8th Pre-ICIS International Research Workshop on Information Technology Project Management (IRWITPM 2013)
Workshop Proceedings

12/14/2013
AIS Special Interest Group on Information Technology Project Management
Contents

Workshop Welcome ................................................................................................................................... 3
Workshop Committee ................................................................................................................................. 3
SIG ITProjMgmt Officers .......................................................................................................................... 3
Special Thanks .......................................................................................................................................... 4
Reviewer Thanks ........................................................................................................................................ 4
SponsorThanks .......................................................................................................................................... 4

Completed Research Papers

Co-sourcing in software development offshoring: A case study of risk perception and alleviation by Bjarne Rerup Schlichter and John Stouby Persson ........................................................ .................................................. 5
Conflict Resolution Satisfaction and IS Program Effectiveness: Constructive Controversy Theory by James J. Jiang, Victor Chen, Sigi Goode, and Walter Fernandez .................................................................................................. 15
Delivering IT PMO Value: Understanding Stakeholder Perceptions & Expectations by Siddhartha Arumugam, Judy McKay, and Nicholas Grainger ........................................................................................................... 27
Highlighting Communication Activities and Inefficiencies Between Agile vs. Waterfall Methods: An Agent Based Model of Knowledge Sharing by Meghann L. Drury-Grogan and Deanna M. Kennedy ...................................................... 46
Leadership, Regulatory Focus and Project Performance by Chia-Yu Lai and Jack Shih-Chieh Hsu .............. 58
Lean Software Development: Evaluating Techniques for Parsimonious Feature Selection of Evolving Information Systems Products by Adarsh Kumar Kakar, Joanne E. Hale, David P. Hale ................................................................. 71
The Nature of Adherence to Planning – Systematic Review of Factors Influencing its Suitability as Criterion for IS Project Success by Dirk Basten and Ali Sunyaev ............................................................................................... 88
*The Planning Fallacy as an Explanation for Over-Requirement in Software Development by Ofira Shmueli, Nava Pliskin, and Lior Fink ............................................................................................................................... 100
The Structures of Computation: A Distributed Cognitive Analysis of Requirements Evolution in Diverse Software Development Environments by Sean Hansen, Amol Kharabe, and Kalle Lyytinen .......................................................... 111
Toward a Typological Theory of Information System Project Team Management Styles by Simon Bourdeau and Henri Barki .............................................................................................................................................. 125
Understanding the Information Technology Growth Options: Effects of Gender and Experience on Option Exercise Decisions by Sarah S. Khan, Ram L. Kumar, Kexin Zhao, and Antonis Stylianou ................................................................. 140

Research In Progress Papers

Agile Information Systems Development Teams: Is Empowerment Taken for Granted? by Behnaz Gholami .... 151
The Impact of Group Cohesiveness on Decision-Making Outcomes under Conditions of Challenging and Hindrance Time Pressure by Garry Lohan, Tom Acton, and Kieran Conboy ............................................................................................................................... 159
Project Management with a Purpose: Delivering Project Management Education through Service Learning by Christina N. Outlay and R. A. Mendoza ........................................................................................................ 167

Note: * marks best paper award.
WORKSHOP WELCOME

Welcome to Milan for the 8th International Research Workshop on IT Project Management (IRWITPM 2013) sponsored by the AIS Special Interest Group for Information Technology Project Management. This year’s workshop includes eleven completed research papers and four research-in-progress papers.

It is my sincere hope that this year’s workshop will continue to facilitate the exchange of ideas between IT project management researchers, educators, and practitioners from around the world and provide an opportunity for us to renew and extend our network of IT project management colleagues.

I would also like to take this opportunity to thank the workshop authors, reviewers, participants, organizers, and sponsors. Without these individuals, our eighth annual meeting would not have been possible. Thank you again for engaging with this AIS SIG, and I hope you continue to participate in its activities.

Alanah Mitchell, Appalachian State University, SIGITProjMgmt President

WORKSHOP COMMITTEE

Alanah Mitchell, Appalachian State University (Workshop Program Co-Chair)

Stacie Petter, University of Nebraska at Omaha (Workshop Program Co-Chair)

SIG ITPROJMGMT OFFICERS

Alanah Mitchell, Appalachian State University (President)

Michael Cuellar, North Carolina Central University (Secretary)

Radu Vlas, University of Houston – Clear Lake (Treasurer)

John Tripp, Baylor University (Communications and Publicity Chair)

Cecil Chua, University of Auckland (Membership and Community Relations Chair)

Deepak Khazanchi, University of Nebraska at Omaha (Founder)
SPECIAL THANKS

Reviewer Thanks

We would like to give a special thank you to our reviewers this year whose developmental reviews are critical to the success of this workshop.

Sponsor Thanks

We would like to extend a thank you to our sponsor for their financial support.
Co-sourcing in software development offshoring: A case study of risk perception and alleviation

Bjarne Rerup Schlichter*
Aarhus University, Denmark
brs@asb.dk

John Stouby Persson*
Aalborg University, Denmark
john@cs.aau.dk

*) The authors contributed equally to the paper

ABSTRACT

Software development projects are increasingly geographical distributed with offshoring, which introduce complex risks that can lead to project failure. Co-sourcing is a highly integrative and cohesive approach, seen successful, to software development offshoring. However, research of how co-sourcing shapes the perception and alleviation of common offshoring risks is limited. We present a case study of how a certified CMMI-level 5 Danish software supplier approaches these risks in offshore co-sourcing. The paper explains how common offshoring risks are perceived and alleviated when adopting the co-sourcing strategy in a mature (CMMI level 5) software development organization. We found that most of the common offshoring risks were perceived and alleviated in accordance with previous research, with the exception of the task distribution risk area. In this case, high task uncertainty, equivocality, and coupling across sites was perceived more as risk alleviation than risk taking. This perception of task distribution was combined with high attention to the closely interrelated structure and technology components in terms of CMMI and the actors’ cohesion and integration in terms of Scrum.

Keywords

Risk Management, Co-sourcing, Distributed Software Projects, CMMI, Scrum, Agile

INTRODUCTION

Global competition, need for flexibility and resources with new types of expertise as well as reduction of costs drives software developing companies to engage in geographical distributed software projects (Lacity et al. 2009; Persson et al. 2009). Companies may pursue these opportunities by engaging in co-sourcing, where an outsourcing provider and a client meld their IT competencies to accomplish the clients work (Kaiser and Hawk 2004). Nevertheless, as in other business engagements there are risks associated with this practice. While software companies can improve their business processes by the use of CMMI (capability maturity model, integrated) and agile methods, the risks related to offshoring may still be present. Therefore, more studies on the dynamic interactions between out-sourcing and firm capabilities and emergent models of IT outsourcing are needed for understanding how to manage risks across a portfolio of contracts and suppliers (Lacity et al. 2010). Earlier studies have identified risk factors and alleviation methods as such in traditional supplier-client offshoring relationships (Iacovou and Nakatsu 2008; Lamersdorf et al. 2012; Persson and Mathiassen 2010; Singh and Nigam 2012). However, available research is limited on how co-sourcing shapes the perception of offshoring risks where intermediate organizations are involved. More specifically, there is limited research of how an agile method such a Scrum combined with the highest CMMI level influences the perception and mitigation of risks in software development offshoring. Leading us to the following research question:

How are common offshoring risks perceived and alleviated when adopting the co-sourcing strategy in a mature (CMMI level 5) software development organization?

This paper presents how Systematic, a CMMI level 5 company applying Scrum, perceives and alleviates risks in a co-sourcing environment across two countries involving an intermediate organization. First, the paper introduces the theoretical background on co-sourcing and risk alleviation in offshoring. The research approach section describes the case and how we collected and analyzed data. The findings section presents our analysis of the company’s alleviation of risk and we identify three risk areas where the case company pursues a high level of risk contrary to
the suggestions from the literature. Hereafter we discuss how our analysis address the research question and contributes to previous research. Finally, we summarize the conclusion of the paper.

THEORETICAL BACKGROUND

Offshore outsourcing involves cross-organizational transactions by the use of external agents to perform one or more organizational activities (Dibbern et al. 2004). In software development, this can apply to anything from the use of contract programmers to third-party facilities management. A software risk denotes an aspect of a development task, process, or environment, which, if ignored, increases the likelihood of project failure (Lyytinen et al. 1998). Both domestic and offshore outsourcing in software development involves risks (Nakatsu and Iacovou 2009) and numerous research efforts have investigated risks particular to offshoring and distribution (Iacovou and Nakatsu 2008; Lamersdorf et al. 2012; Persson and Mathiassen 2010; Singh and Nigam 2012). However, the pursued ideal of an effective offshoring setup may differ significantly among software companies, influencing the perception and prioritization of offshoring risks. While risk frameworks have different organizational attention shaping in terms of Leavitt’s (1964) socio-technical model involving structure, actors, technology, and task (Lyytinen et al. 1998), the pursued organizational setup in offshoring may shape the perception of the associated risks and their alleviation. The four components of structure, actors, technology, and task are strongly related such that changes in one component will have planned or unplanned effects on the others (Lyytinen et al. 1998). Thus, more research is needed of how risk attention is shaped by different strategies for the setup of software development offshoring.

Offshoring setups may pursue high levels of cohesion, interdependency, and integration, while other setups pursue high levels of independence and low coupling among sites. In the pursuit of high cohesion, companies may co-locate the software developers (Persson 2013; Šmite et al. 2010) adopt agile methodologies (Jalali and Wohlin 2012; Persson et al. 2012) and strive for virtual team setups with high levels of trust (Siebrad et al. 2009; Söderberg et al. 2013). In addition to the widespread virtual team conceptualization (Curseu et al. 2008; Ebrahim et al. 2009; Martins et al. 2004; Powell et al. 2004; Schiller and Mandviwalla 2007), the high cohesion approach in software development offshoring has been conceptualized as co-sourcing (Kaiser and Hawk 2004). Kaiser and Hawk (2004) define co-sourcing as an outsourcer and client melding their IT competencies to accomplish the client’s work. Based on a case study from the financial industry Kaiser and Hawk (2004) suggest five steps involving engagement, commitment, interchange, co-sourcing, and alignment. The goal of alignment in outsourcing means alignment between the two firms in commitment and values through mutually orientated adaptation of strategy and organization (Kaiser and Hawk 2004). However, such mutual adaption may appear less feasible in offshoring setups involving intermediating organizations or departments, as in the Irish bridge involving two-stage offshoring (Olsson et al. 2008) or offshore middlemen (Mahnke et al. 2008). In this way, available research provides limited explanation of how a high cohesion strategy such as co-sourcing shapes the perception of offshoring risks with intermediating organizations.

The processes of software development have different conceptualizations of the ideal practice at the operational level. One of these is the CMMI for development (CMMI Product Team 2006), which prescribes 5 levels of maturity ranging from initial, managed, defined, and quantitatively managed, to optimizing at level 5. Elevating the CMMI certification at the client organization has been suggested as a best practice in offshoring (Rottman and Lacity 2006). Specifically to close the process gap between client and supplier organizations (Rottman and Lacity 2006) since at one point in time more than half of the firms worldwide that were certified at level 5 were in India (Matloff 2005). However, CMMI has been criticized in relation to offshoring since a level 5 supplier certification provides no guarantee of successful outcomes (Matloff 2005). Interestingly, CMMI has been combined with agile methods even though the two approaches may be contradictory in some aspects (Persson 2010; Santana et al. 2009; Turner and Jain 2002). Such successful combination has been shown in a case with CMMI level 5 and Scrum (Sutherland et al. 2008a). The adoption of agile methods in offshoring has several accounts of success (Persson et al. 2012; Sutherland et al. 2008b) and reflects a high cohesion approach to offshoring. However, the presented research provides limited explanation of how an agile method such as Scrum combined with a high CMMI level shape the perception of offshoring risks.

Software development risks can be managed by numerous approaches, e.g. the eight presented by Keshlaf and Riddle (2011) in their developing effort of a 9th approach for distributed settings. In fact, numerous research efforts has proposed risk frameworks for offshoring and distribution of software development (Iacovou and Nakatsu 2008; Lamersdorf et al. 2012; Persson and Mathiassen 2010; Singh and Nigam 2012). Persson et al. (2009) present a framework that systematically integrates a decade of research on global software, virtual teams, distributed projects,
and outsourcing into 8 risk areas and 35 resolution techniques with mutual links. Each of these 8 risk areas (Table 1) is an abstraction of 3 risk factors.

<table>
<thead>
<tr>
<th>Leavitt’s (1964) Socio-technical components</th>
<th>#</th>
<th>Risk Area</th>
<th>Description of high risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>1</td>
<td>Task Distribution</td>
<td>High task uncertainty, equivocality, and coupling across sites.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Knowledge Management</td>
<td>Inhibited knowledge creation, capture, and creation across sites.</td>
</tr>
<tr>
<td>Structure</td>
<td>3</td>
<td>Geographical Distribution</td>
<td>High spatial, temporal and goal distribution among sites.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Collaboration Structure</td>
<td>Breakdowns in collaboration, coordination and process alignment across sites.</td>
</tr>
<tr>
<td>Actors</td>
<td>5</td>
<td>Cultural Distribution</td>
<td>Dividing language barriers, work culture and cultural bias across sites.</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Stakeholder Relations</td>
<td>Low stakeholder commitment, mutual trust and relationship building across sites.</td>
</tr>
<tr>
<td>Technology</td>
<td>7</td>
<td>Communication Infrastructure</td>
<td>Limited personal communication, media support, and teleconference management.</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Technology Setup</td>
<td>Poor network capability, tool capability and configuration management.</td>
</tr>
</tbody>
</table>

Table 1 Risk areas for distribution of software development (Persson et al. 2009)

The integrative risk management framework for distributed software development (Persson et al. 2009) explicitly address the four components of Leavitt’s (1964) socio-technical model and it is arguably compatible with CMMI (Persson and Mathiassen 2010). However, the framework provides no explanation of how these risk areas (Table 1) are perceived and alleviated when adopting the co-sourcing strategy in a mature (CMMI level 5) software development organization. Thus, in the following, we present our research approach for investigating a CMMI level 5 certified software company employing agile methods and co-sourcing in offshoring with the risk management framework as the analytical lens.

**RESEARCH APPROACH**

This section presents the case and its related context followed by an explanation of how we collected and analyzed data. The case study approach was in the terms of Cavaye (1996) single case with interpretive use of qualitative data for discovery. This interpretive research approach allowed us to investigate how co-sourcing shape offshoring risks in its organizational and cross-cultural context as socially constructed and thus open to several interpretations by organizational actors but also to us as researchers (Klein and Myers 1999; Walsham 1995; Walsham 2006).

