, D. F. "Beyond Task and Maintenance: Defining External Functions in Groups," *Groups And Organizational Studies* (13), 1988, pp. 468-494.

Ancona, D. G., and Caldwell, D. F. "Bridging the Boundary: External Activity and Performance in Organizational Teams," *Administrative Science Quarterly* (37), 1992a, pp. 634-665.

Ancona, D. G., and Caldwell, D. F. "Demography and Design: Predictors of New Product Team Performance," *Organization Science* (3:3), 1992b, pp. 321-341.

Andres, H.P., and Zmud, R.W. "A contingency approach to software project coordination," *Journal of Management Information Systems* (18:3), 2001, pp. 41-70.

Argote, L. "Input uncertainty and organizational coordination in hospital emergency units," *Administrative Science Quarterly* (27), 1982, pp. 143-174.

Argote, L. *Organizational Learning. Creating, Retaining, and Transferring Knowledge* Kluwer Academic Publishers, Boston, MA, 1999.

Awazu, Y. "Knowledge management in distributed environments: Roles of informal network players," 37th Hawaii International Conference on System Sciences, Hawaii, 2004.

Barclay, D., Higgins, C., and Thompson, R. "The Partial Least Squares (PLS) Approach to Causal Modeling, Personal Computer Adoption and Use as an Illustration." *Technology Studies* (2:2), 1995, pp. 285-309.

Beath, C. M. "Strategies for Managing MIS Projects: A Transaction Cost Approach," Proceedings of the 4th International Conference on Information Systems, Houston, TX, 1983, pp. 133-147.

Bikson, T. K., Cohen, S. G., and Mankin, D. "Information Technology and High Performing Teams," In *Supporting Work Team Effectiveness: Best Management Practices for Fostering High Performance*, E.D. Sundstrom (Ed.), Jossey-Bass, San Francisco, CA, 1999, pp. 215-245.

Blau, P. M. *Exchange and power in social life* Wiley, New York, 1964.

Boehm, B. W. *Software risk management* IEEE Computer Society Press, Washington D.C., 1989.

Bowman, B. J. "Building Knowledge Management Systems," *Information Systems Management* (19:3), 2002, pp. 32-40.

Bradach, J. L., and Eccles, R. G. "Price, Authority, and Trust: From Ideal Types to Plural Forms," In *Annual Review of Sociology*, W.R. Scott and J. Blake (Eds.), 1989, pp. 97-118.

Brown, C. V. "Horizontal Mechanisms Under Differing IS Organization Contexts," *MIS Quarterly* (23:3), 1999, pp. 421-454.

Brown, J. S., and Duguid, P. "Organizational Learning and Communities of Practice," *Organization Science* (2:1), 1991, pp. 40-57.

Campion, M. A., Medsker, G. J., and Higgs, A. C. "Relations Between Work Group Characteristics and Effectiveness: Implications for Designing Effective Work Groups," *Personnel Psychology* (46:4), 1993, pp. 823-850.

Campion, M. A., Papper, E. M., and Medsker, G. J. "Relations Between Work Team Characteristics and Effectiveness: A Replication and Extension," *Personnel Psychology* (49), 1996, pp. 429-452.

Chin, W. W. "The Partial Least Squares Approach to SEM," In *Modern Methods for Business Research,* G.A. Marcoulides (Ed.), Lawrence Erlbaum Associates, Mahwah, NJ, 1998.

Chin, W. W., Marcolin, B. L., and Newsted, P. R. "A Partial Least Squares Latent Variable Modeling Approach for Measuring Interaction Effects: Results from a Monte Carlo Simulation Study and An Electronic-Mail Emotion / Adoption Study," *Information Systems Research* (14:2), 2003, pp. 189-217.

Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, (2nd ed.) Lawrence Erlbaum, Hillsdale, NJ, 1988.

Cohen, S. G., and Bailey, D. B. "What Makes Teams Work: Group Effectiveness Research from the Shop Floor to the Executive Suite," *Journal of Management* (23:3), 1997, pp. 239-290.

Cohen, W. M., and Levinthal, D. A. "Absorptive Capacity: A New Perspective on Learning and Innovation," *Administrative Science Quarterly* (35), 1990, pp. 128-152.

Cole, R. E. "Introduction," *California Management Review* (45:3), Spring 1998, pp. 15-21.

Constantine, L., and Lockwood, L. "Orchestrating Project Organization and Management," *Communications of the ACM* (36:10), 1993, pp. 31-43.

Cronbach, L. J. "Coefficient Alpha and the Internal Structure of Tests," *Psychometrika* (16), 1951, pp. 297-234.

Cummings, J. N. "Work Groups, Structural Diversity, and Knowledge Sharing inaA Global Organization," *Management Science* (50:3), 2004, pp. 352-364.

Curtis, B., Krasner, H., and Iscoe, N. "A Field Study Of The Software Design Process For Large Systems," *Communications of the ACM* (46:1), 1988, pp. 99-101.

Daft, R. L., and Lengel, R. H. "Organizational Information Requirements, Media Richness and Structural Design," *Management Science* (32:5), 1986, pp. 554-571.

Daft, R. L., Lengel, R. H., and Trevino, L. K. "Message Equivocality, Media Selection, and Manager Performance: Implications for Information Systems," *MIS Quarterly* (11:3), 1987, pp. 355-366.

Dasgupta, P. "Trust as a Commodity," In *Trust: Making and Breaking Cooperative Relations*, D. Gambetta (Ed.), Blackwell Publishing, New York, 1988, pp. 47-72.

Davenport, T. H., and Prusak, L. *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, Cambridge, MA, 1997.

Faraj, S., and Sproull, L. "Coordinating Expertise in Software Developing Teams," *Management Science* (46:12), 2000, pp. 1554-1568.

Fiske, S., Kinder, D., and Larter, W. "The Novice and the Expert: Knowledge Based Strategies in Political Cognition," *Journal of Experimental Social Psychology* (19), 1983, pp. 381-400.

Fornell, C., and Larker, D. F. "Evaluating Structural Equation Models with Unobservable Variables and Measurement Error," *Journal of Marketing Research* (18), 1981, pp. 39-50.

Friedlander, F. "The Ecology of Work Groups," In *Handbook of Organizational Behavior*, J. Lorsch (Ed.), Prentice Hall, Englewood-Cliffs, NJ, 1985, pp. 301-314.

Fulk, J., and DeSanctis, G. "Electronic Communications And Changing Organizational Forms," *Organization Science* (6:4), 1995, pp. 337-350.

Galbraith, J. *Designing Complex Organizations*, Addison-Wesley, Reading, MA, 1973.

Galbraith, J. *Organizational Design* Addison-Wesley, Reading, MA, 1977.

