MEHTA
Knowledge Integration in Software Teams

DECISION SCIENCES INSTITUTE
Relational Capital and Knowledge Heterogeneity in Software Teams: Effects on Knowledge Integration

(Full Paper Submission)

Nikhil Mehta
University of Northern Iowa
Nikhil.mehta@uni.edu

ABSTRACT

We examine the role of knowledge heterogeneity and relational capital on knowledge integration in software teams. Knowledge heterogeneity and relational capital are key compositional characteristics of software teams. While both knowledge heterogeneity and relational capital have been examined in teams, there is a need to better understand their relationship to critical parameters that could influence performance of software teams. Based on absorptive capacity and social capital literatures, we hypothesize and test relationships between these team characteristics and internal and external knowledge integration. The hypotheses are tested using data from 150 project leaders from sixteen CMM Level 5 software companies.

KEYWORDS: Knowledge integration, Software teams, Knowledge heterogeneity, Relational capital

INTRODUCTION

Software development is a knowledge intensive process, and teams leverage their knowledge resources by integrating them into collective project-level knowledge (Okhuysen & Eisenhardt, 2002). Two forms of knowledge integration—internal and external have been identified as critical to software project performance (Tiwana, Bharadwaj, & Sambamurthy, 2003). While internal knowledge integration refers to active synthesis of team’s internal skill-sets, know-how, and expertise, and their application to various project tasks (Liu, 2006; Mitchell, 2006), external knowledge integration refers to importing relevant information, know-how, experience, insights, and feedback from experts and other sources external to the team, and synthesizing them into the decisions and actions of team members (Gray & Meister, 2004; Haas, 2006). Thus, internal knowledge integration, in conjunction with external knowledge integration, helps software teams develop a robust body of project knowledge.

The importance of internal and external knowledge integration in software development has motivated researchers to understand various team characteristics that impact integration. Two key team characteristics include team’s knowledge and relational capital.

A team’s knowledge refers to the heterogeneity of its knowledge resources, especially the backgrounds, skill-sets and expertise of its members. Software teams with more knowledge typically have members with diverse backgrounds, skill-sets, and technical and functional expertise (Tiwana & McLean, 2005).

Given the importance of knowledge as a key resource of software development (Walz, Elam, & Curtis, 1993), knowledge of the team may influence knowledge integration (Tiwana & McLean,
2005). Drawing parallels from the absorptive capacity literature, “…there are benefits to diversity of knowledge structures across individuals that parallel the benefits of diversity of knowledge within individuals (Cohen & Levinthal, 1990 pg. 133).” Team’s knowledge heterogeneity captures the key intra-team characteristics of the composition of its internal knowledge resources, through which those resources are applied to software development activities, and through which those resources influence team’s integration of additional knowledge inputs from external sources (Tiwana & McLean, 2005). Prior studies have only examined the impact of knowledge heterogeneity on internal knowledge integration (Tiwana & McLean, 2005). To develop a more robust understanding of heterogeneity’s impact on key software development processes, we also examine its impact on external knowledge integration.

The second team characteristic being examined in this study is team’s relational capital. A team’s relational capital represents its social context, and refers to the mutual trust and close working relationships among its members, and their propensity to reciprocate each other’s efforts. Software development is as much a social process as it is technical (Kirsch, 1997), and a software team’s relational capital may have critical influence on team’s key processes such as internal and external knowledge integration. Prior research has investigated the impact of relational capital on internal knowledge integration (Tiwana & McLean, 2005). More recent studies have examined this relationship among student teams (Robert, Dennis, & Ahuja, 2008). We investigate the impact of relational capital on both internal and external knowledge integration in software development teams.

Our study makes two key contributions to the literature. First, we extend theoretical developments in the areas of team-level absorptive capacity, social capital and knowledge integration literatures. Second, we validate operationalization of difficult-to-measure constructs for future examination of these theories.

Data for this research were obtained from 150 software project leaders in 16 capability maturity model (CMM) Level-5 [CMM Level-5 certification is a software development process standard widely adopted by the IT industry] software companies in India. Results show that both relational capital and knowledge heterogeneity have a positive influence on teams’ internal as well as external knowledge integration. We now present the research hypotheses, followed by empirical analysis and results. Finally, we discuss the significance of our findings.