**The case**

The software company *Systematic* was established in 1985 and have more than 450 employees at offices in Denmark, the United Kingdom, the United States, Australia, Germany, Finland and Sweden. They have four core business areas. *Systematic* is the largest privately owned Danish software development company and is one of few European companies that has reached and sustained a CMMI level 5 certification since 2005 (Pries-Heje et al. 2008). Especially larger customers often requires a high maturity level. Their later addition of the agile method Scrum in 2006 supposedly enhanced the productivity with a factor two (Sutherland et al. 2008a), even though some research has claimed that CMMI can be in conflict with agile methods as Scrum (Santana et al. 2009; Turner and Jain 2002). Scrum is an iterative and incremental development model where planning is concurrent to the development activities and the work is divided into smaller chunks (often weekly) called sprints. Each sprint is planned to be self-contained leading to a new running version on the road to the final software product (Jakobsen and Sutherland 2009). *Systematic* has outsourced system development activities offshore for some years, primarily with a cost-reduction focus, with varying degree of success. In 2010, *Systematic* initiated cooperation with the offshoring intermediary company *Conscensia* and in autumn 2012, they bought 25 % of the company. *Conscensia* is a Danish company established in 2006 selling facilitation of software development offshoring to Ukraine (cities of Lviv and Kiev).
The case study takes it’s offset in one of the divisions of Systematic following the development of ‘one of the main product lines. Software development is done by more than 100 developers in seven groups all divided into one or more teams where each team is staffed by both Danish and Ukrainian developers. We have focused on two teams: Team F (20 persons, 7 in Ukraine) and Team H(35 persons, 10 in Ukraine). The Ukrainian software developers are residing in facilities belonging to the Danish service provider Conscensia. Conscensia, established in 2006, provides offices including infrastructure, finding and recruitment of competences matching clients’ needs in relation to both technical and interpersonal skills as well as other human resource services of software developers in form of local facilitation of the developers (e.g. coaching, cultural training, career advisory and assistance with communication between the teams across countries). At the location in Lviv, Conscensia is organized with two delivery managers (A and B) with reference to the Vice President (VP) of Global Delivery and a Chief Operating Officer (COO) with reference to the CEO. The CEO and the VP are situated in Denmark. A local IT department manager, a Recruitment Manager and a Career Advisor, supports the COO. In all, more than 100 developers are situated in the Lviv premises.

The two Systematic teams, supported by Delivery Manager A, develop mission critical software, primarily based on .Net and Java. Both teams apply Scrum in their development process and they sit in their own open offices at each location. The teams use IntelliJ/IDEA as Integrated Development Environment, Rational Team Concert (RTC) to manage source code, and Concurrent Version System (CVS) to manage documentation. Lync facilitates the majority of communication, such as live calls and shared screens. Daily scrum meetings are held for 15 minutes in the morning in dedicated rooms using large screens and laptops showing each other’s environments. The teams are organized with a product-manager and headed by a project manager and a one or more scrum-master for each sub-team.

Data collection

The data collection included document studies and individual semi-structured interviews with team members and management from both Systematic and Conscensia. We initiated the case study with informal meetings with managers in Systematic (in Denmark) and Conscensia (in Ukraine) in spring 2012. To get an overview of the overall organization, we did exploratory interviews with managers and developers in the early summer 2012 in Lviv. We developed an interview guide based on this explorative phase focused on their offshoring challenges and alleviation strategies. This guide supported our semi-structured interviews in Lviv and Aarhus autumn 2012 and spring 2013. The pilot interviews conducted with managers of Conscensia and a couple of software developers brought about several changes to the interview guide such as framing and focusing questions for software professionals. They furthermore provided an understanding of the environment and the challenges faced by the organizations and helped identify additional candidates for interviewing.

We interviewed four members off each team with different roles and nationalities as well as managers from Conscensia and Systematic. After interviewing the Danish side of the case, we interviewed the Ukrainian side once more to qualify observations and challenge provisional findings. Each interview lasted from 40 to 60 minutes, was recorded, and fully transcribed verbatim. To ensure correct information regarding e.g. use of technology and to maintain good relations with the interviewees the transcriptions was sent for verification. In all, we did 19 interviews combined with informal meetings. In addition to the interviews, we took pictures of the premises (offices and facilities for scrum-meetings) and collected supporting documents such as organograms, sketches of workplaces, presentations, and product descriptions.

Data analysis

We analyzed the interview transcripts and documents to uncover the involved participants’ attention to or alleviation of risks related to offshoring. Searching for deviations from established theory by approaching the analysis as a critical dialogue between the theoretical frameworks presented in the background section and our empirical work (Alvesson and Kärreman 2007). To identify incidents, alleviation, or perceptions related to offshoring risks, we searched and coded the transcripts in NVivo (Bazeley 2007). We coded statements pertaining to offshoring risks and grouped them to reveal patterns or other findings. For further triangulation, managers in Systematic and Conscensia reviewed the analyses, which lead to a few corrections providing alternative interpretations and questioning of findings (Klein and Myers 1999). In the following, we present our findings related to the eight risk areas (table 1) for software development offshoring (Persson et al. 2009).
FINDINGS

This section presents the case company’s perception and alleviation of risks. For each of the eight risk areas in table 1, we identified the level of risk and the associated perspective in the case (table 2), primarily from our interviews with management. Furthermore, we identified the main two risk alleviation initiatives for each risk area in table 2.

<table>
<thead>
<tr>
<th>#</th>
<th>Risk area</th>
<th>Risk attention</th>
<th>Risk alleviation</th>
</tr>
</thead>
</table>
| 1 | Task Distribution | Pursue high risk by: High task equivocality with very limited specification that gives high task uncertainty at the vendor site combined with high coupling requiring extensive cross-site coordination | - Daily scrum meetings based on video-conferencing and extensive code-reviewing  
- Well defined processes and division of responsibilities (CMMI-5 certified) |
| 2 | Knowledge Management | Pursue medium risk by: Knowledge creation and capture is mainly at the client side with only some knowledge integration across sites | - Partial ownership of intermediary company and focus on staff retention  
- Training by client domain experts done at vendor side |
| 3 | Geographical Distribution | Pursue low risk by: Low distribution with limited time zone differences across only two yet distanced sites that share major goals | - Developers in Ukraine participate in only one team  
- Site selection and recruitment that lower distribution |
| 4 | Collaboration Structure | Pursue low risk by: Recruiting collaborative team members and establishing shared coordination mechanisms and processes across sites | - Danish team lead take part in recruitment  
- Support by delivery manager from intermediary company and CMMI-5 imposed structure |
| 5 | Cultural Distribution | Pursue low risk by: Harmonizing the work culture with English as a shared language and low cultural bias to address the fundamental differences across sites | - Screening and training in cultural awareness of staff by intermediary company  
- Teambuilding during frequent visits across sites |
| 6 | Stakeholder Relations | Pursue low/medium risk by: Recruiting committed participants with a team and client oriented identity that trust the organization and manage integration of new members, while still maintaining some differences between sites | - Surveys twice a week of employee satisfaction and opinions with local budgets for team socialization activities  
- Sharing of customer and product stories |
| 7 | Communication Infrastructure | Pursue low risk by: Strong communication support in terms of interaction media and teleconference management but only some social and personal communication | - Standard use of set up across all teams with optional collaborative tools  
- Video-conferencing of daily Scrum and urge for additional contact across sites |
| 8 | Technology Setup | Pursue low risk by: Setting up reliable network capabilities, compatible tools, and configuration management | - Well defined and stable local infrastructure provided by intermediary company  
- Software Development tools (IDE, RTC, and CVS) highly integrated into CMMI-5 processes across sites. |

Table 2: Risk attention and alleviation in Systematic’s Co-sourcing activities
Three risk areas (1, 2, and 6 in table 2) call for special interest since the case company pursues a high level of risk, yet they do not perceive it as such, contrary to the suggestions from the literature (Persson et al. 2009). An analysis related to these three areas is presented below in more detail. This analysis was guided by the department manager’s emphasis on access to more flexible and lower cost resources than found domestically as a main driver for co-sourcing in Systematic.

**Task Distribution**

*Systematic* pursues a high-risk strategy for this risk area according to the research (Persson et al. 2009) by intentionally providing very limited specification of development tasks for the Ukrainian site. Requiring extensive cross-site coordination for carrying out development tasks, which according to Persson et al. (2009) should be avoided. The sourcing manager at *Systematic* states: ‘When talking about outsourcing you tend to forget what the task is about. It is about a team that produces software together. Then they may sit in different places and talk different languages, but that does not change the basic task of collaborating on making software. We would never write a large requirements specification and through it after someone internally. We would never ask a customer for a large requirements specification and then to stay away. Why should you do that just because it’s outsourcing’. In *Systematic* the managers argue that limited specification of the tasks to be done by Ukrainian developers are beneficial for the process, since it promote local understanding and engagement forcing cross-site team integration by dialogue: ‘...their contribution is simply larger...’ (*Systematic* department manager) and ‘...there shall not be more Ukrainians than Danes, all should be integrated into teams, able to fulfill all tasks’ (*Systematic* project manager). It seems that this attitude towards task distribution has led to more engaged Ukrainian software developers leading to a more productive environment: ‘...they appreciate getting more responsibilities...and I believe that in the future they will be more engaged in training new colleagues’ (*Systematic* project manager).

Daily scrum meetings by video-conferencing appear to support the management of above mentioned high task uncertainty. This combined with well-defined processes and division of responsibilities as imposed by their CMMI level 5 structure. Thus, they manage the high task uncertainty and equivocality by establishing a high certainty for the process of working with these tasks. Furthermore, they cope with high task coupling by establishing high coordination and collaboration capabilities as reflected in the pursuit of low risk for the risk areas related to this (see risk area 4 and 6 in table 2) and supported by the *Conscensia* Delivery Manager A, who constantly monitors and coaches the working processes.

**Knowledge Management**

*Systematic* pursues a medium risk by mainly creating and capturing knowledge at the client side opposed to all sites contributing more equally as suggested by Persson et al. (2009). The sourcing manager at *Systematic* states: ‘We must get our domain experts to visit Ukraine, the more the Ukrainian team members knows the better....it matters in the daily small decisions how things works in the large context’. Furthermore, a *Systematic* project manager argue the limited knowledge integration should be reduced as ‘it would be nice with more local domain knowledge...we must improve that’. The limited creation of knowledge at the Ukrainian side exposes *Systematic* for further risk exacerbation if losing Ukrainian staff due to their desire to learn more: ‘....a small issue related to their career-development, they can’t get to know everything...the best of them (can) be lost at the top....we have decided to accept that risk’ (*Systematic* department manager). Thus, the managers at *Systematic* are less coherent in their understanding of the knowledge management risk area and its need for alleviation. Also suggesting that the medium risk exposure on knowledge management is pursued less intentionally compared to task distributed.

*Systematic* approaches knowledge management risks in several ways. As stated in the interviews, domain experts are sent to Ukraine to train the local staff, but more interesting is how *Systematic* benefits from their partial ownership of the intermediary company. The ownership assures that knowledge, e.g. about processes, not will be lost and can be influenced in-directly at board level. At the same time, the intermediary company assists in staff retention by providing alternative employment and career paths for *Systematic* team members when needed. In this way, they reduce the risk of losing knowledgeable staff. Thus, they alleviate their risks in managing the creation and integration of domain knowledge by strong management and structuring of process knowledge.

**Stakeholder Relations**
Systematic pursues a low/medium risk by maintaining some differences between sites and the attitude of staff towards colleagues from the other sites, even though they are recruiting team-oriented staff. This difference is especially visible in two areas. First, is trust not directly mirrored: ‘...it seems that in Denmark trust has swift nature, where in Ukraine....trust must be earned’ (Systematic sourcing manager) and there is a factor two difference of how fast Danish and Ukrainian developers are up to speed and of the sub-teams own understanding of how effective they are. It seems that management in Systematic perceives Ukrainian developers as cheap but also slower compared to the more expensive developers placed in Denmark. Systematic is assisted by Conscensia local cultural training and mediation.

To monitor and be able to react on decreasing levels of trust and satisfaction Systems performs biweekly on-line surveys among staff (both Ukrainians and Danes). One of the issues identified during these surveys and related performance talks was to remember to share customer and product stories with the Ukrainian side as done in the Danish offices. Thus, the Systematic managers alleviate stakeholder relations risks by treating Ukrainian and Danish developers as equals while still maintaining differences in trust, identity, and integration.

The analysis of interviews identified three risk areas where Systematic pursues medium or high risk that involved different risk attention and alleviation than proposed in the literature. In the following, we discuss how the Systematic co-sourcing strategy and its attention shaping of common offshoring risks contributes to previous research presented in the theoretical background section.

**DISCUSSION**

In the following, we review our analysis of the Systematic offshoring case in relation to the research gaps presented in the theoretical background section and our research question: How are common offshoring risks perceived and alleviated when adopting the co-sourcing strategy in a mature (CMMI level 5) software development organization?

The investigated case of co-sourcing show that they perceives and alleviates most risk areas in accordance with previous research of software development offshoring risks (Persson et al. 2009). Task distribution, however, is a notable exception as Systematic intentionally and successfully pursue high task uncertainty, equivocality, and coupling. Thus, Systematic perceive these risk factors of task distribution more in terms of risk alleviation rather than risk taking. The success of this strategy may be explained by their alleviation of the other risk areas related to structure, actors, and technology, allowing a different perception of task distribution. Thus, the co-sourcing strategy (Kaiser and Hawk 2004) allow high task uncertainty by a strong alignment of structure, actors, and technology (supported by a CMMI maturity level 5). This risk attention shaping analysis complement the Lytinen et al. (1998) use of Leavitt’s (1964) socio-technical model to show how risk areas may be not only be alleviated indirectly by addressing the other three components in the system model but also the perception of a risk area. Elaborating how the co-sourcing strategy for offshoring (Kaiser and Hawk 2004) shapes risk attention in socio-technical terms (Leavitt 1964; Lytinen et al. 1998). The risk attention shaping of offshoring approaches other than co-sourcing is however still an important avenue for future research.

In the theoretical background section, we argued that available research provides limited explanation of how a high cohesion strategy such as co-sourcing (Kaiser and Hawk 2004) shapes the perception of offshoring risks with intermediating organizations (Mahnke et al. 2008). In our case, the intermediary company Conscensia primarily facilitated the alleviation of risks directly related to structure and actors. They mediated independent software developers that Systematic would approve and integrate into their teams. This makes Systematic capable of using their own sophisticated structure and technology benefiting from Conscensia’s capabilities. However, as the co-sourcing approach made the two companies more entangled and mutually dependent Systematic acquired 25% ownership of Conscensia. This is a way of reducing the mutually dependency risk of co-sourcing, that has been given limited attention by Kaiser and Hawk (2004).