Galeghar, J., and Kraut, R. E. "Computer-Mediated Communication for Intellectual Teamwork: An Experiment in Group Writing," *Information Systems Research* (5:2), 1994, pp. 110-138.

Garner, W. R. *Uncertainty and Structure as Psychological Concepts*, Wiley, New York, 1962.

Gefen, D., and Straub, D. "A Practical Guide to Factorial Validity Using PLS-GRAPH: Tutorial and Annotated Example," *Communications of the AIS* (16), 2005, pp. 91-109.

Gold, A. H., Malhotra, A., and Segars, A. H. "Knowledge Management: An Organizational Capabilities Perspective," *Journal of Management Information Systems* (18:1), 2001, pp. 185-214.

Goodhue, D. L., and Thompson, R. L. "Task-Technology Fit and Individual Performance," *MIS Quarterly* (19:2), 1995, pp. 213-236.

Goodman, P. S., and Darr, E. D. "Computer-Aided Systems and Communities: Mechanisms for Organizational Learning in Distributed Environments," *MIS Quarterly* (22:4), 1998, pp. 417-440.

Grant, R. M. "Prospering in Dynamically-Competitive Environments: Organizational Capability as Knowledge Integration," *Organization Science* (7:4), 1996a, pp. 375-387.

Grant, R. M. "Toward a Knowledge-Based Theory of the Firm," *Strategic Management Journal* (17), 1996b, pp. 109-122.

Guinan, P. J., Cooprider, J. G., and Faraj, S. "Enabling Software Development Team Performance During Requirements Definition: A Behavioral Versus Technical Approach," *Information Systems Research* (9:2), 1998, pp. 101-125.

Gulati, R. "Does Familiarity Breed Trust? The Implications of Repeated Ties for Contractual Choice in Alliances," *Academy of Management Journal* (38), 1995, pp. 85-112.

Hansen, H. "Knowledge Networks: Explaining Effective Knowledge Sharing in Multiunit Companies," *Organization Science* (13:3), 2002, pp. 232-248.

Hansen, M. T., Nohria, N., and Tierney, T. "What's Your Strategy For Managing Knowledge?" *Harvard Business Review*, 1999, pp. 106-116.

Hiltz, S. R., Johnson, K., and Turoff, M. "Experiments in Group Decision Making: Communication Process and Outcome in Face-to-Face Versus Computerized Conferences," *Human Communications Research* (13), 1986, pp. 225-252.

Hinds, P., and Kiesler, S. "Communication Across Boundaries: Work Structure, and Use of Communication Technologies in a Large Organization," *Organization Science* (6:4), 1995, pp. 373-393.

Hoegl, M., Weinkauf, K., and Gemuenden, H. G. "Interteam Coordination, Project Commitment, and Teamwork in Multiteam R&D Projects: A Longitudinal Study," *Organization Science* (15:1), 2004, pp. 38-55.

Hollingshead, A. B. "Perceptions of Expertise and Transactive Memory in Work Relationships," *Group Processes Intergroup Relations* (3:3), 2000, pp. 257-267.

Hollingshead, A. B. "Cognitive Interdependence and Convergent Memory in Transactive Memory," *Journal of Personality and Social Psychology* (81:6), 2001, pp. 1080-1089.

Huber, G. P. "The Nature and Design of Post-Industrial Organizations," *Management Science* (30), 1984, pp. 928-951.

Jarvenpaa, S. L., and Staples, D. S. "The Use of Collaborative Electronic Media for Information Sharing: An Exploratory Study of Antecedents," *Journal of Strategic Information Systems* (9:2-3), 2000, pp. 129-154.

Jehn, K. A., and Mannix, E. "The Dynamic Nature of Conflict: A Longitudinal Study of Intergroup Conflict and Group Performance," *Academy of Management Journal* (44:2), 2001, pp. 238-251.

Jemison, D. B. "The Importance of Boundary Spanning Roles in Strategic Decision Making," *Journal of Management Studies* (21:2), 1984, pp. 131-152.

Kale, P., Singh, H., and Perlmutter, H. "Learning and Protection of Proprietary Assets in Strategic Alliances: Building Relational Capital," *Strategic Management Journal* (21:3), 2000, pp. 217-237.

Kankanhalli, A., Tan, B. C. Y., and Wei, K. "Seeking Knowledge in Electronic Knowledge Repositories: An Exploratory Study," Twenty-Second International Conference on Information Systems, New Orleans, 2001.

Kankanhalli, A., Tan, B. C. Y., and Wei, K. "Contributing Knowledge to Electronic Knowledge Repositories: An Empirical Investigation," *MIS Quarterly* (29:1), 2005, pp. 113-143.

Kazanjian, R. K., Drazin, R., and Glynn, M. A. "Creativity and Technological Learning: The Roles of Organization Architecture and Crisis in Large-Scale Projects," *Journal of Engineering Technology Management* (17), 2000, pp. 273-298.

Kim, K. K. "Organizational Coordination and Performance in Hospital Accounting Information Systems: An Empirical Investigation," *Accounting Review* (63:3), 1988, pp. 472-489.

Kim, K. K., and Umanath, N. S. "Structure and Perceived Effectiveness of Software Development Subunits: A Task Contingency Analysis," *Journal of Management Information Systems* (9:3), 1992-93, pp. 157-181.

Kirsch, L. J. "The Management of Complex Tasks in Organizations: Controlling the Systems Development Process," *Organization Science* (7:1), 1996, pp. 1-21.

Ko, D., Kirsch, L. J., and King, W. R. "Antecedents oKnowledge Transfer From Consultants to Clients in Enterprise System Implementations," *MIS Quarterly* (29:1), 2005, pp. 59-85.

Kogut, B., and Zander, U. "Knowledge of the Firm, Combinative Capabilities, and the Replication of Technology," *Organization Science* (3), 1992, pp. 383-397.

Komi-Sirvio, S., Mantyniemi, A., and Seppanen, V. "Toward a Practical Solution for Capturing Knowledge for Software Projects," *IEEE Software* (19:3), 2002, pp. 60-62.

Koushik, M. V., and Mookerjee, V. S. "Modeling Coordination in Software Construction: An Analytical Approach," *Information Systems Research* (6:3), 1995, pp. 220-254.

Kraut, R. E., Fish, R., and Chalfonte, B. "Requirements and Media Choice in Collaborative Writing," *Human-Computer Interaction* (7), 1992, pp. 375-407.

Kraut, R. E., and Streeter, L. A. "Coordination in Software Development," *Communications of the ACM* (38:3), 1995, pp. 69-81.

Kydd, C. T. "Understanding the Information Content in MIS Management Tools," *MIS Quarterly* (13:3), 1989, pp. 277-290.

Lakhani, K., and von Hipple, E. "How Open Source Software Works: "Free" User-to-User Assistance," 3rd Intangibles Conference, Stern School of Business, New York University, 2000.