THEORY AND HYPOTHESES

The term integration originally refers to “the quality of the state of collaboration that exists among departments which are required to achieve unity of effort by the demands of the environment” (Lawrence & Lorsch, 1967, p.11). Software teams need to integrate individually-held knowledge from multiple sources. In the absence of effective integration of individual knowledge inputs, teams run the risk of building their ignorance into the software solution (Tiwana & McLean, 2005). Teams also need to integrate knowledge about business domains and software development technologies. This process of combining diverse streams of specialized knowledge is defined as knowledge integration (Nonaka, 1994; Grant, 1996).

Prior research suggests the presence of internal knowledge integration in software teams. Teams combine internally available stocks of know-how, skills, and abilities into team-level expertise, and utilize this expertise to significantly improve software outcomes (Faraj & Sproull, 2000). Recent perspectives suggest that teams provide an appropriate environment for integration of individually-owned skills, know-how, and expertise into a team-level systemic knowledge base (Grant, 1996; Nahapiet & Ghoshal, 1998). Team members integrate
Knowledge Integration in Software Teams

individually-owned knowledge through communicating verbally about the project, exchanging tangible artifacts, coordinating their expertise, and sharing information about who knows what (Rulke & Galaskiewicz, 2000). We define this active combination of diverse streams of specialized knowledge as internal knowledge integration.

In addition to internal knowledge integration, teams also integrate knowledge from external sources at various stages of the project (Walz et al., 1993; Tiwana, 2004). The expectation that accessing external knowledge should be beneficial for software teams is consistent with organizational learning theories which propose that learning from outside sources can improve organizational performance (Argote, 1999; Carlile & Rebentisch, 2003). Prior studies have observed that teams improve project performance by integrating project-related information, insights, and feedback from their interpersonal networks inside as well as outside the organization (Cummings, 2004; Kraut & Streeter, 1995).

Knowledge Heterogeneity

For all their benefits, internal and external knowledge integration are challenging activities (Zellmer-Bruhn, 2003), especially in software teams, which typically constitute of, and have access to, diverse knowledge resources (Curtis, Krasner, & Iscoe, 1988). Team members are primary repositories of a team’s knowledge (Argote, 1999), as they bring in diverse sets of expertise and skills (Tiwana & McLean, 2005). They also bring with them access to diverse external knowledge inputs in their respective domains (Ancona & Caldwell, 1992). The breadth of members’ expertise may influence team’s internal knowledge integration (Cummings, 2004). Prior research has observed that teams with members holding diverse sets of expertise and skills may integrate them to develop better project outcomes (Curtis, Krasner, & Iscoe, 1988; Nahapiet & Ghoshal, 1998). Interestingly, another body of research suggests that team members from different technical and functional domains may lack a sufficient overlap in their understanding of each other’s uniquely held knowledge, which may affect the team’s knowledge integration negatively (Rulke & Galaskiewicz, 2000; Tiwana & McLean, 2005). Sometimes, team members are also less willing to share information with individuals perceived as different (Mesmer-Magnus & DeChurch, 2009). These apparent contradictions in prior research about the influence of knowledge heterogeneity on knowledge integration make it an interesting issue for further examination.

Relational Capital

Another interesting issue for examination emerges from the social aspect of knowledge integration processes (Kogut & Zander, 1992). Because they are predominantly social processes, team’s social context is expected to have a critical influence on their effectiveness. In this study, we focus on the relational dimension of team’s social context, which relates to the nature and quality of relationships among team members (Nahapiet & Ghoshal, 1998). Referred to as team’s relational capital, it embodies mutual trust and closeness of relationships among team members (Gulati, 1995; Kale, Singh, & Perlmutter, 2000; Lewicki & Bunker, 1995). Such goodwill among team members generates group solidarity and a positive give-and-take attitude (Robert et al., 2008).

Team members may exchange knowledge relatively easily with co-workers with whom they share close working relationships (Teigland & Wasko, 2003). High level of mutual trust also helps overcome free-riding behavior, which may promote generalized reciprocity of know-how and expertise among team members (Bradach & Eccles, 1989). Prior studies have observed
that high level of relational capital improves knowledge exchange among scientists (Bouty, 2000).