We argued in the theoretical background that an agile method such as Scrum (Sutherland et al. 2008a) combined with a high CMMI level (Rottman and Lacity 2006) might shape the perception of offshoring risks. The high CMMI level appeared to provide risk alleviation related to structure and technology in the risk framework (Table 2). Thus, a high CMMI certification may help offshoring risk alleviation, but not to keep up with a vendor’s high certification as claimed by Rottman and Lacity (2006). In the Systematic case, the vendor side is independent developers partly organized by an intermediary. CMMI helped in terms of a well-defined and continually improved structure for including these developers. On the other hand, the agile method Scrum appear to have shaped the perception of the
task distribution risk area by embracing uncertainty, equivocality, and coupling as unavoidable. They manage this risk area, not only by daily interaction through video-based standup meetings and frequent code-reviews (table 2), but also through successful risk alleviation of the other risk areas in conjunction with the support of CMMI.

The Persson et al. (2009) framework integrates risks related to software development offshoring in accordance with CMMI (Persson and Mathiassen 2010) and socio-technical terms (Leavitt 1964; Lyytinen et al. 1998). However, findings from the Systematic case study show the framework has limited attention to the different framing of task distribution with the introduction of agile methods. The framework does not explicate how the understanding and alleviation of one risk area may alleviate or even exacerbate other risk areas. Yet our case study show how these relations can be very important for understanding risk perception and alleviation as seen with the task distribution risk area. These findings, illustrate the importance of future research on tool and framework support for more explicit management of interrelationships between risk components (El-Masri and Rivard 2012).

CONCLUSION

This paper explained how common offshoring risks are perceived and alleviated when adopting the co-sourcing strategy in a mature (CMMI level 5) software development organization. While most of the common offshoring risks were perceived and alleviated in accordance with previous research, the perception of the task distribution risk area was different from previous research on offshoring risks. In this case study of co-sourcing in a mature (CMMI level 5) software development organization, they perceived high task uncertainty, equivocality, and coupling across sites to be risk alleviation rather than risk taking. The perception of task distribution was furthermore shaped by high attention to the closely interrelated structure, actors, and technology components. While our findings show how co-sourcing may shape the risk perception of task distribution, additional research is still needed of the management of interrelationships between different risk components, the underlying software ecology, and how trust influences the processes.

ACKNOWLEDGMENTS

The research has received support from Center for IT Project Management and Innovation at Aarhus University. We also thank Conscensia and Systematic for their helpfulness and participation. Ternopil National Economic University has provided a living discursive atmosphere and highly appreciated logistic support during stays in Ukraine. Bjarne Rerup Schlichter is professor by honor at TNEU.

REFERENCES

Conflict Resolution Satisfaction and IS Program Effectiveness: Constructive Controversy Theory

James J. Jiang  
Department of Business Administration  
College of Management  
National Taiwan University, Taiwan  
jjjiang@ntu.edu.tw

Victor Chen  
Department of International Business  
National Cheng Kung University, Taiwan  
vchen@mgt.ncku.edu.tw

Sigi Goode  
College of Business and Economics  
The Australian National University, Australia  
sigi.goode@anu.edu.au

Walter Fernandez  
Australian School of Business  
University of New South Wales, Australia  
w.fernandez@unsw.edu.au

ABSTRACT
Conflict occurrence during information systems (IS) development/implementation is an unavoidable phenomenon. Conflict strains interactions and trust, leads to further conflict, and has a negative effect on system implementation success. Therefore, resolving conflict that arises during the IS implementation process has been a crucial issue for decades. Unfortunately, the existing empirical evidences were not consistent with this conventional wisdom and lacking of a theoretical explanation. We argue that the integrative conflict management is an effective mechanism for conflict resolution under the multi-project system development context. We construct a model using controversy conflict theory in order to explain the effect of conflict resolution on IT projects implementation effectiveness. We test the model using data from 183 large-scale IT implementation projects. Our model and findings support our theory that integrative conflict management supports effective conflict resolution and program outcomes.

Keywords  
Program, conflict, resolution, constructive controversy.

INTRODUCTION
User participation and involvement is a critical activity for the information system development (ISD) success (McKeen et al., 1994; Wang et al., 2006). It also creates unavoidable conflict among stakeholders due to different backgrounds and needs (Robey et al., 1989; Hartwick and Barki, 1994; Barki and Hartwick, 2001; Liang et al., 2010, 2012) and this unresolved conflict often has negative impacts on team performance and project performance (Robey et al., 1993; Sawyer, 2001; Liu et al., 2009). As a result, successfully managing conflict during the ISD process has long been suggested as one of the most critical issues in IS project implementation (Hartwick et al., 1994; Robey et al., 1993, 1989; Robey and Farrow, 1982).

Two streams of conflict management have been pursued in past research. One stream has suggested that conflict resolution satisfaction is the key factor explaining the relationship between conflict, conflict resolution and overall project performance (Robey et al., 1989; Hartwick et al., 1994; Barki et al., 2001). A goal of conflict management is to reach satisfactory conflict resolution among stakeholders, thereby avoiding negative impacts of conflict. Empirical evidence has shown only weak support for this assertion (Barki et al., 2001); however, empirical evidence has revealed a significant relationship between the style of conflict management and project outcomes (Barki et al., 2001; Robey et al., 1993). Specifically, integrative conflict management, which attempts to maximize the aggregate outcomes of conflicting parties and to create the largest collective reward (Janssen et al., 1999; Johnson and Johnson, 1979; Lax, 1987), was the most effective conflict resolution mechanism among the other conflict management styles (e.g. compromising, asserting, avoiding, accommodating). Other IS research, however, has focused on identifying the match between the type of conflict and conflict management style (Sawyer, 2001; Kankanhalli et al., 2007; Chou and Yeh, 2007). Empirical evidence has consistently shown that an integrative conflict management style was the most effective conflict resolution mechanism for resolving task-related conflicts during ISD processes (Jiang et al., 2013; Kankanhalli et al., 2007; Chou et al., 2007).
Both streams of literature are converging to the same principal outcome: that integrative conflict management could be adopted for resolving task-related conflicts occurred during ISD processes. However, empirical studies have not confirmed whether conflict resolution satisfaction is an effective mediator between conflict and project performance. At least two shortcomings are evident in existing studies. First, studies conducted in other disciplines have suggested that cooperation and mutual coordination behaviors are the two most important concepts for explaining the relationship between satisfactory conflict resolution and final team outcomes (Simons and Peterson, 2000). However, the impact of satisfactory conflict resolution on teamwork has been overlooked in the IS conflict literature. This link may yield empirical support for the relationship between satisfactory conflict resolution, team performance and project outcomes. Second, the relationship between conflict management mechanisms and project performance has lacked a theoretical explanation. How do conflict management mechanisms directly impact project performance, in addition to the impact through satisfactory conflict resolution? Social psychology theorists have argued that constructive controversy conflict management can enhance collaboration among team members. Given the central role that conflict management researchers ascribe to interference, explaining the effect of conflict on project and project team performance without understanding the effect on teamwork processes seems to be a serious omission.

The main objective of this study is, therefore, to examine whether conflict resolution satisfaction is a significant mediator between conflict and project outcomes, by incorporating teamwork process variables as explanatory factors. Based on conflict management theory, we propose that trust, mutual support, and cooperation are the consequences of satisfactory conflict resolution (Tjosvold, 1998; Simons et al., 2000; Guerra et al., 2005) which, in turn, enhance final project outcomes (Pinto et al., 1993; Hoegl et al., 2004). Furthermore, we argue that the effect of integrative conflict management on project performance will be mediated by these teamwork process variables.

RESEARCH BACKGROUND

Conflict is the awareness of discrepancies in opinions, incompatible wishes, or irreconcilable desires among parties involved in a relationship (Boulding, 1963). Conflict has been found to be multidimensional (Jehn, 1995), and researchers distinguish between relationship conflict and task conflict. Relationship conflict refers to tension, animosity, and annoyance among members within a group. Task conflict is the disagreement among the group members’ idea and opinions about the task being performed. Relationship conflict has been found to be detrimental to group performance, while task conflict can be beneficial to team effectiveness when the conflict is satisfactorily resolved (Jehn, 1995; Simons et al., 2000).

Unresolved conflict can strain relationships and trust between parties. It could lead to the development of further conflict, and has a strong, negative effect on overall software product success and customer satisfaction (Deutsch, 1990). Therefore, satisfactory conflict resolution represents a key issue in successful management and implementation of ISD projects (Behfar et al., 2008). Satisfactory conflict resolution reflects positive residual outcomes and behaviors at the closure of a conflict episode (Barki et al., 2001; Karau and Williams, 1993). The positive relationship outcomes are often defined in terms of positive perceptions of overall satisfaction, positive perceptions of cooperation, supportive behaviors, increased decision autonomy, and higher perception of relationship quality (Janssen et al., 1999; Locke and Latham, 2002; Wall et al., 1986).

Conflict management refers to the strategies implemented by group members to reduce or solve conflict (Jehn and Mannix, 2001; Van de Vliert, 1997). Based on Pruitt and Rubin (1986)’s dual concern theory, five conflict management styles have been observed including: asserting, accommodating, compromising, avoiding, and collaborative/integrative (Barki et al., 2001; Volkema and Bergmann, 1995; Thomas and Pondy, 1977). These conflict management styles vary depending on the concern for the self and concern for others (Pruitt, 1983). Literature shows that the conflict management strategy has direct impacts on group performance: approaches based on compromise (e.g. matching others’ concessions) and integrative conflict management (e.g. high concern for self and others) have been observed as effective for resolving IS development conflict (Chou et al., 2007).

Antecedents of conflict relate to the characteristics of team members, teams, projects, and organizations (Hinds and Bailey, 2003). Robey, et al. (1989) and Barki and Hartwick (1994) highlight how individual participation and influence affecting conflicts, its resolution and project performance. Constant et al. (1994) describe how resource sharing affects the way software developers work together. More recently, Jiang et al. (2013) found that the diversity of members’ knowledge background affects the levels of conflict experienced by IS team members. Conflict
resolution management examines how conflict management strategy/behavior impacts the relationship between conflict and team performance.

The major theme in prior literature is that satisfactory conflict resolution mediates conflict and team performance (Hartwick et al., 1994; Robey et al., 1989, 1993; Barki et al., 2001) and ample research has observed a direct effect of conflict management strategy on satisfactory conflict resolution (Wall and Nolan, 1986; Rahim, 2002) as well as team performance (Chou et al., 2007). The prevailing suggested conflict resolution model is shown in Figure 1. Unfortunately, empirical support for the relationship between satisfactory conflict resolution and team performance has been weak. The focus of this study is to further examine this issue.

![Figure 1 Conflict and Conflict Management Model (Hartwick et al., 1994)](image)

To understand the theoretical underpinnings of conflict management and resolution outcomes in IS development projects, it is first important to understand the conceptualization of conflict in the ISD setting. Barki and Hartwick (2001) identified four components of conflict in the IS environment: interdependence, disagreement, interference and negative emotion. Collectively, these properties span situational (interdependence), cognitive (disagreement), behavioral (interference), and affective (negative emotion) elements of conflict situations. Interdependence is present when individuals share common resources and/or each individual's outcomes are affected by the actions of others, a common characteristic of the IT multi-project setting (Hsu et al., 2011). In the absence of interdependence, the actions of one group have no effect on the outcomes of another group. Even though teams may be in interdependent relationships with others, not all teams will face conflict (Janssen et al., 1999; Jehn et al., 2001). Hence, interdependence is an essential situational element, but not a sufficient condition for conflict to affect final outcomes.

Disagreement can occur when groups have dissimilar objectives. Even though individuals or teams may be in disagreement with others, not all will face conflict when the areas of disagreement are irrelevant, minor or unimportant. Hence, disagreement is an essential cognitive element, but not a sufficient condition for conflict to manifest. The disagreement should relate to critical tasks or goal-related issues. Interference occurs when a group interferes with or opposes another’s attainment of its outcomes or goals. Interference represents a key behavioral
element of any conflict between groups, undermining smooth resource allocation across project teams. Negative emotions emerge in conflict when there are major disagreements, or when parties interfere with the attainment of each other’s goals. Elimination of negative emotions, like jealousy, anger, anxiety, in favor of positive attitudes is an expected consequence of conflict resolution. Hence, interference and negative emotion are actually consequences of the unresolved conflicts among members or groups.

Accordingly, we expect that satisfactory conflict resolution will tangibly reduce subsequent negative attitudes and behaviors among program team members. However, the mechanism of action that translates these teamwork behaviors into satisfactory conflict resolution outcomes remains unexplained. Resolving conflict on its own does not necessarily lead to mutual support and cooperation between team members, because residual effects of the conflict itself may persist. Unfortunately, the IS literature has overlooked these potentially important teamwork process factors in seeking to explain the ISD conflict management phenomenon.

CONSTRUCTIVE CONTROVERSY THEORY

The theory of constructive controversy was proposed by Johnson and Johnson (1979) as an effective guide to resolve conflicts and to increase the quality of decision making. Constructive controversy theory was originally developed based on Deutsch’s social interdependence theory (Deutsch, 1977) and cognitive development theories (Piaget, 1976). It involves what Aristotle called ‘deliberate discourse’, wherein participants thoroughly discuss the advantages and disadvantages of proposed actions, in order to synthesize novel solutions (Johnson, 2008). Different from other conflict management methods like dominating (Rahim, 2002), distributing (Miranda and Bostrom, 1993) or avoiding (Barki et al., 2001), constructive controversy aims to provide not just a solution to the conflict, but also to reach a better conclusion that satisfies every party who joins the conflict to reach consensus with each other (Johnson, 2008; Johnson et al., 1979; Tjosvold, 2008; Johnson, Johnson, and Smith, 2000). To implement constructive controversy, people in conflict are gathered to discuss their sentiments and derive positive outcomes, using processes such as reconceptualization, perspective-taking, creative decision-making, and curiosity-building (Johnson, Johnson, and Smith, 2000; Johnson, 2008; Johnson et al., 1979; Alper et al., 1998; Deutsch et al., 2011).