Lawrence, P. R., and Lorsch, J. W. *Organization and Environment*, Harvard Business School Press, Boston, MA, 1986.

Lee, C. C., and Grover, V. "Exploring Mediation Between Environmental and Structural Attributes: The Penetration of Communication Technologies in Manufacturing Organizations," *Journal of Management Information Systems* (16:3), 1999/2000, pp. 187-217.

Lee, H., and Choi, B. "Knowledge Management Enablers, Processes, and Organizational Performance: An Integrative View and Empirical Examination," *Journal of Management Information Systems* (20:1), 2003, pp. 179-228.

Lefton, M., and Rosengren, W. R. "Organizations And Clients: Lateral And Longitudinal Dimensions," *American Sociological Review* (31), 1966, pp. 802-810.

Lewicki, R. J., and Bunker, B. B. "Trust in Relationships: A Model of Trust Development and Decline," In *Conflict, Cooperation, and Justice,* B.B. Bunker and J.Z. Rubin (Eds.), Jossey-Bass, San Francisco, CA, 1995, pp. 133-173.

Lewis, K. "Knowledge and Performance in Knowledge-Worker Teams: A Longitudinal Study of Transactive Memory Systems," *Management Science* (50:11), 2004, pp. 1519-1533.

Loch, C. H., and Terwiesch, C. "Communication and Uncertainty in Concurrent Engineering," *Management Science* (44:8), 1998, pp. 1032-1048.

Lord, R., and Maher, R. "Alternative Information-Processing Models and their Implications for Theory, Research, and Practice," *Academy of Management Review* (15:1), 1990, pp. 9-28.

Majchrzak, A., Beath, C. M., and Lim, R. A. "Managing Client Dialogues During Information Systems Design to Facilitate Client Learning," *MIS Quarterly*) Forthcoming.

Malone, T. W., Yates, J., and Benjamin, R. I. "Electronic Markets and Electronic Hierarchies," *Communications of the ACM* (30:6), 1987, pp. 484-497.

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

Marks, M. A., Mathieu, J. E., and Zaccaro, S. J. "A Temporally Based Framework and Taxonomy of Team Processes," *Academy of Management Review* (26:3), 2001, pp. 356-376.

Markus, M. L. "Power, Politics, and MIS Implementation," *Communications of the ACM* (26:8), 1983, pp. 430-444.

Marsden, P. V. "Network Data and Measurement," *Annual Review of Sociology* (16), 1990, pp. 435-463.

Maruping, L. M., and Agarwal, R. "Managing Team Interpersonal Processes Through Technology: A Task-Technology Fit Perspective," *Journal of Applied Psychology* (89:6), 2004, pp. 975-990.

Mathiassen, L., and Pourkomeylian, P. "Managing Knowledge in a Software Organization," *Journal of Knowledge Management* (7:2), 2003, pp. 63-80.

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

Miller, E. J., and Rice, A. K. *Systems of Organization: The Control of Task and Sentient Boundaries*, Tavistock Publications, London, 1967.

Misztal, B. *Trust in Modern Societies*, Polity Press, Cambridge, England, 1996.

Moran, P., and Ghoshal, S. "Value Creation by Firms," Academy of Management Best Paper Proceedings, 1996, pp. 41-45.

Mowery, D. C., and Oxley, J. E. "Inward Technology Transfer and Competitiveness: The Role of National Innovation Systems," *Cambridge Journa of Economics* (19), 1995, pp. 67-93.

Mukherji, S. "Knowledge Sharing in Software Development Teams," in: *Unpublished Dissertation*, Bangalore, India, Indian Institute of Management, 2002.

Nahapiet, J. "Managing Relationships with Global Clients: Value-Creation Through Cross-Border Networks," 16th Annual Conference of the Strategic Management Society, Phoenix, AZ, 1996.

Nahapiet, J., and Ghoshal, S. "Social Capital, Intellectual Capital, and the Organizational Advantage," *Academy of Management Review* (23:2), 1998, pp. 242-266.

Nelson, K. M., and Cooprider, J. G. "The Contribution of Shared Knowledge to IS Group Performance," *MIS Quarterly* (20:4), 1996, pp. 409-432.

Nemeth, C. "Minority Dissent as a Stimulant to Group Performance," In *Group Process and Productivity,* S. Worchel, W. Wood and J.A. Simpson (Eds.), Sage, Newbury Park, CA, 1992.

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

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

Nohria, N., and Eccles, R. G. "Face-To-Face: Making Network Organizations Work," In *Networks and Organizations: Structure, Form, and Action,* N. Nohria and R.G. Eccles (Eds.), Harvard Business School Press, Boston, MA, 1992, pp. 288-908.

Nonaka, I. "The Knowledge Creating Company," *Harvard Business Review* (69:6), 1991, p 96.

Nonaka, I., and Takeuchi, H. *The Knowledge Creating Company*, Oxford University Press, New York, 1995.

Norman, P. M. "Knowledge Acquisition, Knowledge Loss, and Satisfaction in High Technology Alliances," *Journal of Business Research* (57:6), 2004, pp. 610-619.

Nunnally, J. C. *Psychometric Theory*, McGraw Hill, New York, 1967.

Okhuysen, G., and Eisenhardt, K. "Integrating Knowledge in Groups: How Formal Interventions Enable Flexibility," *Organization Science* (13:4), 2002, pp. 370-386.

Orlikowski, W. J. "Learning From Notes: Organizational Issues in Groupware Implementation," In *Computerization and Controversy,* R. Kling (Ed.), Academic, New York, 1996, pp. 173-189.

Orlikowski, W. J., and Yates, J. "Genre Repertoire: Examining the Structuring of Communicative Practices in Organizations," *Administrative Science Quarterly* (39:4), 1994, pp. 541-574.

Pawlowski, S. D., and Robey, D. "Bridging User Organizations: Knowledge Brokering and the Work of Information Technology Professionals," *MIS Quarterly* (28:4), 2004, pp. 645-672.

Pearce, J. L., and Gregersen, H. B. "Task Interdependence and Extrarole Behavior: A Test for Mediating Effects of Felt Responsibility," *Journal of Applied Psychology* (76:6), 1991, pp. 838-844.

Perrow, C. "A Framework for the Comparative Analysis of Organizations," *American Sociological Review* (32), 1967, pp. 194-208.

Pfeffer, J. "A Resource Dependence Perspective on Intercorporate Relations," In *Structural Analysis of Business*, M.S. Mizruchi and M. Schwartz (Eds.), Academic Press, New York, 1986, pp. 117-132.

Powell, W. "Learning from Collaboration: Knowledge and Networks in the Biotechnology and Pharmaceutical Industries," *California Management Review* (40:3), 1998, pp. 228-240.

Quinn, J. B. *Intelligent Enterprises*, Free Press, New York, 1992.