On the other hand, team members with low relational capital have difficulty in reciprocating voluntarily with their uniquely held knowledge, beyond their knowledge commitments to the project (Tiwana & McLean, 2005). To summarize, effective knowledge integration in software teams may rely on team’s relational capital.

In the following sections, we develop hypotheses to explore how knowledge heterogeneity and relational capital influence internal and external knowledge integration in software teams. Figure 1 illustrates the research model.

**Figure 1: Research Model**

![Research Model Diagram]

**Knowledge Heterogeneity and Knowledge Integration**

A team’s knowledge heterogeneity fulfills a fundamental pre-condition for its knowledge integration – the presence of assorted knowledge among team members (Moran & Ghoshal, 1996).

Software project outcomes are contingent on team members contributing different but complementary information. In heterogeneous teams, people are likely to expect that others do not possess the unique knowledge they do, thus everyone needs to contribute their distinctive knowledge to accomplish the project goals. Such expectations among team members to provide their unique knowledge inputs, and to rely on others for complementary inputs, may provide the groundwork for better integration of team’s knowledge resources (Hollingshead, 2001; Lewis, 2004).

Given the expectation that project outcomes are contingent on everyone’s unique contribution, members of heterogeneous software teams also tend to acquire and learn more information in their own domains (Wittnebaum, Vaughan, & Stasser, 1998; Hollingshead, 2001).

Furthermore, diverse teams have experts in multiple domains, and thus have better capacity to integrate knowledge in multiple domains (Cohen & Levinthal, 1990). Additionally, given the diversity of team members’ backgrounds, heterogeneous teams may have access to diverse external networks, which can be utilized to integrate relevant external knowledge inputs. Therefore, a team with diverse knowledge base is expected to have a more elaborate schema to understand project-related issues, develop more diverse alternatives to project problems, and to apply knowledge resources more creatively than homogeneous teams (Tiwana & McLean, 2005).
Teams with knowledge heterogeneity also typically try to develop a shared context for smoother project execution. To develop a shared understanding of project-related issues, team members actively integrate diverse perspectives and ideas, thereby facilitating knowledge integration.

To summarize, teams with more knowledge heterogeneity, as compared to homogeneous teams, not only have more capacity and motivation to integrate their internal knowledge resources, but also have access to diverse external knowledge inputs. Based on these arguments, we hypothesize:

**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.

### Relational Capital and Knowledge Integration

A team with extensive relational capital will typically have high level of mutual trust among its members. Trust may reduce the mutual suspicion among members about free-riding of their uniquely held knowledge (Davenport & Prusak, 1998), thus providing them justification for open knowledge sharing with others (Tsai & Ghoshal, 1998). High level of mutual trust also motivates team members to take risks in exchanging their uniquely held knowledge (Nahapiet, 1996; Ring & Van de Ven, 1994), thus enabling the team to try creative combinations of its internal knowledge resources (Nahapiet & Ghoshal, 1998). Trust also increases credibility of information exchanged among team members, thus increasing the possibility of more liberal use of each other’s information inputs (Dennis, 1996). This, combined with improved interpersonal communication due to high mutual trust (Misztal, 1996), creates a climate conducive to internal knowledge integration.

A team with high relational capital also typically has strong interpersonal ties among its members. Strong interpersonal ties improve interpersonal communication, thereby improving the awareness of various external knowledge resources available to the team (Ko, Kirsch, & King, 2005). Strong interpersonal communication also eases the exchange and integration of more sticky and tacit knowledge inputs (Tiwana & McLean, 2005; Marsdeb, 1990).

Teams with high relational capital also have among their members a generalized reciprocity (Kale et al., 2000). Thus, members of such teams are more likely to share their uniquely held knowledge with each other. Reciprocity may also improve external knowledge integration since team members may be more open to acquiring knowledge inputs from external sources as a reciprocal gesture towards their colleagues.