Constructive controversy is often compared to concurrence seeking, debate and individualistic decisions (Deutsch et al., 2011; Johnson, Johnson, and Smith, 2000; Johnson, 2008). In concurrence seeking, group members constrain discussions to avoid argument and promote realistic appraisal of alternative ideas and courses of action. Debate occurs when two or more individuals discuss compatible positions: a judge declares a winner on the basis of who presents the best position. Individualistic decisions exist when individuals consider the issue alone while perceiving their goals to be unrelated to those of others. Jiang, et al. (2013) has shown that constructive controversy efforts made by counterparts (e.g. trying hard to seek an agreement toward those incompatibilities of opinions, ideas, theories and conclusions among each other) would positively impact coordination among project teams. In the structure of cooperation, team members believe that their goal achievements are positively correlated, only if the other members reach their goals and they can reach their own goals (Tichy et al., 2010). Individuals in cooperation recognize that they want each other to pursue their goals effectively (Alper et al., 1998). Individuals in competition believe that their goal achievements are negatively correlated: other members reaching their goals will prohibit the individual achieving their own goal (Alper et al., 1998; Tichy et al., 2010). Individuals with perceived competitive goal interdependence conclude that they are better off when others act ineffectively (Alper et al., 1998). Therefore, in the structure of cooperation, team members are more likely to work for mutual benefit than competitively, because they understand that others’ goal attainment helps them; and they can be successful together (Lewicki et al., 1998; Tichy et al., 2010).

Based on the above discussion, we propose a research model in Figure 2. Integrative conflict management should lead to positive group outcomes by maximizing the total outcomes of conflicting parties as well as to create the largest value of the collective reward at stake (Janssen et al., 1999; Johnson et al., 1979; Lax, 1987). In particular, we argue that integrative conflict management would not only an effective mechanism for conflict resolution satisfaction as suggested in the existing literature but also directly impact team members’ mutual support and cooperation behaviors. Furthermore, based on the conflict management literature, the relationship between conflict resolution satisfaction and teamwork variables (i.e., trust, mutual support, and cooperation) were examined in this study. Importantly, the trust, mutual support, and cooperation are proposed in this study as full mediators of the effects of both conflict resolution satisfaction and integrative conflict management on final project performance.
HYPOTHESES

Integrative Conflict Management and Conflict Resolution Satisfaction

Integrative conflict management techniques aim to reconcile participants’ divergent interests, where all parties are encouraged to view the conflict as an opportunity to include all parties' needs in order to deliver the best possible outcome (Kuhn and Poole, 2000; Carnevale and Isen, 1986). Under this model, parties in conflict feel that their concerns are heard and acknowledged by the other party. This approach to conflict resolution is characterized by greater levels of information sharing between parties and a drive to search for alternative outcomes that might benefit all parties (Rahim, 2002).

Because this more open approach to conflict management aims to build a consensus and compromise, parties are more likely to have positive feelings about the conflict and its resolution. If both parties feel that their concerns are being heard and recognized, they are also more likely to feel the positive affective and attitudinal outcomes that are associated with satisfaction. The focus is not combative or adversarial (Kuhn et al., 2000) and parties are more likely to feel that their concerns have been met. Integrative conflict management also builds satisfaction by using more of the conflict resolution techniques that parties want to use. Prior research has shown that employees prefer to use active listening, mentoring and coaching, role-playing and informal problem solving rather than conventional arbitration and grievance processes (Pruitt, 1983). By feeling more positive about other participants and their goals, and by using techniques that participants feel better using, integrative conflict management techniques build satisfaction with the conflict resolution process. This argument leads to our main hypothesis:

\[ H1: \text{An integrative conflict resolution strategy among IS team members will lead to satisfactory conflict resolution during IS implementation process.} \]

Integrative Conflict Management and Mutual Support

Integrative Conflict Management encourages an openness between conflicted parties (Conbere, 2001). Participants in the integrative conflict management process are encouraged to approach the resolution process with an open attitude. As part of this process, integrative conflict management works to address the sources of conflict (Lynch,
As well as the visible signs and displays of conflict (Jehn et al., 2001; Jehn, 1997). As a result, all parties are better able to identify the sources and results of the conflict. Identifying these sources helps participants in the process to avoid these problems in the future, thereby avoiding conflict activity itself.

At the individual project level, coordination is important because shared scarce resources must be available when they are required. However at the program level, mutual support supersedes coordination in order to realize the positive outcomes of productive group relationships (Parolia et al., 2011, 2013).

Because participants can better understand both the sources and the symptoms of conflict in both themselves and other participants, there is a stronger sense that conflict resolution participants can build a mutual understanding of each other’s requirements. In contrast to distributive conflict management processes that typically lead to mutual exclusion (Kuhn et al., 2000), integrative conflict management works towards a mutual support of each partner in the process (Li et al., 2012). This mutual understanding of each other’s requirements produces mutual support between participants. This argument leads to our second hypothesis:

**H2a:** An integrative conflict resolution strategy among IS team members will lead to mutual support among team members during IS implementation process.

**Integrative Conflict Management and Cooperation**

Prior research has shown that unresolved or poorly managed conflict results in poorer cooperation between parties. Conflict resolution mechanisms built on opposition or confrontation compel participants to seek for their own benefit at the expense of the other party in order to strengthen their position amid scarce resources (Kuhn et al., 2000). Competition for potentially scarce resources produces adversarial relationships between parties, leading to reduced coordination behaviors (Pesämaa et al., 2009). Because these parties do not wish to work together to resolve conflict, or if each party feels that scarce resources will be distributed unfairly among participants, they are more likely to avoid working together (Kumar and van Dissel, 1996).

Under the integrative conflict management approach, participants are encouraged to share their ideas and concerns through a variety of resolution techniques (Conbere, 2001; Lynch, 2001). By encouraging parties to share their perspectives and views, each party builds a better understanding and knowledge of the other party’s goals and requirements. The integrative conflict management approach itself hence improves cooperation between parties by improving what is known between parties, thereby building a consensus of what should be done in future (Parolia et al., 2011). Because parties are encouraged to work together to bring about useful outcomes, they are more likely to cooperate with each other in order to achieve high “joint benefit” (Carnevale et al., 1986). This argument leads to our third hypothesis:

**H2b:** An integrative conflict resolution strategy among IS team members will lead to cooperation among team members during IS implementation process.

**METHODOLOGY**

In order to understand the conflict effects among projects, we required a context in which this conflict behavior would be present. Accordingly, we chose enterprise system implementations as the context of our study: while enterprise systems are a popular information system type among firms (Ko et al., 2005; Liang et al., 2007), their implementations are characterized by high rates of failure, uncertainty and conflict (Nah et al., 2001; Al-Mashari et al., 2003; Allen, 2005). Most enterprise system implementations also depend heavily on the joint work of developmental teams (Robey et al., 2002). We hence felt that this would be a suitable context for our study.

We sought a research method that would allow us to gather a large number of variables from a range of firms at different locations (Dillman et al., 2008). These variables related to ongoing processes at the firms, and hence archival data sources were not appropriate. To satisfy these requirements, we selected a questionnaire survey to collect data. This method has been used successfully in a range of prior project management and IS research literature (Pinsonneault and Kraemer, 1993) and accordingly we felt it would be appropriate for this work.

We selected a sample of the top 1000 firms in Taiwan. We drew our sample from a list published by China Credit Information Service, Ltd. This database, which has been well used in prior literature, yielded company, ownership,
and contact details for our sample. Sample members had to have begun implementing an enterprise system development program, and had to have implemented at least two projects within the program. For each firm, we sought three key informants as respondents to the survey, furnished to us by a lead contact at the firm (such as the CIO). Firms that agreed to participate were sent three questionnaires by postal mail. We followed the advice of Dillman (2008) in designing the instrument, including testing and validating items (Helgeson et al., 2002). The survey instruments were written in Chinese and translated and checked by three experts with extensive research experience. The instrument is available from the corresponding author. Excluding incomplete responses and those that did not conform to the study requirements, we had 183 usable responses.

DATA ANALYSIS AND RESULTS

Structural equation modeling (SEM) was used to build and assess the measurement and structural models (Anderson and Gerbing, 1988). One factor testing was used to detect potential common method variance. In the unrotated factor structure, the first factor failed to comprise a majority of the variance from the constructs. The possibility for common method variance was hence deemed to be low.

A Confirmatory Factor Analysis (CFA) of the six constructs was then conducted in IBM AMOS 20.0 to test convergent and discriminant validity. Estimation of the CFA with six constructs resulted in a good fit statistic ($X^2 = 180.83; df = 104; X^2/df = 1.73; NFI = 0.935; CFI = 0.971; AGFI = 0.846; RMSEA = 0.064$). All factor loadings were above the recommended 0.5 and were significant (see Table 3).

Convergent validity was assessed by calculating composite reliability and average variance extracted (AVE). Composite reliability values exceeded the 0.7 threshold (Hair et al., 1995), which indicated adequate internal consistent among the respective constructs. AVE, the ratio of the sum of the variance captured by the construct and measurement variance, was applied as the measure for convergent validity (Bagozzi and Yi, 1988). Composite reliability values exceeded the recommended 0.7 threshold, and the AVE for each construct exceeded the recommended 0.5 level. These findings indicated good convergent validity.

<table>
<thead>
<tr>
<th>Table 3 Constructs, Items and Item Loadings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Construct</td>
</tr>
<tr>
<td>Conflict Resolution Satisfaction</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Mutual Support</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Cooperation</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Program Effectiveness</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Trust</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Integrative Conflict Management</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
This study used the estimated correlation between all constructs pairs and the shared AVE squared root of each construct as criteria to demonstrate discriminant validity. The AVE square root of each construct should be greater than its correlation with the other constructs (Fornell and Larcker, 1981). The results of this testing, provided in Table 2, show that the AVE square root of each construct is greater than its correlation with the other constructs. This evidence suggests good discriminant validity.

Table 4 Square Root Average Variance Extracted

<table>
<thead>
<tr>
<th>Construct</th>
<th>Conflict Resolution Satisfaction</th>
<th>Mutual Support</th>
<th>Cooperation</th>
<th>Program Effectiveness</th>
<th>Trust</th>
<th>Conflict Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conflict Resolution Satisfaction</td>
<td><strong>0.900</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mutual Support</td>
<td>0.219</td>
<td><strong>0.846</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooperation</td>
<td>0.285</td>
<td>0.408</td>
<td><strong>0.930</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program Effectiveness</td>
<td>0.586</td>
<td>0.359</td>
<td>0.340</td>
<td><strong>0.933</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trust</td>
<td>0.188</td>
<td>0.355</td>
<td>0.820</td>
<td>0.247</td>
<td><strong>0.903</strong></td>
<td></td>
</tr>
<tr>
<td>Conflict Management</td>
<td>0.665</td>
<td>0.293</td>
<td>0.293</td>
<td>0.537</td>
<td>0.171</td>
<td><strong>0.788</strong></td>
</tr>
</tbody>
</table>

Boldface numbers on the leading diagonal are the square root of the variance shared between constructs. Off diagonal elements are the correlations among constructs.

SEM analysis with maximum likelihood estimation was used to test the hypothesized paths. Estimation of the structural model with six constructs resulted in good model fit statistics ($X^2 = 242.41; df = 111; X^2/df = 2.184; NFI = 0.913; CFI = 0.950; AGFI = 0.820; RMSEA = 0.081$). The significance of the structural coefficients, depicted in Figure 2, provides additional evidence in support of the research model. Estimated $R^2$ for the five dependent variables were, 45% for Conflict Resolution Satisfaction, 37% for Trust, 19% for Mutual Support, 70% for Cooperation and 17% for Program Effectiveness. Hypothesis 1, that Integrative Conflict Management positively affected Conflict Resolution Satisfaction, was accepted ($\beta = 0.67, p < 0.001$). Hypothesis 2a, that Integrative Conflict Management positively affected Mutual Support, was accepted ($\beta = 0.26, p < 0.1$). Hypothesis 2b that Integrative Conflict Management positively affected Cooperation, was not accepted ($\beta = 0.14, NS$).
DISCUSSION AND CONCLUSIONS

Prior IS literature has observed that a ‘problem-solving’ conflict management style is an effective mechanism to tackle ISD intra-project team conflict (Barki et al., 2001). It proposed that conflict resolution satisfaction fully mediates the relationship between conflict and final implementation outcomes. Unfortunately, this theoretical argument has not been validated in extant empirical evidence (Barki et al., 2001). Based on controversy conflict theory (Deutsch et al., 2011; Johnson, 2008), Jiang et al. (2013) reaffirmed that integrative conflict management, a conflict resolution method focusing on collaboration in consideration of all parties, is effective in resolving conflict in the multi-project context. Controversy conflict theory begins with the existence of a strong goal among members (Johnson, Johnson, and Tjosvold, 2000). Jiang, et al. (2013) suggested that integrative conflict management will lead to a consensus decision on delivery and the required level of commitment toward completing the mutually-shared, overall multi-project goals. This prior work pointed to the importance of integrative conflict management for resolving conflict under the multi-project context: our aim in this paper was to test these relationships empirically. Using controversy conflict theory as a basis, we modeled the effect of conflict resolution satisfaction on project outcomes, through trust, mutual support and cooperation. We tested this model on 183 large enterprise implementation projects by way of a survey.

Our results supported our main theory that integrative conflict management led to conflict resolution satisfaction. Integrative conflict management was also significantly related to mutual support among teams. While our initial results did not confirm a positive relationship between integrative conflict management and cooperation, additional post-hoc testing revealed a positive relationship once non-significant paths had been removed from the model.

Practically, our results suggest that involving conflicted participants ought to lead to better outcomes than merely resolving the incident of conflict itself. While prior work has emphasized that resolving conflict is important, our results now show that an approach based on including and involving participants in the resolution process will lead to better mutual support and cooperation. Our findings emphasize that an integrated approach will be beneficial in the multi-project environment, where separate teams must work together in the face of limited resource availability.

Our study may be open to two limitations. First, our study explored implementation of large-scale enterprise systems. We chose this research context in order to understand the conflicts that often characterize such

![Figure 3 Structural Model of Integrative Conflict Management](image-url)

* * * \( p < 0.05; ** p < 0.01; *** p < 0.001 \)
implementation projects. However, implementation projects for other system types may exhibit different conflict behaviors and outcomes. Second, our study was conducted on large firms in Taiwan. Companies of other sizes or located in other regions may feature different project types and practices.

Several areas for further research extend from this study. First, better conflict management ought to result in fewer conflict episodes and superior resolution. Because conflict can produce negative effects on teams and their interrelationships, it would be useful to see whether firms employing integrated conflict management tools are able to undertake more projects than firms that do not, or whether they are able to employ fewer resources to match the same level of overall project output. Benefits of this nature ought to be visible at the portfolio investment level. Second, because many firms employ third party consultants in their enterprise system implementations, it would be interesting to see whether these integrative conflict management tools are also effective in bridging the conflict divide with these external parties and whether their use in that context also translates to the positive project outcomes observed in this study. The effectiveness of these techniques on virtual teams, characterized by significant geographical distance and thinner social interrelationships, also warrants further analysis. Third, our model improves understanding of how conflict can be managed in the multi-project setting. Our theory basis of controversy conflict theory illustrates that some conflict types can be useful in these environments. It would be useful to develop an understanding of the signs and symptoms of this ‘beneficial’ conflict so that it can be more effectively marshaled in the firm, and distinguished from more destructive and unproductive conflict types.