Ramesh, B. "Process Knowledge Management with Traceability," *IEEE Software* (19:3), 2002, pp. 50-55.

Ring, P. S., and Van de Ven, A. H. "Structuring Cooperative Relationships Between Organizations," *Strategic Management Journal* (13:7), 1994, pp. 483-498.

Roberts, J. "From Know-How to Show-How? Questioning the Role of Information and Communication Technologies in Knowledge Transfer," *Technology Analysis and Strategic Management* (12:4), 2000, pp. 429-443.

Robinson, J. P., Shaver, P. R., and Wrightsman, L. S. "Criteria for Scale Selection and Evaluation," In *Measures of Personality and Social Psychology Attitudes*, J.P. Robinson, P.R. Shaver and L.S. Wrightsman (Eds.), Academic Press, San Diego, CA, 1991.

Rulke, D., and Galaskiewicz, J. "Distribution of Knowledge, Group Network Structure, and Group Performance," *Management Science* (46:5), 2000, pp. 612-625.

Ryu, C., Kim, Y. J., Chaudhary, A., and Rao, H. R. "Knowledge Acquisition via Three Learning Processes in Enterprise Information Portals: Learning-by-Investment, Learning-by-Doing, and Learning-from-Others," *MIS Quarterly* (29:2), 2005, pp. 245-278.

Saavedra, R., Earley, P. C., and Van Dyne, L. "Complex Interdependence in Task Performing Groups," *Journal of Applied Psychology* (78:1), 1993, pp. 61-72.

Sabbagh, K. *21st-Century Jet: The Making and Marketing of Boeing 777*, Scribner, New York, 1996.

Sambamurthy, V., Poole, M. S., and Kelly, J. "The Effects of Variations in GDSS Capabilities on Decision-Making Processes in Groups," *Small Groups Research* (24), 1993, pp. 523-546.

Sambamurthy, V., and Subramani, M. "Introduction to Special Issue on Information Technologies and Knowledge Management," *MIS Quarterly* (29:1), March 2005, pp. 1-7.

Schultze, U., and Orlikowski, W. J. "A Practice Perspective on Technology-Mediated Network Relations: The Use of Internet-Based Self-Serve Technologies," *Information Systems Research* (15:1), 2004, pp. 87-106.

Schumpeter, J. A. *The theory of economic development: An enquiry into profits, capital credit, interest and the business cycle*, Harvard University Press, Cambridge, MA, 1934.

Scott, J. E. "Organizational Knowledge and the Internet.," *Decision Support Systems* (23:1), 1998, pp. 3-17.

Scott, W. R. *Organizations: Rational, Natural, and Open Systems*, (3rd ed.) Prentice Hall, Englewood Cliffs, NJ, 1992.

Sher, P. J., and Lee, V. C. "Information Technology as a Facilitator for Enhancing Dynamic Capabilities through Knowledge Management," *Information & Management* (41), 2003, pp. 933-945.

Smith, K. G., Collins, C. J., and Clark, K. D. "Existing Knowledge, Knowledge Creation Capability, and the Rate of New Product Introduction in High-Technology Firms," *Academy of Management Journal* (48:2), 2005, pp. 346-357.

Spender, J. C., and Grant, R. M. "Knowledge and the Firm: Overview," *Strategic Management Journal* (17), 1996, pp. 5-9.

Stone, E. F. *Research Methods in Organizational Behavior,* Goodyear, Santa Monica, CA, 1978.

Struas, S. G., and McGrath, J. E. "Does Medium Matter? The Interaction of Task Type and Technology on Group Performance and Member Reactions," *Journal of Applied Psychology* (79:1), 1994, pp. 87-97.

Szulanski, G. "Exploring External Stickiness: Impediments to the Transfer of Best Practices Within the Firm," *Strategic Management Journal* (17), 1996, pp. 27-43.

Tanriverdi, H. "Information Technology Relatedness, Knowledge Management Capability, and Performance of Multibusiness Firms," *MIS Quarterly* (29:2), 2005, pp. 311-334.

Teigland, R., and Wasko, M. M. "Integrating Knowledge Through Information Trading: Examining the Relationship Between Boundary Spanning Communication and Individual Performance," *Decision Sciences* (34:2), 2003, pp. 261-286.

Tempelton, G. F., Lewis, B. R., and Snyder, C. A. "Development of a Measure for the Organizational Learning Construct," *Journal of Management Information Systems* (19:2), 2002, pp. 175-218.

Thomas-Hunt, M. C., Ogden, T. Y., and Neale, M. A. "Who's Really Sharing? Effects of Social and Expert Status on Knowledge Exchange Within Groups," *Management Science* (49:4), 2003, pp. 464-477.

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

Tiwana, A. "Affinity to Infinity in Peer to Peer Knowledge Platforms," *Communications of the ACM* (46:5), 2003, pp. 77-80.

Tiwana, A. "Knowledge Partitioning in Outsourced Software Development: A Field Study," International Conference on Information Systems, Seattle, Washington, 2003a, pp. 259-270.

Tiwana, A., Bharadwaj, A., and Sambamurthy, V. "The Antecedents of Information Systems Development Capability in Firms: A Knowledge Integration Perspective," Twenty-Fourth International Conference on Information Systems, Seattle, Washington, 2003.

Tiwana, A., and McLean, E. "Expertise Integration and Creativity in Information Systems Development," *Journal of Management Information Systems* (22:1), 2005, pp. 13-43.

Trevino, L. K., Lengel, R. H., and Daft, R. L. "Media Symbolism, Media Richness, and Media Choice in Organizations," *Communications Research* (14:5), 1987, pp. 553-574.

Tsai, W., and Ghoshal, S. "Social Capital and Value Creation," *Academy of Management Journal* (41:4), 1998, pp. 464-476.

Tsoukas, H. "The Firm as a Distributed Knowledge System: A Constructionist Approach.," *Strategic Management Journal* (17:Winter Special Issue), 1996, pp. 11-25.

Turner, J. A. "A Comparison of the Process of Knowledge Elicitation with that of Information Requirements Determination," In *Challenges and Strategies for Research in Systems Development,* W.W. Cotterman and J.A. Senn (Eds.), John Wiley & Sons, Ltd., New York, 1992.

Tushman, M. L. "Work Characteristics and Subunit Communication Structure: A Contingency Analysis," *Administrative Science Quarterly* (24:1), 1979, pp. 82-97.

Tushman, M. L., and Nadler, D. "Information Processing as an Integrated Concept in Organizational Design," *Academy of Management Review* (3:3), 1978, pp. 613-624.

Ungson, G. R., Braunstein, D. N., and Hall, P. D. "Managerial Information Processing: A Research Review," *Administrative Science Quarterly* (26), 1981, pp. 116-134.