In light of the above discussion, we propose that a team’s relational capital will improves its inherent transparency and openness, thereby supporting its internal and external knowledge integration. Thus:

**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.
SCALE DEVELOPMENT AND DATA COLLECTION

All research variables were measured using multi-item seven-point Likert scales. The items were subjected to the conceptual validation exercise based on recommendations by Moore and Benbasat (1991). Four sets of the items were printed on separate cards and were given, unordered, to four IS doctoral students. 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 items of the same construct. The four sorters correctly assigned 96 percent of the items to their intended constructs; the inter-rater reliability was 0.98.

Based on the feedback from this exercise, two items for internal knowledge integration were dropped. This was followed by a pilot study involving 50 project leaders in a CMM Level 5 firm that was not included in final data collection. The pre-test data helped in the final formulation of constructs and in testing the validity and reliability of various scales. Construct validity was established by conducting exploratory factor analysis using Principal Components Analysis (PCA) extraction method with Varimax rotation. Reliability was calculated for each group of items using Cronbach’s alpha coefficient, and they were all greater than 0.7 (Cronbach, 1951). The instrument validation ensured that items were unambiguous and relevant to software development context. The final items and their sources are presented in the Appendix.

Data were collected through an online questionnaire-based survey. Links to the questionnaire were forwarded to 225 project leaders in sixteen CMM Level-5 certified software firms. CMM Level 5 ensured that the software development processes across all participating firms were highly uniform and mature. Other than process maturity, CMM levels have not been examined for their impact on knowledge integration in software teams, so it will be difficult to say if CMM Level 5 companies have high internal and external knowledge integration.

Project leaders were chosen as they have an overall understanding of various project-related issues. Because the outcome of a project may affect a respondent’s perception, each respondent was asked to answer the questions twice, once in response to a project with the most successful outcome, and once for a project with the least successful outcome. Half of the respondents answered the questions beginning with the most successful project, and half answered beginning with the least successful project. Each style of questionnaire was alternatively assigned as project leaders entered the website. In addition, demographic-based questions were included. Only project leaders with prior experience with, or currently involved in, projects developing business application software or packaged software were selected.

One hundred and fifty respondents filled the questionnaire (response rate of 66.68%). Eighty respondents filled the questionnaire in most to least order, while 70 respondents filled it in the opposite order. Of the total respondents, 22% were females and 78% were males. Project leaders had average industry experience of 8.9 years. Average project duration was 12.16 months with an average of 16.9 members per team. Comparisons of the sample demographics with population figures from an industry standard research report (NASSCOM-Hewitt, 2005) yielded no significant differences, suggesting that non-response bias was unlikely to be an issue (Armstrong & Overton, 1977). Harman’s one factor test was conducted to test for common method bias. All the variables were entered into an exploratory factor analysis, using unrotated principal components factor analysis (Podsakoff et al., 2003). The analysis revealed the presence of five distinct factors with eigenvalues greater than 1.0, rather than a single factor.
The five factors together accounted for about 69 percent of the total variance; the largest factor did not account for a majority of the variance (33%). Thus, no general factor was apparent. To reduce the likelihood that other variables confounding the relations examined in this research, our analysis controlled for project duration, project team size, project leader’s experience, project interdependence, and project uncertainty. Project duration was measured in months, while project leader’s experience was measured in number of years worked in the software development industry. Team size was measured in number of team members. 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. Project uncertainty is defined as the lack of critical knowledge inputs about the project. For the purpose of this study, we measured requirements uncertainty, which is considered most critical to software development. Both project uncertainty and project interdependence were measured using two separate 3-item scales modified from prior literature.

Constructor Reliability and Validity

As presented in Table 1 below, all constructs demonstrated adequate reliability with Cronbach’s alpha scores well above the recommended cut-off of 0.70 and averaging 0.82 across all constructs. Composite reliabilities ($\rho_c$), which are a more accurate measure of internal consistency as they avoid the assumption of equal weighting of items, were even higher.