REFERENCES


Delivering IT PMO Value: Understanding Stakeholder Perceptions & Expectations

Siddhartha Arumugam
Swinburne University of Technology
sarumugam@swin.edu.au

Judy McKay
Swinburne University of Technology
jmckay@swin.edu.au

Nicholas Grainger
Swinburne University of Technology
ngrainger@swin.edu.au

ABSTRACT

IT Project Management Offices (IT PMOs) are important to the IT project management landscape. Despite being set up to ensure IT project success, many IT PMOs struggle to survive, partly as a result of tensions and challenges. One prevailing tension that IT PMO managers face is the struggle to justify their IT PMO value. Prior research has uncovered a key factor behind this tension – the fact that these IT PMOs play multiple roles and have to meet the competing demands and expectations of stakeholders. This study examines the functions of the IT PMO and its core values with a visual framework and demonstrates its effectiveness in helping the IT PMO team understand its stakeholders’ perceptions and expectations. With this shared understanding, the IT PMO team is empowered to develop strategies to better service stakeholders, and therefore be perceived as delivering value.

Keywords

Project Management, IT Project Management Office, Stakeholder Perceptions and Expectations, IT PMO Value.

INTRODUCTION

Organisations are under constant pressure to innovate, especially so in today’s volatile landscape which is argued to have driven the need for IT-enabled business transformation projects (Sauer, Gemino and Reich, 2007). Because these projects have grown in “strategic and operational importance”, much is expected of them (Sauer and Reich, 2009:185). Moreover, many organisations must now simultaneously juggle the management of multiple, interrelated projects at the same time. These IT projects must therefore be managed effectively in order to have a better chance of success (Aubry, Hobbs, Thuillier, 2009; Dai and Wells, 2004; Singh, Kail and Kasi, 2009). Hence the extensive interest in the potential of the IT Project Management Office (IT PMO) (Computer Economics, 2011). By providing a focused environment for the formal training and development of organisation-wide project capabilities such as project management methodologies and processes, project governance processes, project quality assurance, training, project knowledge management, and the like (Andersen, Henriksen and Aarseth, 2007), the IT PMO offers the potential to achieve good organisational outcomes and ensure the delivery of business value from these substantial investments.

Unfortunately, a majority of IT PMOs do not survive beyond two years (Hobbs and Aubry, 2007), likely due to the fact that the IT PMO concept is itself beset with tensions and challenges (McKay, Marshall, Arumugam and Grainger, 2013). One prevalent tension that IT PMO managers face is the value of their IT PMO is often questioned (ESI International, 2013; Hobbs and Aubry, 2007; McKay et al., 2013) despite the IT PMO being established with the aim of ensuring the success of IT projects. Hence the considerable interest amongst academic researchers investigating the value of the IT PMO (Aubry et al., 2009; Hobbs and Aubry, 2010; Pellegrinelli and Garagna, 2009; Thomas and Mullaly, 2007). In their book, Hobbs and Aubry (2010) suggested that the IT PMO performs multiple roles and is closely linked with multiple entities across the organisation. This implies that the perceived value of the IT PMO team is dependent on its stakeholders with potentially diverse values and preferences, and representing the various entities within the organisation.

1 When we are specifically referring to IT PMOs, we specify ‘IT PMO’. Where we are broadly describing aspects of PMOs in general, we use the term ‘PMO’.
We believe that this is the key to understanding the tension that IT PMO leaders face in justifying the value of the IT PMO. Despite acknowledging the importance of these stakeholders’ expectations, there is little empirical research on what is believed to be the key to perceived value of the IT PMO: the stakeholders’ expectations and perceptions of the core values and functions of the IT PMO. This is further supported by the fact that practitioners also believe that the key for IT PMOs in creating and proving their business value is by aligning with the organisation’s core values and to be perceived [by key stakeholders] as effectively executing its core functions (Hayes, 2011). This study examines the functions of the IT PMO (Hobbs and Aubry, 2007; Hobbs and Aubry, 2010) and its core values (Quinn and Rohrbaugh, 1981) using the concepts of perceptions, expectations, and satisfaction (Boulding, Kalra, Staelin and Zeithaml, 1993; Kettinger and Lee, 1997; Parasuraman, Zeithaml and Berry, 1985; Spreng and Mackoy, 1996).

Hence, the purpose of this study is to aid members of the IT PMO team understand what their stakeholders expect of them by using a visual framework to illustrate these stakeholders’ perceptions and expectations of the IT PMO. With this shared understanding, the IT PMO team would be empowered to develop strategies to better service their key stakeholders, and therefore be perceived as delivering value. This paper is structured as follows: in the following section, we provide a review of concept of the IT PMO and the study of perceptions and expectations in the academic literature. This is followed by a description of our research methodology and design. We then conclude with a discussion about the results of our findings.

LITERATURE REVIEW

IT PMO Functions, Values and Tensions

The positive growth of IT PMOs in organisations worldwide (Computer Economics, 2011) could be argued to result from the need for better management of IT projects (Aubry, Muller, Hobbs and Blomquist, 2010; Dai and Wells, 2004; Desouza and Evaristo, 2006). This is especially so with the increasing numbers and complexity of such projects, programs and portfolios, as well as the unacceptably high rates of project failures (Singh et al., 2009). Despite the fact that they are set up to ensure the success of IT projects, it is ironic that the future of IT PMOs seems to be somewhat uncertain. Many IT PMOs do not survive beyond two years (Hobbs and Aubry, 2007), possibly as a result of the tensions and challenges that plague these IT PMOs (McKay et al., 2013). For example, some IT PMO leaders face the tension of whether project ownership and responsibilities should be with the IT PMO or remain with business unit managers (Pellegrinelli and Garagna, 2009). For leaders of newly-set up IT PMOs, budget and resources may be channelled away from business units to fund these IT PMOs, potentially resulting in tensions created by this power shift (McKay et al., 2013). Other IT PMO leaders experience tensions which are conflicting, such as the choice they have to make between emphasising standardisation or being flexible and responsive to business demands (Hurt and Thomas, 2009; Pellegrinelli and Garagna, 2009); or between centralising or decentralising management capabilities throughout the organisation (Curlee, 2008).

One very pressing issue that IT PMO leaders face is their struggle to deliver value to their organisations (Hobbs and Aubry, 2007; McKay et al., 2013). Some IT PMOs are being valued by their organisations, while others struggle to demonstrate value, with yet others being disbanded altogether (Hobbs and Aubry, 2007). This is a fact confirmed with recent global surveys on PMOs reporting that more than 50 percent of the total number of respondents claiming that the value of their PMO has been questioned (ESI International 2011, 2012, 2013). A recent empirical study also revealed that although IT PMO teams may be performing the exact PMO functions they were setup to do, the leaders [of these IT PMOs] still struggle to justify their value to the rest of the business (McKay et al., 2013). While the value of the IT PMO to individual projects can be measured with traditional project management metrics such as ‘on time’ and ‘within budget’ for example, it is more problematic to determine the actual value delivered by the IT PMO, as aggregating value delivered from individual projects may not necessarily be indicative of the overall value delivered.

A PMO is defined as “an organisational body or entity assigned various responsibilities related to the centralised and coordinated management of those projects under its domain. The responsibilities of the PMO can range from providing project management support functions to actually being responsible for the direct management of a project” (PMI, 2008:89). Using this as a broad guideline, Hobbs and Aubry (2007, 2010) conducted an empirical study and identified 27 different roles and functions of the IT PMO, categorised into five main groups: (1) monitoring, controlling, and reporting project performance; (2) developing project management competencies and methodologies, and promoting project management; (3) multi-project management, including prioritising,
coordinating and resource allocation; (4) strategic management and planning; and (5) organisational learning, including post-implementation reviews, project audits, and managing lessons-learned databases. While these 27 functions are considered to be important to most IT PMOs, they need not all be adopted by every IT PMO (Hobbs and Aubry, 2010).

Cameron and Quinn (2011) argue that the core values (or guiding principles) of an organisation [in this case the IT PMO] is the key to its sustainability. The competing values framework (Quinn and Rohrbaugh, 1981, 1983) offers a representation of the [competing] values they have aggregated from the organisational literature: (1) human relations (HR), concerning employee well-being; (2) open systems (OS), being flexible and adapting to the changing environment; (3) rational goal (RG), emphasising productivity and efficiency; and (4) internal process (IP), focusing on organisation and structure. This research uses the competing values framework in the study because it offers the ability to chart and compare the conflicting perceptions and expectations of the IT PMO. In addition, this framework has also been applied in the study of paradoxes and pluralism (Aubry et al., 2011; Cameron, 1986; Quinn, 1988), making it fitting for this study.

Practitioners believe that the alignment of the IT PMO with the organisation, and the perceived effective execution of its roles and functions are crucial to the IT PMO being perceived as creating and delivering value to its organisation (Hayes, 2011). Hence, we explore issues around alignment as well as the values and functions of the IT PMO in this study to understand the perceived value delivery of the IT PMO.

**Stakeholder Perceptions and Expectations**

Most IT PMOs have a diverse range of stakeholders, defined here as “individuals or groups who will be impacted by, or can influence the success or failure of an organisation’s activities” (Walker, Bourne and Rowlinson, 2008:73). One contributing factor to the difficulty that IT PMO leaders face in demonstrating the business value derived from the IT PMO stems from the diversity of perceptions and expectations of the various stakeholders in the organisation have in terms of the core values and roles and functions of the IT PMO. The perceptions and expectations of the IT PMO team members might not necessarily be aligned with these stakeholders (Hobbs and Aubry, 2010). Since the role of the IT PMO requires the IT PMO team to interact with its stakeholders within the organisation, it is therefore essential for the IT PMO team to effectively manage its relationships with these stakeholders. While these stakeholders could have a direct and significant influence over the organisation’s projects and the IT PMO, they may have different cultures and values to that of the IT PMO itself (Hobbs and Aubry, 2010). Therefore, delivering services to the satisfaction of a broad range of stakeholders with differing needs and expectations is indeed challenging for the IT PMO team.

In the academic literature, the concept of satisfaction is seen to result from the comparison between the expectations of the recipients of a service and their perceptions of the actual service delivered (Oliver, 1980; Parasuraman et al., 1985; Pitt, Watson and Kavan, 1997; Spreng and Mackoy, 1996; Tesch, Miller, Jiang and Klein, 2005). In this context, ‘expectations’ convey “the desires and wants” of the recipients of a service (Parasuraman, Zeithaml and Berry, 1988:17), while ‘perceptions’ refers to what recipients perceive about the service actually delivered. Satisfaction essentially is the “discrepancy between what the individual expects and what he/she perceives he/she is getting” (Tesch et al., 2005:346), where the key factor of satisfaction is “meeting customers’ desires” (Spreng and Mackoy, 1996:211). In other words, if the stakeholder’s perceptions of service delivery of the IT PMO is congruent with his or her expectations, the IT PMO may then be perceived [by that stakeholder] as satisfying his/her requirements, and thus as delivering value. Hence, in order to be perceived as delivering value, the IT PMO team needs to first develop a shared understanding of the perceptions and expectations of the stakeholder. Shared understanding refers to mutual knowledge, mutual beliefs, and mutual assumptions (Clark and Brennan, 1991; Mulder, Swaak and Kessels, 2002) that exist between more than one party. Only then would the IT PMO team be in a better position to articulate strategies and actions to address any incongruence in these perceptions and expectations.

This study aims to aid the IT PMO team in understanding the implications of expectations and perceptions of the values and functions of the IT PMO, empowering members to better service their stakeholders. ‘Expectations’ as operationalised in this study refer to what participants desire of the IT PMO (or what participants expect that the IT PMO should be doing), while ‘perceptions’ refer to what participants think the IT PMO is currently doing. Therefore, the research question to achieve the objective of this study is:
How can the IT PMO team be effectively assisted in developing a shared understanding and feel empowered to better manage their stakeholders’ expectations and perceptions?

To achieve this objective, a framework based on Quinn and Rohrbaugh’s (1981, 1983) competing values framework and Hobbs and Aubry’s (2007, 2010) PMO functions was used to elicit perceptions and expectations from stakeholders and to graphically display them for discussion amongst IT PMO team members.

RESEARCH METHODOLOGY AND DESIGN

In order to build knowledge while “grounded in reality” (Van de Ven, 2007:5), we have employed action research in this study where the researcher is actively involved in the investigation in a real-world situation with the aim to both improve it and to build knowledge (Checkland and Holwell, 1998) – in this case about the value delivery of the IT PMO. The nature of action research and its contributions to both practical problem-solving of real-world concerns and knowledge creation makes it an attractive proposition for conducting Information Systems (IS) research (Avison, Lau, Myers and Nielsen, 1999; Baskerville and Wood-Harper, 1996; Mathiassen, Chiasson and Germonprez, 2012; McKay and Marshall, 2001). In keeping with the ‘pragmatic philosophy’ of action research (Baskerville and Myers, 2004), we will be adopting a mix of methods, choosing and combining the appropriate methods of data collection – qualitative and quantitative, to study the various perceptions and expectations of the PMO team and its stakeholders. This decision has been informed by the growing acknowledgement of the value of ‘mixed-method’ research by academic researchers (Creswell, 2010; Greene and Hall, 2010; Teddlie and Tashakkori, 2010a, 2010b).

The organisation with the IT PMO that we investigated in this study is a Government statutory authority with more than 2,500 employees; and we use the pseudonym GOV1. The formation of the IT PMO had been regarded as an important initiative amongst the senior executives. However, there seemed to be the concerns as to whether or not the IT PMO had delivered the expected benefits to GOV1. Despite the fact that the IT PMO had been in operation for about five years, it was still relatively immature in terms of its internal processes and organisation. As the organisation was considering the restructure of the IT PMO, this research was regarded as timely and hence was endorsed by the senior management. The objectives of this study were seen to be beneficial to the organisation as they reorganised their IT PMO functions. With the assistance of the IT PMO leader, we identified participants from the IT PMO team as well as key stakeholders who represented the major business units that had been closely engaged with the IT PMO working on major IT projects. A total of ten participants were involved: five senior members of the IT PMO, and five senior-level business unit managers.