Van de Ven, A. H., Delbecq, A., and Koeing, J. R. "Determinants of Coordination Modes Within Organizations," *American Sociological Review* (41:2), 1976, pp. 322-338.

von Hipple, E. *The Sources of Innovation*, Free Press, New York, 1988.

Wageman, R. "Interdependence and Group Effectiveness," *Administrative Science Quarterly* (40:1), 1995, pp. 145-180.

Walsh, B., and Baker, A. "Project Management and Communications Patterns in Industrial Research," *R&D Management* (2), 1972, pp. 103-109.

Walz, D. B., Elam, J. J., and Curtis, B. "Inside A Software Design Team: Knowledge Acquisition, Sharing, And Integration," *Communications of the ACM* (36:10), 1993, pp. 63-77.

Wasko, M. M., and Faraj, S. "It is What One Does: Why People Participate and Help Others in Electronic Communities of Practice," *Journal of Strategic Information Systems* (9:2/3), 2000, pp. 155-173.

Wittenbaum, G. M., Vaughan, S. I., and Stasser, G. "Coordination in Task Performing Groups," In *Social Psychological Applications to Social Issues: Theory and Research on Small Groups*, R.S. Tindale, J. Heath, E.J. Edwards, E.J. Posavac, F.B. Brynt, Y. Suarez-Balcazar, E. Henderson-King and J. Myres (Eds.), Plenum, New York, 1998, pp. 177-204.

Wold, H. "Soft Modeling: The Basic Design and Some Extensions," In *Systems Under Indirect Observations: Causality, Structure, Prediction*, K.G. Joreskog and H. Wold (Eds.), North-Holland, Amsterdam, 1982.

Wong, C., and Campion, M. A. "Development and Test of a Task Level Model of Motivational Job Design," *Journal of Applied Psychology* (76:6), 1991, pp. 825-837.

Yan, A., and Louis, M. R. "The Migration of Organizational Functions to the Work Unit Level: Buffering, Spanning, and Bringing Up Boundaries," *Human Relations* (52:1), 1999, pp. 25-47.

Yoo, Y., and Alavi, M. "Media and Group Cohesion: Relative Influences on Social Presence, Task Participation, and Group Consensus," *MIS Quarterly* (25:3), 2001, pp. 371-390.

Zahra, S. A., and George, G. "Absorptive Capacity: A Review, Reconceptualization, and Extension," *Academy of Management Review* (27:2), 2002, pp. 185-203.

Zellmer-Bruhn, M. "Interruptive Events and Team Knowledge Acquisition," *Management Science* (49:4), 2003, pp. 514-528.

Zigers, I., and Kozar, K. A. "An Exploratory Study of Roles in Computer Supported Groups," *MIS Quarterly* (18:3), 1994, pp. 277-297.

Zmud, R. W. "Management of Large Software Development Efforts," *MIS Quarterly* (4:2), 1980, pp. 45-55.

Zmud, R. W. "The Effectiveness of External Information Channels in Facilitating Innovation Within Software Development Groups," *MIS Quarterly* (7:2), 1983, pp. 43-58.

Zmud, R. W., Lind, M. R., and Young, W. F. "An Attribute Space for Organizational Communication Channels," *Information Systems Research* (1), 1990, pp. 440-457.

APPENDIX

SURVEY INSTRUMENTS

Q1: 2005 - 2006 Project Leader Survey

Welcome! We invite you to participate in an important research study that examines knowledge usage in software teams. Your opinion is very valuable to the success of this study. The results of this study may help project leaders/ project managers like you gain better understanding of issues such as:

- How your team members use their own knowledge, and knowledge from various external sources to achieve project goals.
- Do characteristics of your team affect your team's knowledge usage.
- Do project characteristics (e.g. complexity) affect you team's knowledge usage.
- How do knowledge management (KM) systems affect your team's knowledge usage

All responses are completely **ANONYMOUS**. There will be no way to identify the person who filled out the survey. With this in mind, please take a few minutes to provide your honest opinion to each statement.
You may decide to withdraw at any time during the research (without penalty). However, after you have provided anonymous information you will be unable to withdraw your responses after participation since there will be no way to identify individual information. Your decision whether or not to participate will not jeopardize your future relations with Auburn University of the Department of Management.

If you have any questions I invite you to ask them now. If you have questions later, please contact either of the persons mentioned below, and we will be happy to answer them.

Nikhil Mehta	Dr. Terry Byrd
Phone: +1-334-844-6534	Phone: +1-334-844-6543
(mehtan1@auburn.edu)	(tbyrd@auburn.edu)

For more information regarding your rights as a research participant you may contact the Office of Human Subjects Research by (334)-844-5966 or e-mail at hsubjec@auburn.edu or IRBChair@auburn.edu .

HAVING READ THE INFORMATION PROVIDED, YOU MUST DECIDE WHETHER TO PARTICIPATE IN THIS RESEARCH PROJECT. IF YOU DECIDE TO PARTICIPATE, THE DATA YOU PROVIDE WILL SERVE AS YOUR AGREEMENT TO DO SO.

Directions

Please read carefully all of the instructions before beginning this questionnaire. The questions in this survey are broken up into two sections labeled "Most Successful Project" and "Least Successful Project." Each section will have specific directions. Please read each statement carefully. For each statement, circle the response that best represents your opinion. Please answer each question.

General Information

What is your gender (please circle): Female Male

Years of experience in the software industry (approximate): ____

Name of your current organization: _________________________

Current Position (please circle): Project Leader Project Manager

SECTION 1 – MOST SUCCESSFUL PROJECT

Please think of the MOST SUCCESSFUL PROJECT in your experience as the project leader/ project manager. Some parameters of the most successful project may include, but are not limited to:

- It was completed within budget and time
- All/most of the performance requirements were met
- All/most of the critical risks were mitigated
- Most critical issues were resolved before delivery
- A project that YOU feel was completed most satisfactorily

Please provide some information about that project:

- Project Start Date: _____________________ (month/year)
- Project End Date: ______________________ (month/year)
- Number of Team Members: ____
- Nature of the Project (please check): Product Development______ Developing Customized Solution________

The statements below refer to various aspects of that (most successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The statements below refer to the outcomes of that (most successful) project.

Compared to other projects, the outcomes of that project were more unpredictable	1	2	3	4	5	6	7
Compared to other projects, it took more time to foresee the outcomes of that project	1	2	3	4	5	6	7
Compared to other projects, it was more difficult to understand the outcomes of that project	1	2	3	4	5	6	7

How would you disagree or agree with each of the following statements about the conversion of requirements specifications to software, in that (most successful) project?