<table>
<thead>
<tr>
<th>VARIABLE NAME</th>
<th>CRONBACH’S ALPHA</th>
<th>COMPOSITE RELIABILITY</th>
<th>AVE VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Interdependence</td>
<td>.74</td>
<td>.85</td>
<td>.66</td>
</tr>
<tr>
<td>Project Uncertainty</td>
<td>.72</td>
<td>.83</td>
<td>.62</td>
</tr>
<tr>
<td>Knowledge Heterogeneity</td>
<td>.80</td>
<td>.88</td>
<td>.71</td>
</tr>
<tr>
<td>Relational Capital</td>
<td>.76</td>
<td>.85</td>
<td>.66</td>
</tr>
<tr>
<td>Internal Knowledge Integration</td>
<td>.91</td>
<td>.94</td>
<td>.80</td>
</tr>
<tr>
<td>External Knowledge Integration</td>
<td>.72</td>
<td>.84</td>
<td>.65</td>
</tr>
</tbody>
</table>

Convergent validity was supported by all the item loadings on respective constructs being statistically significant (minimum $t=15.50, p<0.00$) (Fornell & Larker, 1981). Discriminant validity was measured by calculating average variance extracted (AVE) values for all constructs (see Table 1). These were greater than the recommended value of 0.5 (Chin, 1998). The square root of AVE for each construct was greater than the construct’s correlation with other constructs further demonstrating discriminant validity (Fornell & Larker, 1981) (see Table 2).

ANALYSIS AND RESULTS

Descriptive Statistics and Correlations

Descriptive statistics and correlations for the study variables are provided in Table 2. Internal knowledge integration had a negative correlation with project uncertainty ($r = -.25, p<0.01$), and a positively correlation with project interdependence ($r = 0.32, p<0.01$) and with project duration ($r = 0.13, p <0.51$). External knowledge integration also had a negative correlation with project uncertainty ($r = -.28, p<0.01$), and a positive correlation with project interdependence ($r = 0.36, p<0.01$). Also, team size had a positive correlation with project duration ($r = 0.20, p <0.01$).
Hypotheses Testing

Hypotheses were tested using fixed effects regression with clustered heteroskedasticity corrected standard errors (White, 1980). Since there are multiple projects per firm, we used fixed effects by firm and clustered standard errors to remove any firm-specific effects that could impact the results. After mean centering the independent variables (Aiken & West, 1991), we introduced into a regression equation the control variables (Model 1) and independent variables, namely knowledge heterogeneity and relational capital (Model 2). Separate models were run for IKI and EKI.

Results show that among the control variables, project interdependence was positively related to both internal and external knowledge integration, while project uncertainty had a significant negative impact on internal as well as external knowledge integration (see Table 3). However, project leader experience was found to be related to external knowledge integration only. The overall R-square for the controls-only model for IKI was 0.17, and for EKI was 0.20.

In model 2, knowledge heterogeneity and relational capital were added. We found that knowledge heterogeneity was positive and significant in its relationship to IKI (p<0.001) and EKI (p<0.001), thereby supporting Hypotheses 1a and 1b. Relational capital was also positive and significant in its relationship to IKI (p <0.001) and EKI (p <0.001), thereby supporting Hypotheses 2a and 2b. The R-square for model 2 (which included all the controls and independent variable) for IKI was 0.41, and for EKI was 0.28, demonstrating that the final models were a good fit with the data and adequately explained the overall variance in the data. Table 3 presents the hierarchical regression analysis results for IKI and EKI respectively.

Table 2: Descriptive Statistics and Correlations

<table>
<thead>
<tr>
<th>S.NO.</th>
<th>VARIABLE</th>
<th>MEAN (SD)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Project Duration</td>
<td>12.16 (9.34)</td>
<td>na</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Team Size</td>
<td>16.24 (20.84)</td>
<td>.20**</td>
<td>na</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Member Experience</td>
<td>8.90 (3.19)</td>
<td>.10</td>
<td>.07</td>
<td>na</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Project Interdependence</td>
<td>4.38 (1.25)</td>
<td>.04</td>
<td>.05</td>
<td>.04</td>
<td>.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Project Uncertainty</td>
<td>3.92 (1.16)</td>
<td>-.11</td>
<td>.10</td>
<td>-.10</td>
<td>-.10</td>
<td>.78</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Knowledge Heterogeneity</td>
<td>4.88 (1.31)</td>
<td>.13*</td>
<td>.09</td>
<td>-.01</td>
<td>.39**</td>
<td>.07</td>
<td>.84</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Relational Capital</td>
<td>5.01 (1.10)</td>
<td>.10</td>
<td>.09</td>
<td>.02</td>
<td>.26**</td>
<td>-.11*</td>
<td>.26**</td>
<td>.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Internal Knowledge Integration</td>
<td>5.31 (1.22)</td>
<td>.13*</td>
<td>.02</td>
<td>-.05</td>
<td>.32**</td>
<td>-.25**</td>
<td>.33**</td>
<td>.55**</td>
<td>.89</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>External Knowledge Integration</td>
<td>4.42 (1.28)</td>
<td>.06</td>
<td>.04</td>
<td>-.06</td>
<td>.36**</td>
<td>-.28**</td>
<td>.31**</td>
<td>.33**</td>
<td>.57**</td>
<td>.80</td>
</tr>
</tbody>
</table>