The first part of data collection in our study (see Table 1) involved the administration of a questionnaire to gather participants’ perceptions of what the IT PMO is currently doing/emphasising, as well as their expectations of what the IT PMO should be doing. As this study aims to investigate participants’ perceptions and expectations of the IT PMO’s core values as well as its roles and functions, there were two sets of questions in the questionnaire: questions on core values, based on the competing values framework (Quinn and Rohrbaugh, 1983); and questions about the roles and functions, based on an established list of PMO functions (Hobbs and Aubry, 2007, 2010). In Section 1 of the questionnaire, each participant separately scored his/her perceptions first (what core values that the IT PMO is currently emphasising) and then their expectations (what core values that the IT PMO should be emphasising) based on a Likert rating scale from 1 (very low emphasis) to 5 (very high emphasis). Data from all participants was then individually plotted on spatial diagrams (see Appendix 3), and the area bounded by all participants shaded. We established the list of core values of the IT PMO (see Appendix 1) based on Quinn and Rohrbaugh’s (1983) competing values framework, which is itself an aggregation of all the organisational effectiveness models in the literature. They are: (1) emphasis on flexibility and adaptability, (2) being consultative and inclusive of people from other business units, (3) resourcefulness in the acquisition of resources, (4) focus on being productive and efficient, (5) emphasis on planning and goal-setting, (6) focus on meeting set goals and objectives, (7) emphasis on stability and control, (8) emphasis on standardised processes and procedures, (9) emphasis on communication and information management, (10) team cohesion and staff morale amongst project teams, (11) emphasis on managing human resources within the project teams, and (12) focus on the development of human resources (i.e. training, mentoring) within the project teams. Also based on a Likert rating scale from 1 (strongly disagree) to 5 (strongly

2 Although interviews with participants were also conducted as part of the study, the discussion in this paper is based on the data from the questionnaire alone.
agree), the data of each participant’s scores of perceptions of the functions that the IT PMO is currently performing and expectations (of the functions that are considered important for the IT PMO to perform) in Section 2 of the questionnaire were individually plotted on spatial diagrams (see Appendix 4), and the area bounded by responses from all participants shaded. The roles and functions of the IT PMO (see Appendix 2) were developed for this study based on Hobbs and Aubry’s (2007, 2010) empirically-established list of PMO functions.

<table>
<thead>
<tr>
<th>Part of Study</th>
<th>Data Collection Method</th>
<th>Participants</th>
</tr>
</thead>
</table>
| Part 1 of Study (May 2013 – Jul 2013) | **PMO Questionnaire** Administration of a Likert-scale questionnaire comprising the following two sections:  
  • Section 1: questions eliciting participants perceptions and expectations of the core PMO values that the IT PMO is currently, and should be emphasising  
  • Section 2: questions eliciting participants’ perceptions and expectations of the PMO functions that the IT PMO is currently, and should be performing. | Senior Business & Management Team (5 members); Senior IT PMO Team (5 members) |
| Part 2 of Study (Jul 2013 – Sep 2013) | **Workshop 1 (Jul 2013)** Presentation of findings from **PMO Questionnaire** data by Research team, and facilitation of discussion and articulation of strategies amongst IT PMO team members to better engage stakeholders, followed by administration of a feedback questionnaire. | Senior IT PMO Team (5 members) |
|                                        | **Workshop 2 (Sep 2013)** Presentation by IT PMO team of its initial efforts in articulating strategies to improve perceptions of value delivery amongst key stakeholders. | Senior IT PMO Team (4 members) |

Table 1. Data Collection Summary

The data from the questionnaire was then used to graphically plot profile diagrams and then analysed to study the perceptions and expectations of the IT PMO team as well as its stakeholders. We adopted the visual representation of Quinn and Rohrbaugh’s (1983) competing values framework as it is described as “visually and cognitively comprehensive” (Thompson, 1993:102) and it gives people a better sense of data, making it easier to perceive relationships, make comparisons, and identify patterns (Cameron, Quinn, DeGraff and Thakor, 2006). Data is therefore illustrated as a profile on the model, giving rise to a more implicit interpretation to help IT PMO teams develop shared understanding, and therefore be better equipped to develop strategies to manage their stakeholders’ perceptions and expectations.

For the second part of our study, the findings based on the questionnaire data were presented in a workshop session (Workshop 1) with five senior members of the IT PMO team to facilitate in the articulation of strategies amongst the workshop participants to better engage their stakeholders. After the workshop session, all participants completed a short feedback questionnaire with eight Likert-scale questions and an open-ended comments section to elicit their perceptions of the effectiveness of the workshop, framework and diagrams, and the overall engagement process. At the request of the IT PMO team from GOV1, a second workshop session (Workshop 2) was conducted six weeks later where the IT PMO team shared the progress the IT PMO had made.

RESULTS

At the first workshop session (Workshop 1) with the IT PMO team, we presented the data from the questionnaire, employing a visual representation that enabled us to graphically present both the current state of the stakeholders’ and the IT PMO team’s perceptions and expectations of the IT PMO. This was apparently helpful in guiding discussion where members of the IT PMO team attempted to make sense of the differences in perceptions both within their own team and in the stakeholders’ perspectives. This provided the opportunity for the IT PMO team to work towards a shared understanding of some of the key issues they faced in gaining recognition for their contribution to the organisation (for example, setting the IT PMO’s vision, goals and objectives, which were previously not in place as there had been a recent restructure of the IT PMO. The information on the perceptions and expectations of the core values of the IT PMO was helpful in this respect.)

Core Values that the IT PMO should be Emphasising
Information from the diagrams (see Appendix 3) are summarised and presented in Table 2, comparing between the stakeholders’ and IT PMO team members’ perceptions and expectations of the core values of the IT PMO. From the first column in Table 2, we can see that the five core values that stakeholders want the IT PMO to emphasise are: (1) productivity and efficiency, (2) planning and setting goals and objectives, (3) meeting goals and objectives, (4) effective communications and information management, and (5) development of human resources, such as training, within project teams. The last column in Table 2 also highlights the four core values that the IT PMO team considers important to focus on. Here, it is apparent that although there are two core values (communications and information management, and development of human resources) that both the IT PMO team and its stakeholders agree is important, the IT PMO team needs to also focus on productivity and efficiency, planning, setting, and meeting its goals and objectives. On the other hand, the other two core values that the IT PMO team considers important – flexibility and adaptability, and standardised processes and procedures – are not really considered important by all its stakeholders. The table (see Table 2) and diagrams (in Appendix 3) help demonstrate a clear gap between the values that the IT PMO team members currently think they are emphasising/should emphasise, and what their key stakeholders really want of them.

This information provides an important insight for the IT PMO team – helping the team better understand its stakeholders. It essentially highlights what is critical to its key stakeholders, gives the IT PMO team an idea of where it needs to be, and shows the team where the IT PMO is on- and off-track. The IT PMO team must therefore develop strategies towards achieving an alignment in both key stakeholders’ and the IT PMO team’s sets of expectations in order that the IT PMO is considered by its stakeholders as delivering value.

<table>
<thead>
<tr>
<th>Stakeholder Expectations (core values stakeholders want the IT PMO to emphasise)</th>
<th>Stakeholder Perceptions (core values stakeholders perceive the IT PMO is/is not emphasising)</th>
<th>IT PMO Perceptions (core values the team perceives the IT PMO is/is not emphasising)</th>
<th>IT PMO Expectations (core values the team wants the IT PMO to emphasise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emphasis on productivity &amp; efficiency</td>
<td>Team members think the IT PMO is somewhat emphasising productivity &amp; efficiency</td>
<td>Emphasis on effective communications &amp; information management</td>
<td>Emphasis on the development of human resources (i.e. training) within project teams</td>
</tr>
<tr>
<td>Emphasis on planning &amp; setting of goals &amp; objectives</td>
<td></td>
<td></td>
<td>Emphasis on flexibility &amp; adaptability</td>
</tr>
<tr>
<td>Emphasis on meeting set goals &amp; objectives</td>
<td></td>
<td></td>
<td>Emphasis on standardised processes &amp; procedures</td>
</tr>
<tr>
<td>Emphasis on effective communications &amp; information management</td>
<td>Stakeholders do not think the IT PMO is emphasising the development of human resources</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emphasis on the development of human resources (i.e. training) within project teams</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Stakeholder & IT PMO Team Perceptions & Expectations of the Core Values of the IT PMO

An interesting finding that was revealed from the diagrams in this study (see Appendix 3, Figure A3a) was the lack of alignment amongst the IT PMO team members themselves in their perceptions of the core values being emphasised in the IT PMO. The findings caused considerable interest and discussion amongst the members of the IT PMO during the workshop as it drew their attention to the state of misalignment the IT PMO team was in. This clearly demonstrates the need for the IT PMO team to change its strategy and its values being emphasised as a team. The IT PMO leader felt that that this study was conducted at an apposite time [in light of the restructure the IT PMO team was experiencing], and the findings have motivated the IT PMO team to develop strategies to redefine the goals and objectives of the IT PMO.
IT PMO Functions that are Important to Stakeholders

Information from the diagrams (see Appendix 4) is summarised and presented in Table 3, comparing between stakeholders’ and IT PMO team members’ perceptions and expectations of the roles and functions of the IT PMO.

<table>
<thead>
<tr>
<th>Stakeholder Expectations (PMO functions the stakeholders deem important for IT PMO to do)</th>
<th>Stakeholder Perceptions (PMO functions the stakeholders perceive the IT PMO is/is not doing)</th>
<th>IT PMO Perceptions (PMO functions the team members perceive the IT PMO is/is not doing)</th>
<th>IT PMO Expectations (PMO functions the team members deem important for the IT PMO to do)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regularly track &amp; monitor all IT projects</td>
<td>Stakeholders think the IT PMO is regularly tracking &amp; monitoring all IT projects</td>
<td></td>
<td>Regularly track &amp; monitor all IT projects</td>
</tr>
<tr>
<td>Enforce project governance for all IT projects</td>
<td>Stakeholders do not think the IT PMO is enforcing the project governance for all IT projects</td>
<td>Team members think the IT PMO is enforcing the project governance for all IT projects</td>
<td>Enforce project governance for all IT projects</td>
</tr>
<tr>
<td>Prescribe standardised IT project management methodologies for the organisation</td>
<td></td>
<td></td>
<td>Prescribe standardised IT project management methodologies for the organisation</td>
</tr>
<tr>
<td>Promote the adoption of standardised IT project management methodologies</td>
<td>Stakeholders do not think the IT PMO is promoting the adoption of standard IT project management methodologies</td>
<td></td>
<td>Promote the adoption of standardised IT project management methodologies</td>
</tr>
<tr>
<td>Develop performance measures for IT project managers</td>
<td></td>
<td></td>
<td>Develop performance measures for IT project managers</td>
</tr>
<tr>
<td>Promote soft skills (i.e. communications, interpersonal, etc.) amongst project team members</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provide project management tools for IT project managers and IT project teams</td>
<td></td>
<td></td>
<td>Provide project management tools for IT project managers and IT project teams</td>
</tr>
<tr>
<td>Participate (i.e. sharing expertise, experience) in the development of business case for IT projects</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Track &amp; ensure that IT projects are aligned with business strategy</td>
<td>Stakeholders do not think the IT PMO is tracking and ensuring this</td>
<td></td>
<td>Track &amp; ensure that IT projects are aligned with business strategy</td>
</tr>
<tr>
<td>Track &amp; ensure the delivery of expected benefits from IT projects</td>
<td>Stakeholders do not think the IT PMO is tracking and ensuring this</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keep up with current business trends</td>
<td></td>
<td></td>
<td>Keep up with current business trends</td>
</tr>
<tr>
<td>Implement &amp; manage a ‘lessons-learned’ knowledge base</td>
<td></td>
<td></td>
<td>Implement &amp; manage a ‘lessons-learned’ knowledge base</td>
</tr>
<tr>
<td>Ensure ‘lessons-learned’ are effectively communicated to subsequent IT projects</td>
<td></td>
<td></td>
<td>Ensure ‘lessons-learned’ are effectively communicated to subsequent IT projects</td>
</tr>
<tr>
<td>Conduct &amp; document post-project reviews</td>
<td></td>
<td></td>
<td>Conduct &amp; document post-project reviews</td>
</tr>
</tbody>
</table>

Table 3. Stakeholder & IT PMO Team Perceptions & Expectations of the Roles & Functions of the IT PMO

In this case, there were a total of 14 roles and functions that the stakeholders deemed important that the IT PMO team should perform (see Table 3). The IT PMO team, on the other hand, considered 24 PMO functions as
important for the IT PMO to do; out of which only eleven were in alignment with the stakeholders. Again, this information is important for the IT PMO team, highlighting the functions considered critical by key stakeholders and helping equip members develop strategies towards achieving an alignment in expectations.

Another useful set of data for the IT PMO team is the information on the perceptions of the IT PMO (see columns 2 & 3 in Table 3). It provides the actionable areas for the IT PMO team to focus on in order to be perceived as delivering value by these key stakeholders. For example, for the function to ‘enforce project governance for all IT projects’, both key stakeholders and the IT PMO team agree is important for the IT PMO to do this. However, whilst the IT PMO team members think that the IT PMO is enforcing governance of all IT projects, the key stakeholders think otherwise. Here, the diagrams (see Appendix 4) together with Table 3 have made visible this issue for the IT PMO team to diagnose and address thereafter.

The findings about the stakeholders’ perceptions and expectations have highlighted the differing and competing views the stakeholders have of the IT PMO, confirming the need for the IT PMO team members to develop effective strategies, for example establishing an effective communications strategy, to better engage their stakeholders. The summary tables (see Tables 2 & 3), together with the four spatial diagrams (see Appendices 3 & 4) offer a wealth of information for the IT PMO team to better understand its stakeholders’ perceptions and expectations – potentially beyond what has been discussed. However, in keeping with the scope of this study, we have limited our discussion to the above results.

**DISCUSSION**

Based on the points of view of the workshop participants [senior IT PMO team members] as a measure of IS effectiveness (Seddon, Staples, Patnayakuni and Bowtell, 1999) as well as our own reflections of this action research study, the visual framework is considered effective in graphically representing the perceptions and expectations of stakeholders and helping the IT PMO gain an insight of what their stakeholders were thinking. With this awareness and understanding, members of the IT PMO team were empowered to think about strategies to address the differences/gaps in these perceptions and expectations and hence help them work towards being perceived by their key stakeholders as delivering value. It is important to note that these spatial diagrams offer the audience, in this case, the IT PMO team members, the visibility of what the stakeholders want relative to their own perceptions and expectations. With this visibility, they can then recognise whether or not, and what they have to change.