Compared to other projects, there was more confusion in that project about developing software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, established procedures and practices could not be relied upon in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, an understandable sequence of steps could not be followed in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7

The statements below refer to any unexpected or novel technological problems that occurred in that (most successful) project							
Compared to other projects, that project faced highly unexpected or novel problems related to **software platforms**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software programming languages**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software coding**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software implementation**	1	2	3	4	5	6	7

The statements below refer to the team members who worked on that (most successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The following statements explore how the team members of that (most successful) project synthesized and combined their knowledge, expertise, and skills to achieve project goals.							
Team members combined their individual expertise to jointly solve project-related problems	1	2	3	4	5	6	7
Team members combined their individual perspectives to develop a shared project concepts	1	2	3	4	5	6	7
Team members often gained new insights by sharing their ideas with each other	1	2	3	4	5	6	7
Team members improved their task efficiency by sharing their knowledge with each other	1	2	3	4	5	6	7

Section 1 – Most Successful Project

The statements below explore if the team members of that (most successful) project acquired and utilized project-related knowledge from external sources such as other teams, individuals, and departments. Such knowledge may have been acquired through personal interactions, over the phone, through e-mails, or through the knowledge management (KM) system.

If the required knowledge was not available within the team, members acquired that knowledge from external sources	1	2	3	4	5	6	7
Team members often reused code available from other projects	1	2	3	4	5	6	7
Team members often enhanced their knowledge with inputs from external sources	1	2	3	4	5	6	7

The statements below refer to the diversity of backgrounds, expertise, and skills of the team members of that (most successful) project.

Members of that team varied widely in their areas of expertise	1	2	3	4	5	6	7
Members of that team had a variety of different backgrounds and experiences	1	2	3	4	5	6	7
Members of that team had wide-ranging skills and abilities	1	2	3	4	5	6	7

The statements below refer to the working relationships, mutual trust among the team members of that (most successful) project.

The team was characterized by personal relationships among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by high level of reciprocal behavior among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by mutual trust among members at multiple levels	1	2	3	4	5	6	7

Section 1 – Most Successful Project

The statements below refer to the policy of the team (working on that most successful project) about information coming into the team from external sources (e.g. other teams, individuals, and departments in the company)

The team actively monitored information coming from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively decided what type of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively controlled the internal distribution of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7

The statements below refer to the policy of the team (working on that most successful project) about sharing internal information and other resources with other teams, individuals, and departments in the company

The team avoided releasing internal information to others in the company	1	2	3	4	5	6	7
The team actively controlled the use of its internal resources by others in the company	1	2	3	4	5	6	7

The statements below assess the interdependence of the team (working on that most successful project) with other teams.

The team closely coordinated project-related tasks with other teams	1	2	3	4	5	6	7
The team regularly received project-related information or resources from other teams	1	2	3	4	5	6	7
The team’s performance evaluation was strongly influenced by how well other teams performed	1	2	3	4	5	6	7

Section 1 – Most Successful Project

The following statements explore how the team (working on that most successful project) used various IT-based systems for different purposes. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.

The team used IT-based systems to coordinate with others in the company	1	2	3	4	5	6	7
The team used IT-based systems to search for project-related knowledge	1	2	3	4	5	6	7
The team used IT-based systems to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

The following statements explore how the team (working on that most successful project) used various IT-based systems as compared to other teams you have led. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.

Compared with other teams you have led, this team uses collaborative systems MORE to internally coordinate project-related tasks	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to coordinate with others in the company	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to search for project-related knowledge	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

SECTION 2 – LEAST SUCCESSFUL PROJECT

Now, please think of the LEAST SUCCESSFUL PROJECT in your experience as a project leader/ project manager. Some parameters of the least successful project may include, but are not limited to:

- It was not completed within budget and time
- Many of the performance requirements were not met
- Many of the critical risks were not mitigated
- Many critical issues were not resolved before delivery
- A project that YOU feel was not completed satisfactorily

Please provide some information about that project:

- Project Start Date: ______________________ (month/year)
- Project End Date: _______________________ (month/year)
- Number of Team Members: ____
- Nature of the Project (please check): Product Development______ Developing Customized Solution________

The statements below refer to various aspects of that (least successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The statements below refer to the outcomes of that (least successful) project.							
Compared to other projects, the outcomes of that project were more unpredictable	1	2	3	4	5	6	7
Compared to other projects, it took more time to foresee the outcomes of that project	1	2	3	4	5	6	7
Compared to other projects, it was more difficult to understand the outcomes of that project	1	2	3	4	5	6	7

How would you disagree or agree with each of the following statements about the conversion of requirements specifications to software, in that (least successful) project?

Compared to other projects, there was more confusion in that project about developing software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, established procedures and practices could not be relied upon in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, an understandable sequence of steps could not be followed in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7

The statements below refer to any unexpected or novel technological problems that occurred in that (least successful) project

Compared to other projects, that project faced highly unexpected or novel problems related to **software platforms**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software programming languages**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software coding**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software implementation**	1	2	3	4	5	6	7

The statements below refer to the team members who worked on that (least successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The following statements explore how the team members of that (least successful) project synthesized and combined their knowledge, expertise, and skills to achieve project goals.							
Team members combined their individual expertise to jointly solve project-related problems	1	2	3	4	5	6	7
Team members combined their individual perspectives to develop a shared project concepts	1	2	3	4	5	6	7
Team members often gained new insights by sharing their ideas with each other	1	2	3	4	5	6	7
Team members improved their task efficiency by sharing their knowledge with each other	1	2	3	4	5	6	7
The statements below explore if the team members of that (least successful) project acquired and utilized project-related knowledge from external sources such as other teams, individuals, and departments. Such knowledge may have been acquired through personal interactions, over the phone, through e-mails, or through the knowledge management (KM) system.							
If the required knowledge was not available within the team, members acquired that knowledge from external sources	1	2	3	4	5	6	7
Team members often reused code available from other projects	1	2	3	4	5	6	7
Team members often enhanced their knowledge with inputs from external sources	1	2	3	4	5	6	7

The statements below refer to the diversity of backgrounds, expertise, and skills of the team members of that (least successful) project.

Members of that team varied widely in their areas of expertise	1	2	3	4	5	6	7
Members of that team had a variety of different backgrounds and experiences	1	2	3	4	5	6	7
Members of that team had wide-ranging skills and abilities	1	2	3	4	5	6	7

The statements below refer to the working relationships, mutual trust among the team members of that (least successful) project.

The team was characterized by personal relationships among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by high level of reciprocal behavior among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by mutual trust among members at multiple levels	1	2	3	4	5	6	7

The statements below refer to the policy of the team (working on that least successful project) about information coming into the team from external sources (e.g. other teams, individuals, and departments in the company)

The team actively monitored information coming from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively decided what type of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7

The statements below refer to the policy of the team (working on that least successful project) about sharing internal information and other resources with other teams, individuals, and departments in the company

The team avoided releasing internal information to others in the company	1	2	3	4	5	6	7
The team actively controlled the use of its internal resources by others in the company	1	2	3	4	5	6	7

The statements below assess the interdependence of the team (working on that least successful project) with other teams.