**Correlation is significant at the 0.01 level (2-tailed); * Correlation is significant at the 0.05 level (2-tailed); Numbers on the diagonal are square root values of average variance extracted for each construct.
Table 3: Results of Hierarchical Regression Analyses

<table>
<thead>
<tr>
<th>VARIABLE NAME</th>
<th>INTERNAL KNOWLEDGE INTEGRATION (IKI)</th>
<th>EXTERNAL KNOWLEDGE INTEGRATION (EKI)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model 1</td>
<td>Model 2</td>
</tr>
<tr>
<td>Control Variable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Duration</td>
<td>.10</td>
<td>.05</td>
</tr>
<tr>
<td>Team Size</td>
<td>.01</td>
<td>-.03</td>
</tr>
<tr>
<td>Project Leader Experience</td>
<td>-.10</td>
<td>-.09</td>
</tr>
<tr>
<td>Project Interdependence</td>
<td>.30***</td>
<td>.12*</td>
</tr>
<tr>
<td>Project Uncertainty</td>
<td>-.22***</td>
<td>-.20***</td>
</tr>
<tr>
<td>Independent Variable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knowledge Heterogeneity</td>
<td>.17***</td>
<td>.19***</td>
</tr>
<tr>
<td>Relational Capital</td>
<td>.45***</td>
<td>.20***</td>
</tr>
</tbody>
</table>

N: 300 300 300 300
F-value: 12.44 28.93 15.08 16.41
Δ F: 58.13 15.93
R²: .17 .41 .20 .28
Δ R²: .23 .08

Significant at the *p ≤ 0.10; ** p ≤ 0.05; ***p ≤ 0.01 levels – one tailed

DISCUSSION AND CONCLUSION

The study examined the influence of knowledge heterogeneity and relational capital on internal and external knowledge integration in software teams. To the best of our knowledge, this is among the first studies to examine how a software team’s breadth of expertise and its social context influence its internal and external knowledge integration. This study provides detailed arguments and empirical evidence supporting the respective relationships. The results of this study enhance our understanding of knowledge integration, which is a key process affecting software development performance. The results also pave way for future research examining the interface between team processes and its social and intellectual capabilities.

As predicted, both – software team’s knowledge heterogeneity and relational capital have significant positive impact on team’s internal and external knowledge integration. Results suggest that team’s internal as well as external knowledge integration improve as the heterogeneity of expertise and skills among team members increases. Thus, heterogeneity not only improves the availability of both internal and external knowledge inputs (Smith, Collins, & Clark, 2005), but also enhances team’s capacity to integrate those inputs (Cohen & Levinthal, 1990). This is an interesting addition to the absorptive capacity literature. Future research can examine if team’s knowledge heterogeneity also affects other processes, such as team learning, boundary spanning, and ultimately, team performance.

Results also imply that teams with high levels of relational capital better integrate their internal and external knowledge resources than teams with low relational capital. Thus, it seems like higher relational capital helps team members develop a better shared understanding of project’s knowledge requirements, and how to best fulfill those requirements by integrating team’s internal and external knowledge resources (Ko et al., 2005).
From a practice perspective, our findings have key implications for project leaders. It is worthwhile for them to know the extent of knowledge heterogeneity and relational capital in their teams. Software project leaders seeking to improve project outcomes would do well to consciously manage team’s knowledge heterogeneity and to develop high-levels of relational capital in their teams.