The first workshop session (Workshop 1) was ostensibly beneficial for the IT PMO team, as we observed very enthusiastic participation amongst all the participants, both during the presentation of findings as well as in the discussions. It was useful in that it facilitated a shared understanding of the perceptions and expectations of key stakeholders amongst the participants. As it was presented in a non-threatening manner with all data anonymised, we did not detect any sign of defensiveness amongst the participants. Instead, the presentation of the findings and the facilitation of discussion got all IT PMO team members heavily engaged in constructive discussion despite the fact that some findings showed a disagreement or misalignment amongst IT PMO team members. The insights helped guide team members discuss and resolve some of the disagreements or lack of alignment within their own team. They had recognised that they were too busy managing IT projects but were not attending to their own processes, as to why they exist, and how they were communicating to others in the organisations what they were doing or have accomplished. In their feedback, all IT PMO team members involved agreed that the overall study was effective in highlighting their stakeholders’ perceptions and expectations. The visual diagrams and the summary tables used in the workshop were very helpful and clear in communicating the findings and provided the visibility for IT PMO team to compare and act accordingly. All team members felt that the IT PMO was better equipped to engage their stakeholders as a result of the workshop. All team members of the IT PMO were satisfied with the outcomes of the workshop, and the IT PMO leader affirmed this with his/her comments that the study offered “extremely valuable insight for the group”.

At the second workshop session (Workshop 2), the senior IT PMO team members from GOV1 presented their first efforts in articulating their vision. They presented their strategies on (1) defining the operating model of the IT PMO; (2) clarifying the role of the IT PMO (in what it does, how it is done, and how it is operated as a whole); and (3) communicating key aspects of (1) and (2) to key stakeholders as well as the rest of the business in order to build credibility and recognition that it is striving to deliver value to its stakeholders. Although their work is still ongoing, the IT PMO team seemed to much clearer in its own objectives and purpose after the workshop sessions.
CONCLUSION

This study is an initial trial of the visual framework to investigate the core values and functions of the IT PMO using the concepts of perceptions, expectations and satisfaction. We believe we have effectively answered the research question by demonstrating the effectiveness of the overall engagement process including the questionnaire and visual representations, as well as the workshop sessions in helping facilitate the development of shared understanding amongst the IT PMO team members. In addition, through the work presented by the IT PMO team in Workshop 2, we have evidence that they [the IT PMO team members] have been empowered to start thinking of strategies aimed at improving perceptions of value delivery amongst their key stakeholders.

In summary, this study has shown this instrument to be useful in empowering the IT PMO team to recognise whether or not it has to change, and to go about making these changes; and therefore be able to demonstrate its organisational value.

REFERENCES


**APPENDIX 1**

<table>
<thead>
<tr>
<th>Core values the PMO is perceived/expected to emphasise</th>
<th>Four quadrants in Quinn &amp; Rohrbaugh’s (1983) Competing Values Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Flexibility and adaptability</td>
<td>Open Systems: concerned with keeping up with the changing external environment, therefore underscoring the importance of flexibility, readiness, innovation and growth.</td>
</tr>
<tr>
<td>2 Consultative and inclusive of people from other business units</td>
<td>Rational Goal: concerned with the organisation maintaining competitiveness, hence giving emphasis to goal-setting, efficiency, and productivity.</td>
</tr>
<tr>
<td>3 Resourcefulness in acquiring resources</td>
<td>Internal Process: concerned with organising and structuring the organisation, therefore underscoring the importance of process stability, communications, and information management.</td>
</tr>
<tr>
<td>4 Productivity and efficiency</td>
<td></td>
</tr>
<tr>
<td>5 Planning and goal-setting</td>
<td></td>
</tr>
<tr>
<td>6 Meeting its set goals and objectives</td>
<td></td>
</tr>
<tr>
<td>7 Stability and control</td>
<td></td>
</tr>
<tr>
<td>8 Standardised processes and procedures</td>
<td></td>
</tr>
<tr>
<td>9 Communication and information management</td>
<td></td>
</tr>
<tr>
<td>10 Team cohesion and staff morale amongst project teams</td>
<td></td>
</tr>
<tr>
<td>11 Management of human resources within the project teams</td>
<td></td>
</tr>
<tr>
<td>12 Development of human resources (i.e. training, mentoring) within the project teams</td>
<td>Human Relations: represents an organisation’s concern for its employees, therefore emphasising human resource development, group cohesiveness and morale.</td>
</tr>
</tbody>
</table>

Table A1. Core Values of the IT PMO
APPENDIX 2

<table>
<thead>
<tr>
<th>PMO functions</th>
<th>Five main groups of PMO functions Hobbs &amp; Aubry (2007, 2010)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Report status of IT projects, IT programs (groups of IT projects), or IT portfolios (groups of IT programs) to senior management</td>
<td>Monitoring, controlling, and reporting project performance</td>
</tr>
<tr>
<td>2. Regularly track/monitor all IT projects</td>
<td></td>
</tr>
<tr>
<td>3. Regularly track/monitor only selected IT projects</td>
<td></td>
</tr>
<tr>
<td>4. Use appropriate computer-based tools to monitor IT projects</td>
<td></td>
</tr>
<tr>
<td>5. Directly manage all IT projects</td>
<td></td>
</tr>
<tr>
<td>6. Directly manage only selected IT projects</td>
<td></td>
</tr>
<tr>
<td>7. Enforce project governance for all IT projects</td>
<td></td>
</tr>
<tr>
<td>8. Enforce project governance for only selected IT projects</td>
<td></td>
</tr>
<tr>
<td>9. Prescribe standardised IT project management methodologies for the organisation</td>
<td></td>
</tr>
<tr>
<td>10. Enforce the implementation of standardised IT project management methodologies</td>
<td></td>
</tr>
<tr>
<td>11. Promote the adoption of standardised IT project management methodologies</td>
<td></td>
</tr>
<tr>
<td>12. Provide project management training for IT project managers</td>
<td></td>
</tr>
<tr>
<td>13. Provide training for all staff involved with IT projects in the organisation</td>
<td></td>
</tr>
<tr>
<td>14. Develop performance measures for IT project managers</td>
<td></td>
</tr>
<tr>
<td>15. Measure performance of IT project managers</td>
<td></td>
</tr>
<tr>
<td>16. Define project management competency requirements for IT project managers</td>
<td></td>
</tr>
<tr>
<td>17. Employ only IT project managers with required project management competencies</td>
<td></td>
</tr>
<tr>
<td>18. Promote soft skills (i.e. communications, interpersonal, etc.) amongst project team members</td>
<td></td>
</tr>
<tr>
<td>19. Provide mentoring and project management advice for IT project managers</td>
<td></td>
</tr>
<tr>
<td>20. Provide project management tools for IT project managers and IT project teams</td>
<td></td>
</tr>
<tr>
<td>21. Participate in employment activities (i.e. recruit, select, evaluate, etc.) of IT project managers</td>
<td></td>
</tr>
<tr>
<td>22. Participate in the selection and prioritisation of all IT projects</td>
<td>Multi-project management, including prioritizing, coordinating and resource allocation</td>
</tr>
<tr>
<td>23. Participate in the selection and prioritisation of only selected IT projects</td>
<td></td>
</tr>
<tr>
<td>24. Participate (i.e. sharing expertise, experience, etc.) in the development of business case for IT projects</td>
<td></td>
</tr>
<tr>
<td>25. Manage one or more IT programs (groups of IT projects) and/or IT portfolios (groups of IT programs)</td>
<td></td>
</tr>
<tr>
<td>26. Manage the allocation of resources (i.e. staff, assets, etc.) across IT projects</td>
<td></td>
</tr>
<tr>
<td>27. Have the power to terminate any IT project</td>
<td></td>
</tr>
<tr>
<td>28. Track and ensure that IT projects are aligned with business strategy</td>
<td>Strategic management and planning</td>
</tr>
<tr>
<td>29. Track and ensure the delivery of expected benefits from IT projects</td>
<td></td>
</tr>
<tr>
<td>30. Keep up with current information and communications technology trends</td>
<td></td>
</tr>
<tr>
<td>31. Keep up with current business trends</td>
<td></td>
</tr>
<tr>
<td>32. Demonstrate to senior management that IT delivers business value</td>
<td></td>
</tr>
<tr>
<td>33. Implement and manage a ‘lessons-learned’ knowledge base</td>
<td>Organizational learning, including post-implementation reviews, project audits, and managing lessons-learned databases</td>
</tr>
<tr>
<td>34. Ensure ‘lessons-learned’ are effectively communicated to subsequent IT projects</td>
<td></td>
</tr>
<tr>
<td>35. Conduct and document post-project reviews</td>
<td></td>
</tr>
<tr>
<td>36. Archive project documentation</td>
<td></td>
</tr>
</tbody>
</table>

Table A2. Roles & Functions of the IT PMO
APPENDIX 3

Figure A3a. Core Values that should be emphasised in the IT PMO (Stakeholders’ Expectations)

Figure A3b. Core Values that are being emphasised in the IT PMO (Stakeholders’ Perceptions)
Figure A3c. Core Values that should be emphasised in the IT PMO (IT PMO Team’s Expectations)

Figure A3d. Core Values that are being emphasised in the IT PMO (IT PMO Team’s Perceptions)
Figure A4a. PMO Functions that should be performed by the IT PMO (Stakeholders' Expectations)
APPENDIX 4

Figure A4b. PMO Functions that are being performed by the IT PMO (Stakeholders’ Perceptions)
APPENDIX 4

Figure A4c. PMO Functions that should be performed by the IT PMO (IT PMO Team’s Expectations)
APPENDIX 4

Figure A4. PMO Functions that are being performed by the IT PMO (IT PMO Team’s Perceptions)
Highlighting Communication Activities and Inefficiencies Between Agile vs. Waterfall Methods: An Agent Based Model of Knowledge Sharing

Meghann L. Drury-Grogan
The Schools of Business
Fordham University
5 Columbus Circle, 11-08
New York, NY 10019
mdrury@fordham.edu

Deanna M. Kennedy
School of Business
University of Washington Bothell
18115 NE Campus Way
Bothell, WA 98011
dkennedy@uwb.edu

ABSTRACT
We employed agent-based simulation techniques to create a dynamic multi-level team environment to study communication activities as knowledge sharing occurred. We examined knowledge seekers and knowledge providers who act and react to one another's communication behavior on Waterfall versus Agile teams using rich versus lean media to answer research questions regarding inefficient use of team members. The simulation model was checked for validity against assumptions that project management method drives project schedule and communication media motivates the number of meetings. Results further indicate that (a) slightly more knowledge seekers exceed their knowledge need on Agile teams using lean versus rich media; (b) knowledge overage was reduced by utilizing a Waterfall rather than Agile method, and through the use of lean media; and (c) the maximum time wasted by team members who completed gathering knowledge to meet their initial needs was on Agile teams using lean media.

Keywords
Project Management Procedures; Knowledge Management; Communication; Team Communication; Waterfall Software Development, Agile Software Development; Agent-based Modeling.

INTRODUCTION
Successful projects depend on members to create, apply, and share knowledge (Fong, 2003; Nonaka and Takeuchi, 1995; Yuan, Zhang, Chen, Vogel and Chu, 2009). We define knowledge as “information possessed in the mind of individuals: it is personalized information (which may or may not be new, unique, useful, or accurate) related to facts, procedures, concepts, interpretations, ideas, observations, and judgments” (p. 109, Alavi and Leidner, 2001). The recognized need for knowledge sharing, especially among project team members, has created a burgeoning area of research about the practices and facilitation (e.g., Wang and Ko, 2012). However, while knowledge sharing has been conceptualized as a dynamic process with multi-level scenarios across individual, team, and organization (Nonaka and Takeuchi, 1995), little has been done to show how these conceptualizations could play out over time. As a dynamic process, time plays a critical role in understanding team processes and outcomes (Kozlowski, Chao, Grand, Braun, and Kuljanin, 2013). Additionally, interactive processes, such as knowledge sharing, may depend on the effectiveness of individual level activities (Hackman, 2003; Klein and Kozlowski, 2000). To address the gap in our understanding of knowledge sharing on project teams as a temporal and multi-level phenomenon, we aim to examine knowledge sharing between team members and gain insights about team outcomes.

An individual level activity for sharing knowledge is through team communication where knowledge providers and knowledge seekers connect to fill gaps in knowledge and coordination (Wang and Ko, 2012). Researchers acknowledge the importance of team communication for project team performance (e.g., Hsu, Shih, Chiang and Liu, 2012; Park and Lee, in press) over most other PMBOK (Project Management Body of Knowledge) knowledge areas (Peltoniemi, Jokinen and Mönkkönen, 2004). Recognizing the importance of communication practices, organizations may try to affect team communication through (1) the planned use of different communication media; and (2) the project structure which sets interactive meeting time constraints.
First, the communication media, and particularly the richness of the media available for team member interactions, may affect the direction of communication and capacity to convey cues (Daft and Lengel, 1986) that may influence the timeliness and amount of knowledge shared. Second, organizations often implement Project Management (PM), a system with the intentional purpose of promoting the gathering of knowledge from others (Gray and Meister, 2004). The Waterfall method brings team members together periodically to elaborately discuss milestone achievements by subject area (Jurison, 1999), while the Agile method calls for more frequent yet shorter interaction sessions (Cusumano and Smith, 1995; Hass, 2007). The PM method may set different schedules and frequency for meetings between members further affecting the timeliness of knowledge acquisition.

While organizational inputs may extend the time it takes some members to seek out knowledge, these delays may in turn lead other members to inefficiently utilize their time as they gather and process information beyond what they need to complete their own job. As such, some members may be subject to information overload which brings about fatigue, errors, and conflict (Edmunds and Morris, 2000) as they continue to meet others and gain information. Moreover, the energy wasted in attending unnecessary meetings can lead to burnout (Salanova, Peiro and Schaufeli, 2002). We seek to understand how communication media and PM methods differentially enable knowledge inefficiencies such that some members experience overload and wasted energies. To draw out the outcomes of interest, we track team communication where we assume that knowledge is being acquired during conversations between team members. We pursue three research questions (RQ):

1. To what extent does the available media’s richness and PM method schedule, Waterfall or Agile, affect the number of team members exceeding their knowledge need?

2. To what extent does the available media’s richness and PM method schedule, Waterfall or Agile, affect the maximum amount of knowledge by which a team member exceeds his/her knowledge need?

3. To what extent does the available media’s richness and PM method schedule, Waterfall or Agile, affect the maximum amount of time a team member wastes after filling his/her knowledge need?