The team closely coordinated project-related tasks with other teams	1	2	3	4	5	6	7
The team regularly received project-related information or resources from other teams	1	2	3	4	5	6	7
The team’s performance evaluation was strongly influenced by how well other teams performed	1	2	3	4	5	6	7

The following statements explore how the team (working on that least successful project) used various IT-based systems for different purposes. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.

The team used IT-based systems to coordinate with others in the company	1	2	3	4	5	6	7
The team used IT-based systems to search for project-related knowledge	1	2	3	4	5	6	7
The team used IT-based systems to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

The following statements explore how the team (working on that least successful project) used various IT-based systems as compared to other teams you have led. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.

Compared with other teams you have led, this team uses collaborative systems MORE to internally coordinate project-related tasks	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to coordinate with others in the company	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to search for project-related knowledge	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

Thank You For Your Co-operation!

Q2: 2005 - 2006 Project Leader Survey

Welcome! We invite you to participate in an important research study that examines knowledge usage in software teams. Your opinion is very valuable to the success of this study. The results of this study may help project leaders/ project managers like you gain better understanding of issues such as:

- How your team members use their own knowledge, and knowledge from various external sources to achieve project goals.

- Do characteristics of your team affect your team's knowledge usage.

- Do project characteristics (e.g. complexity) affect you team's knowledge usage.

- How do knowledge management (KM) systems affect your team's knowledge usage

All responses are completely **ANONYMOUS**. There will be no way to identify the person who filled out the survey. With this in mind, please take a few minutes to provide your honest opinion to each statement.
You may decide to withdraw at any time during the research (without penalty). However, after you have provided anonymous information you will be unable to withdraw your responses after participation since there will be no way to identify individual information. Your decision whether or not to participate will not jeopardize your future relations with Auburn University of the Department of Management.

If you have any questions I invite you to ask them now. If you have questions later, please contact either of the persons mentioned below, and we will be happy to answer them.

Nikhil Mehta
Phone: +1-334-844-6534
(mehtan1@auburn.edu)

Dr. Terry Byrd
Phone: +1-334-844-6543
(tbyrd@auburn.edu)

For more information regarding your rights as a research participant you may contact the Office of Human Subjects Research by (334)-844-5966 or e-mail at hsubjec@auburn.edu or IRBChair@auburn.edu .

HAVING READ THE INFORMATION PROVIDED, YOU MUST DECIDE WHETHER TO PARTICIPATE IN THIS RESEARCH PROJECT. IF YOU DECIDE TO PARTICIPATE, THE DATA YOU PROVIDE WILL SERVE AS YOUR AGREEMENT TO DO SO.

Directions

Please read carefully all of the instructions before beginning this questionnaire. The questions in this survey are broken up into two sections labeled "Least Successful Project" and "Most Successful Project." Each section will have specific directions. Please read each statement carefully. For each statement, circle the response that best represents your opinion. Please answer each question.

General Information

What is your gender (please circle): Female Male

Years of experience in the software industry (approximate): ____

Name of your current organization: _________________________

Current Position (please circle): Project Leader Project Manager

SECTION 1 – LEAST SUCCESSFUL PROJECT

Please think of the LEAST SUCCESSFUL PROJECT in your experience as the project leader/ project manager. Some parameters of the least successful project may include, but are not limited to:

- It was not completed within budget and time
- Many of the performance requirements were not met
- Many of the critical risks were not mitigated
- Many of the critical issues were not resolved before delivery
- A project that YOU feel was not completed satisfactorily

Please provide some information about that project:

- Project Start Date: _____________________ (month/year)
- Project End Date: ______________________ (month/year)
- Number of Team Members: ____
- Nature of the Project (please check): Product Development______ Developing Customized Solution________

The statements below refer to various aspects of that (least successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The statements below refer to the outcomes of that (least successful) project.							
Compared to other projects, the outcomes of that project were more unpredictable	1	2	3	4	5	6	7
Compared to other projects, it took more time to foresee the outcomes of that project	1	2	3	4	5	6	7
Compared to other projects, it was more difficult to understand the outcomes of that project	1	2	3	4	5	6	7
How would you disagree or agree with each of the following statements about the conversion of requirements specifications to software, in that (least successful) project?							
Compared to other projects, there was more confusion in that project about developing software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, established procedures and practices could not be relied upon in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, an understandable sequence of steps could not be followed in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7

The statements below refer to any unexpected or novel technological problems that occurred in that (least successful) project							
Compared to other projects, that project faced highly unexpected or novel problems related to **software platforms**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software programming languages**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software coding**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software implementation**	1	2	3	4	5	6	7

The statements below refer to the team members who worked on that (least successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The following statements explore how the team members of that (least successful) project synthesized and combined their knowledge, expertise, and skills to achieve project goals.							
Team members combined their individual expertise to jointly solve project-related problems	1	2	3	4	5	6	7
Team members combined their individual perspectives to develop a shared project concepts	1	2	3	4	5	6	7
Team members often gained new insights by sharing their ideas with each other	1	2	3	4	5	6	7
Team members improved their task efficiency by sharing their knowledge with each other	1	2	3	4	5	6	7

Section 1 – Least Successful Project

The statements below explore if the team members of that (least successful) project acquired and utilized project-related knowledge from external sources such as other teams, individuals, and departments. Such knowledge may have been acquired through personal interactions, over the phone, through e-mails, or through the knowledge management (KM) system.

If the required knowledge was not available within the team, members acquired that knowledge from external sources	1	2	3	4	5	6	7
Team members often reused code available from other projects	1	2	3	4	5	6	7
Team members often enhanced their knowledge with inputs from external sources	1	2	3	4	5	6	7

The statements below refer to the diversity of backgrounds, expertise, and skills of the team members of that (least successful) project.

Members of that team varied widely in their areas of expertise	1	2	3	4	5	6	7
Members of that team had a variety of different backgrounds and experiences	1	2	3	4	5	6	7
Members of that team had wide-ranging skills and abilities	1	2	3	4	5	6	7

The statements below refer to the working relationships, mutual trust among the team members of that (least successful) project.