Although this research highlights key issues in understanding the influence of knowledge heterogeneity and relational capital on internal and external knowledge integration in software projects, more research is needed to explore different dimensions of these relationships. First, it is possible that some project characteristics, such as uncertainty and interdependence (used as controls in this study), may serve as moderators of the effects of knowledge heterogeneity and relational capital on internal and external knowledge integration. For example, the risk posed by project uncertainty may confound a team’s efforts to manage its knowledge heterogeneity and develop its relational capital (Barki, Revard, & Talbot, 1993; Keil, Cule, Lyttinen, & Schmidt, 1998). Thus, it would be interesting to see if the effects examined in this study varied under conditions of uncertainty and interdependence.

Second, we collected data from Indian firms working on outsourced projects, and it would be worthwhile to see if the results hold for internal projects of companies, or for projects outsourced to other countries.

Finally, case studies examining teams actually managing their knowledge heterogeneity, as well as developing high level of relational capital would offer deeper insights.
APPENDIX

Questionnaire Items and Their Sources

<table>
<thead>
<tr>
<th>Constructs and Items</th>
<th>Items Adapted From</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Internal Knowledge Integration</strong></td>
<td>Tiwana et al., 2003; Tiwana &amp; McLean, 2005</td>
</tr>
<tr>
<td>1. Team members combined their individual expertise to jointly solve project-related problems.</td>
<td></td>
</tr>
<tr>
<td>2. Team members combined their individual perspectives to develop shared project concepts.</td>
<td></td>
</tr>
<tr>
<td>3. Team members improved their task efficiency by sharing their knowledge with each other</td>
<td></td>
</tr>
<tr>
<td>4. Team members often gained new insights by sharing their ideas with each other</td>
<td></td>
</tr>
<tr>
<td><strong>External Knowledge Integration</strong></td>
<td>Templeton et al., 2002; Norman, 2004</td>
</tr>
<tr>
<td>5. If the required knowledge is not available within the team, members acquired that knowledge from external sources</td>
<td></td>
</tr>
<tr>
<td>6. Team members often reuse code available from other sources</td>
<td></td>
</tr>
<tr>
<td>7. Team members frequently enhanced their knowledge with inputs from external sources</td>
<td></td>
</tr>
<tr>
<td><strong>Knowledge Heterogeneity</strong></td>
<td>Campion et al. 1993, 1996</td>
</tr>
<tr>
<td>8. Members of this team vary widely in their areas of expertise</td>
<td></td>
</tr>
<tr>
<td>9. Members of this team have a variety of different background and experiences</td>
<td></td>
</tr>
<tr>
<td>10. Members of this team have wide-ranging skills and abilities</td>
<td></td>
</tr>
<tr>
<td><strong>Relational Capital</strong></td>
<td>Tiwana &amp; McLean, 2005; Kale et al., 2000</td>
</tr>
<tr>
<td>11. This team is characterized by personal friendship members at multiple levels</td>
<td></td>
</tr>
<tr>
<td>12. This team is characterized by high reciprocal behavior among members at multiple levels</td>
<td></td>
</tr>
<tr>
<td>13. This team is characterized by mutual trust among members at multiple levels</td>
<td></td>
</tr>
<tr>
<td><strong>Project Interdependence</strong></td>
<td>Andres &amp; Zmud, 2001; Campion et al., 1993; Pearce &amp; Gregerson, 1991</td>
</tr>
<tr>
<td>14. The team closely coordinated project-related tasks with other teams</td>
<td></td>
</tr>
<tr>
<td>15. The team regularly received project-related information from other teams</td>
<td></td>
</tr>
<tr>
<td>16. The team’s performance was strongly influenced by how well other teams performed</td>
<td></td>
</tr>
<tr>
<td><strong>Project Uncertainty</strong></td>
<td>Nidumolu, 1995</td>
</tr>
<tr>
<td>17. Compared to other projects you have worked on, requirements for the project fluctuated quite a bit</td>
<td></td>
</tr>
<tr>
<td>18. Compared to other projects you have worked on, clients for the project had more differences among themselves about the requirements</td>
<td></td>
</tr>
<tr>
<td>19. Compared to other projects you have worked on, a lot of effort was spent in reconciling clients’ requirements for the project</td>
<td></td>
</tr>
</tbody>
</table>
REFERENCES