The team was characterized by personal relationships among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by high level of reciprocal behavior among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by mutual trust among members at multiple levels	1	2	3	4	5	6	7

Section 1 – Least Successful Project

The statements below refer to the policy of the team (working on that least successful project) about information coming into the team from external sources (e.g. other teams, individuals, and departments in the company)							
The team actively monitored information coming from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively decided what type of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively controlled the internal distribution of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The statements below refer to the policy of the team (working on that least successful project) about sharing internal information and other resources with other teams, individuals, and departments in the company							
The team avoided releasing internal information to others in the company	1	2	3	4	5	6	7
The team actively controlled the use of its internal resources by others in the company	1	2	3	4	5	6	7
The statements below assess the interdependence of the team (working on that least successful project) with other teams.							
The team closely coordinated project-related tasks with other teams	1	2	3	4	5	6	7
The team regularly received project-related information or resources from other teams	1	2	3	4	5	6	7
The team’s performance evaluation was strongly influenced by how well other teams performed	1	2	3	4	5	6	7

Section 1 – Least Successful Project

The following statements explore how the team (working on that least successful project) used various IT-based systems for different purposes. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.							
The team used IT-based systems to coordinate with others in the company	1	2	3	4	5	6	7
The team used IT-based systems to search for project-related knowledge	1	2	3	4	5	6	7
The team used IT-based systems to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7
The following statements explore how the team (working on that least successful project) used various IT-based systems <u>as compared to other teams you have led</u>. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.							
Compared with other teams you have led, this team uses collaborative systems MORE to internally coordinate project-related tasks	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to coordinate with others in the company	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to search for project-related knowledge	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

SECTION 2 – MOST SUCCESSFUL PROJECT

Now, please think of the MOST SUCCESSFUL PROJECT in your experience as a project leader/ project manager. Some parameters of the most successful project may include, but are not limited to:

- It was completed within budget and time
- All/Most of the performance requirements were met
- All/Most of the critical risks were mitigated
- Most critical issues were resolved before delivery
- A project that YOU feel was completed most satisfactorily

Please provide some information about that project:

- Project Start Date: _____________________ (month/year)
- Project End Date: ______________________ (month/year)
- Number of Team Members: ____
- Nature of the Project (please check): Product Development______ Developing Customized Solution________

The statements below refer to various aspects of that (most successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The statements below refer to the outcomes of that (most successful) project.							
Compared to other projects, the outcomes of that project were more unpredictable	1	2	3	4	5	6	7
Compared to other projects, it took more time to foresee the outcomes of that project	1	2	3	4	5	6	7
Compared to other projects, it was more difficult to understand the outcomes of that project	1	2	3	4	5	6	7

Section 2 – Most Successful Project

How would you disagree or agree with each of the following statements about the conversion of requirements specifications to software, in that (most successful) project?							
Compared to other projects, there was more confusion in that project about developing software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, established procedures and practices could not be relied upon in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7
Compared to other projects, an understandable sequence of steps could not be followed in that project to develop software that would meet the requirements specifications	1	2	3	4	5	6	7
The statements below refer to any unexpected or novel technological problems that occurred in that (most successful) project							
Compared to other projects, that project faced highly unexpected or novel problems related to **software platforms**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software programming languages**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software coding**	1	2	3	4	5	6	7
Compared to other projects, that project faced highly unexpected or novel problems related to **software implementation**	1	2	3	4	5	6	7

The statements below refer to the team members who worked on that (most successful) project. Please **circle** your answer to each statement using the following rating scale:

1 = you ***strongly disagree*** with the statement
2 = you ***disagree*** with the statement
3 = you ***slightly disagree*** with the statement
4 = you are ***neutral*** about the statement
5 = you ***slightly agree*** with the statement
6 = you ***agree*** with the statement
7 = you ***strongly agree*** with the statement

The following statements explore how the team members of that (most successful) project synthesized and combined their knowledge, expertise, and skills to achieve project goals.

Team members combined their individual expertise to jointly solve project-related problems	1	2	3	4	5	6	7
Team members combined their individual perspectives to develop a shared project concepts	1	2	3	4	5	6	7
Team members often gained new insights by sharing their ideas with each other	1	2	3	4	5	6	7
Team members improved their task efficiency by sharing their knowledge with each other	1	2	3	4	5	6	7

The statements below explore if the team members of that (most successful) project acquired and utilized project-related knowledge from external sources such as other teams, individuals, and departments. Such knowledge may have been acquired through personal interactions, over the phone, through e-mails, or through the knowledge management (KM) system.

If the required knowledge was not available within the team, members acquired that knowledge from external sources	1	2	3	4	5	6	7
Team members often reused code available from other projects	1	2	3	4	5	6	7
Team members often enhanced their knowledge with inputs from external sources	1	2	3	4	5	6	7

Section 2 – Most Successful Project

The statements below refer to the diversity of backgrounds, expertise, and skills of the team members of that (most successful) project.							
Members of that team varied widely in their areas of expertise	1	2	3	4	5	6	7
Members of that team had a variety of different backgrounds and experiences	1	2	3	4	5	6	7
Members of that team had wide-ranging skills and abilities	1	2	3	4	5	6	7
The statements below refer to the working relationships, mutual trust among the team members of that (most successful) project.							
The team was characterized by personal relationships among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by high level of reciprocal behavior among members at multiple levels	1	2	3	4	5	6	7
The team was characterized by mutual trust among members at multiple levels	1	2	3	4	5	6	7
The statements below refer to the policy of the team (working on that most successful project) about information coming into the team from external sources (e.g. other teams, individuals, and departments in the company)							
The team actively monitored information coming from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7
The team actively decided what type of information to acquire from external sources such as other teams, individuals, and departments	1	2	3	4	5	6	7

Section 2 – Most Successful Project

The statements below refer to the policy of the team (working on that most successful project) about sharing internal information and other resources with other teams, individuals, and departments in the company							
The team avoided releasing internal information to others in the company	1	2	3	4	5	6	7
The team actively controlled the use of its internal resources by others in the company	1	2	3	4	5	6	7
The statements below assess the interdependence of the team (working on that most successful project) with other teams.							
The team closely coordinated project-related tasks with other teams	1	2	3	4	5	6	7
The team regularly received project-related information or resources from other teams	1	2	3	4	5	6	7
The team's performance evaluation was strongly influenced by how well other teams performed	1	2	3	4	5	6	7
The following statements explore how the team (working on that most successful project) used various IT-based systems for different purposes. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.							
The team used IT-based systems to coordinate with others in the company	1	2	3	4	5	6	7
The team used IT-based systems to search for project-related knowledge	1	2	3	4	5	6	7
The team used IT-based systems to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

Section 2 – Most Successful Project

The following statements explore how the team (working on that most successful project) used various IT-based systems as compared to other teams you have led. Examples of IT-based systems include Collaborative systems (e-mail, telephone, group support systems) and the knowledge management (KM) system.

Compared with other teams you have led, this team uses collaborative systems MORE to internally coordinate project-related tasks	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to coordinate with others in the company	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to search for project-related knowledge	1	2	3	4	5	6	7
Compared to other teams you have led, the team used IT-based systems MORE to retrieve project-related knowledge (e.g. downloading a document from the KM system)	1	2	3	4	5	6	7

Thank You For Your Co-operation!