Our contribution to the project management literature comes from taking a temporal and multi-level perspective of knowledge sharing to uncover how team members may be inefficiently utilized on projects depending on communication media use and PM method. Moreover, we are able to make novel contributions by studying teams staffed in similar ways, i.e., the same amount of knowledge seekers and knowledge providers with similar requirements while varying organizational inputs. To do so, we employ agent-based modeling to create a dynamic team environment. Researchers have increasingly employed simulation techniques to provide insights into complex interactions among organizational members and their tools (e.g., Espinosa and Carmel, 2003; Noguiera and Raz, 2006). Agent-based models are beneficial for investigating emergent phenomena that results from interactions among agents (Davis, Eisenhardt and Bingham, 2007; Harrison, Lin, Carroll and Carley, 2007). Herein, team members act and react to one another’s communication behavior as knowledge sharing occurs. As such, our parsimonious model provides insights meant to inform future research about potential areas to target regarding knowledge sharing on project teams.

BACKGROUND LITERATURE

This section presents a temporally based, multilevel conceptualization of knowledge sharing to frame our research.

Knowledge Sharing

Teams are groups of individuals that work together, are dependent upon one another and have one or more tasks to perform in order to accomplish various goals (Hackman, 1990; Mayer, Davis and Schoorman, 1995). Specifically, team members rely on each other for knowledge to produce a successful output based on the collective contribution of all team members (Katzenbach and Smith, 2005). As Nonaka and Takeuchi (1995) suggest, teams are agents of knowledge creation whereby individual, team, and organizational contexts interact. Through dialogue team members can share explicit knowledge that can be converted into tacit knowledge (Nonaka and Konno, 1998) and, ultimately, collective learning (Fong, 2003). Establishing places, such as at meetings, where knowledge seekers and knowledge providers can open a dialogue may promote knowledge sharing (Nonaka and Konno, 1998). While the interaction between tacit and explicit knowledge has been conceptualized as a spiraling process, when spread out over time, the course of activities may fluctuate between explicit and tacit activities. Figure 1 demonstrates the conceptualization...
of the way knowledge creation plays out over time through the progression of explicit knowledge sharing at formal meetings (e.g., planning meetings, scrum meetings and retrospective meetings in agile or the planning meetings and milestone meetings in waterfall) and tacit knowledge creation when working individually. Moreover, we are interested in how these activities unfold under different organizational inputs including available communication media and meeting schedules per PM method. We discuss these organizational inputs in greater detail below.

**Figure 1. Knowledge Sharing Conceptualization**

### Media Richness

Temporal differences in interpersonal communication activities may be driven by the communication media being used for interactions. Media richness theory, advanced by Daft and Lengel (1986), suggests that communication channels differ in cue-carrying capacity (i.e., the types and amount of information made available to be processed into knowledge that can be effectively transmitted). As such, different communication media may be considered along a media richness continuum that is anchored by rich media and lean media (Chidambaram and Jones, 1993). Rich media generally has a high cue-carrying capacity because it allows for multiple types and amounts of information to be transferred (i.e., verbal, paraverbal, and nonverbal) (Daft and Lengel, 1986). The types of rich media teams use include face-to-face and other tools such as voice over internet protocol (VOIP) or video-conferencing that allow teams to collaborate synchronously as collaboration can occur via information technologies that allow people to interact with each other simultaneously (e.g., Baker, 2002; Drury and Williams, 2002), though synchronous rich media can increase communication time and costs on projects (Natu and Kennedy, 2012). Based on these findings about cue-carrying capacity and duration, we associate rich media with longer durations but more knowledge content being transferred between knowledge seeker and provider.

Lean media is limited in terms of cue-carrying capacity because it restricts non-verbal and paraverbal cues (Straus, 1997). As a result of this limitation, researchers suggest that teams may exchange shorter messages through these means than through richer media (Boyle, Anderson and Newlands, 1994). Interestingly, however, researchers suggest that when using lean tools, such as email or instant messaging, team members are prone to use them as much if not more often than richer media options (LaToza, Venolia and DeLine, 2006). Thus, we assign shorter messages when lean media is used with less knowledge content being transferred in any one exchange between seeker and provider than using rich media.

### Project Management Methods for Software Development

PM methods are used to facilitate many software development (SD) projects as these types of projects are notably knowledge-intensive, requiring the integration of knowledge from various domains (Patnayakuni, Rai and Tiwana, 2007). PM methods set meeting schedules so that the team members adequately interact around the six project management functions of planning, organizing, executing, monitoring, reporting, and controlling to manage the knowledge quality (Gannon, 1994). As well, meetings allow members to effectively share knowledge because not all team members will possess the knowledge for each of these PM activities (Chau, Maurer and Melnik, 2003) as each has their own limited experience on which to draw (Drury, Acton, Conboy and Golden, 2011). Two methods that have received little direct comparison and are the basis for this research are the Waterfall and Agile methods.
Waterfall Software Development

The more traditional approach and foundational SD standard (Sommerville, 1996) is the Waterfall method where the optimal SD process is a linear, sequential series of project management phases (Cusumano et al., 1995). The classic Waterfall method was modeled after hardware design projects where engineers could predict how parts of the system would interact (Royce, 1970). It is a unidirectional, top down process with a sequence of activities that is non-iterative (Fitzgerald, 2000; Sommerville, 1996) where phases of the process move from one to the next. In this sequential design process, the progress of a project flows steadily downwards through the phases like a waterfall: requirements specification, design, implementation (coding), integration, testing and debugging, installation, and maintenance. One can only move onto the next phase when the preceding phase is completed accurately (Royce, 1970).

Waterfall is a traditional PM method with disciplined, deliberate planning and control methods. Such teams focus on the process of work rather than team members’ creativity as work allocation clarifies what and how work will be done within a specific timeframe (Taylor, 1998). The project is planned up front as tasks are completed one after another in an orderly sequence in distinct project life cycle phases that are easily recognizable and not revisited upon completion (Hass, 2007). The design, development, and testing takes place by different teams, and these teams only put these modules together and test the entire system in the last phase of the project, which usually requires reworking the modules and writing new code to correct problems from the interactions of the modules due to unforeseen problems, mistakes, miscommunications, or design changes that occurred during the project (Cusumano et al., 1995; Jurison, 1999). This process is efficient so long as system requirements remain stable until the end of the project because all requirements are captured before any design and development occurs so customer feedback is not incorporated (Chau et al., 2003). Thus, the major advantage of Waterfall is that it provides a structure for organizing and controlling a SD project, though it must identify user requirements accurately at the beginning of the project (Cusumano et al., 1995).

Agile Software Development

However, past research shows Waterfall was not delivering cost-effective and user-driven software to customers (Lyytinen, 1987) as projects were unpredictable and rarely followed a sequential flow where customers could identify all requirements upfront (Hass, 2007). Agile allows for iterative development that simultaneously designs products and processes (Ballard and Howell, 2003) on small collaborative SD teams (Dybå and Dingsøyr, 2008). These teams work under extreme time pressure to develop and deliver working software to customers in short iterations (Fitzgerald, Hartnett and Conboy, 2006; Fowler and Highsmith, 2001), a time-boxed period of fixed length, often two weeks’ duration (Schwaber and Beedle, 2002). Iterative development is the process of building a system within a short period of time (Larman, 2004). Because of these short iterations, Agile teams are able to respond quickly to changes in the business environment, technology, and customer requirements by continually redesigning and adapting development processes (Henderson-Sellers and Serour, 2005) as they only plan for one iteration at a time.

Agile team interactions focus on communication to convey knowledge (Cockburn and Highsmith, 2001). Researchers report that the most efficient and effective method of conveying knowledge for Agile teams is face-to-face communication (Fowler et al., 2001), so teams work in close proximity to foster face-to-face communication, timely feedback, and informal social interaction, though they can experience a lack of team engagement when members feel a lack of decision ownership and empowerment (Drury, Conboy and Power, 2012). Proximity refers to the actual physical distance between people (Hinds and Kiesler, 2002) as collocation is a key Agile tenet (Green, Mazzuchi and Sarkani, 2010).

Another key tenet to Agile is frequent and continuous communication with the type of communication medium key. There is an emphasis on richer communication medium rather than lean ones, particularly during the beginning of the development project (Green et al., 2010). While lean technologies of today have created more cost-effective ways to communicate over vast differences, they cannot fully replace the power of rich communication (Carmel and Agarwal, 2001). Nevertheless, Agile teams like other teams must balance between rich and lean medium as organizations strive to reduce development timelines, deliver products to market faster, and leverage cheaper SD resources across the world (Green et al., 2010).

METHOD
In this study, agent-based simulation is employed to study a 2 (PM method) x 3 (communication media scenario) design. We use this approach because an abstract model that characterizes real-world environments can provide preliminary insights for these other methods that direct better use of time and budgets towards viable and meaningful effects in other settings (Harrison et al., 2007). The scope of the current model is to reproduce the basic and general communication patterns of team members working on one stage of a SD project. When designing the model we followed practical suggestions (Davis et al., 2007) and established guidelines for decision making and model construction (Rand and Rust, 2011). Figure 2 illustrates the progression through the design decisions. Table 1 reports the decisions for each of the simulation elements utilized in simulation procedures. To ensure our program has face validity, we (1) model parameters after published parameters, theorized relationships, and noted insights from past studies, and (2) provide a preliminary test in the results section to show that the parameters have the expected effects that characterize real-world situations of Waterfall and Agile projects, and projects using rich and/or lean media. The simulation is programmed in NetLogo (Wilensky, 1999).

### Table 1: Simulation Parameters

<table>
<thead>
<tr>
<th>Decision Area</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environment</td>
<td><strong>PM method</strong> to schedule time allotment for meeting and for individual work where knowledge is not shared, with the Waterfall method taking longer than the Agile method (Cusumano et al., 1995). We therefore make the following time allotment for the two PM methods: W) Waterfall method allots 16 hours (time steps) for meeting, 20 hours for individual work. A) Agile method allots 4 hours for meeting, 10 hours for individual work. <strong>Media Scenario</strong> to determine the available media richness. The different scenarios set meeting duration (how much of available meeting time used) and content amount (how much of the provider’s knowledge is shared) (Chidambaram et al., 1993; Daft et al., 1986; LaToza et al., 2006). R) Rich media scenario: 100% meeting duration, 100% content. L) Lean media scenario: 25% meeting duration, 50% content. M) Mix media scenario: equal probability of using rich or lean media and commiserate duration and content quantity.</td>
</tr>
<tr>
<td>Agents</td>
<td><strong>Team members</strong> on a SD project. Knowledge seekers: 16 members (i.e., 2 sub-teams) who need knowledge. Knowledge providers: 16 members (i.e., 2 sub-teams) who provide knowledge to seekers when not already in a meeting with another knowledge seeker.</td>
</tr>
<tr>
<td>Properties</td>
<td><strong>Knowledge need</strong> of knowledge seekers randomly drawn from a Poisson distribution with mean of 40. <strong>Knowledge expertise</strong> of knowledge providers randomly drawn from a Poisson distribution with mean of 5. (This approach assigns positive integer values to knowledge seekers that are sufficiently large enough to require communication with at least one knowledge provider; such interactions may reflect a project with...</td>
</tr>
</tbody>
</table>
Behaviors

At each time step knowledge seekers are either moving to meet a knowledge provider, in a meeting with a knowledge provider agent, or are working individually and not gaining knowledge from another member.

- **Movement**: Knowledge seekers will move in random directions until a knowledge provider is encountered. Movement reflects the difficulty that researchers have noted in members’ abilities with locating and accessing knowledge resources (e.g., Chang, Yen, Chiang and Parolia, 2013).
- **Meeting**: A meeting is initialized and collaboration commences when a knowledge seeker meets an idle knowledge provider. After the meeting, the knowledge seeker’s need is reduced by the provided amount. The knowledge seeker must then move away to work individually for a given amount of time or engage with another knowledge provider.
- **Knowledge Sharing**: Over the course of the project, until stage completion, we allow knowledge seekers to continue gaining knowledge, even when their own need is fulfilled. This behavior maps to real-world activities where teammates may be assuming responsibilities for other members or are otherwise gaining and processing unnecessary knowledge (Salanova et al., 2002).

Timing

- **Time steps**: Steps map to the hours of time that team members may spend collaborating with teammates or working individually on the project.

Stopping Criteria

- **Criterion**: When the cumulative initial knowledge need for all knowledge seekers is exceeded by the cumulative amount of knowledge gained over time. At the time step when the simulation stops, the project stage is considered complete.

Outputs

- **Preliminary Tests (Validation)**: Length of the project in terms of speed and effective meetings.
- **Members Exceeding Their Knowledge Need**: Number of knowledge seekers that went over their need.
- **Maximum Amount of Knowledge Gathered Exceeding Knowledge Need**: The maximum amount any knowledge seeker went over his/her need.
- **Maximum Time Wasted Filling Knowledge Need**: The maximum wasted time that a knowledge seeker spent over their need limit.

<table>
<thead>
<tr>
<th>Table 1. Project Management Simulation Decisions</th>
</tr>
</thead>
</table>

Herein we model independent behaviors over time as members have opportunities to meet and knowledge seekers (i.e., novices) can acquire knowledge from providers (i.e., experts). The need and supply of knowledge are determined based on Poisson distributions where seekers’ needs average 40 knowledge units representing a need across multiple areas, and providers supply average 5 knowledge units representing an expertise in one area. The meeting opportunities occur per a schedule directed by the PM method. As such, the PM method determines when and for how long team members have to interact (i.e., allocated hours for meeting and individual work). Waterfall allows for longer open meetings (16 hours or 2 days) and longer non-meeting times when members are expected to work alone (20 hours) than the Agile method (4 hours of open meeting time followed by 10 hours of individual time). During open meeting times knowledge seekers can contact a knowledge provider and if the provider is free they communicate. The communication media is selected based on set availability; for example, in the rich media scenario, members always meet using rich media (e.g., face-to-face) where 100% of the knowledge given is absorbed, but the time consumed takes 100% of the meeting length. Alternatively, if the mix media scenario is used, each time team members meet they have a 50/50 chance of using rich or lean media. The selection decision is not stochastic, that is, previous use of rich media does not preclude rich media from being selected in the next meeting. The meetings and individual work continue until the culmination of knowledge needed by knowledge seekers is overcome. As such, all teams satisfied exactly or exceeded the team’s cumulative knowledge need (i.e., created overage) by the time the project stage completed. This threshold indicates that the needed knowledge was sufficiently distributed among team members, although any particular knowledge seeker may end up below or above their initial need.

RESULTS

The methods described above were used to simulate six datasets, based on the experimental design, each with 1000 runs. The data collected was then analyzed using MANOVA with Student-Neuman-Keuls (SNK) rankings. The first analysis was conducted to assess the model that the generated activities reflect expectations of real-world projects using Waterfall or Agile methods and rich and/or lean media. The second analysis examines output data to inform insights about the RQs.

Preliminary Tests
To demonstrate that our model provides results that could be expected of projects using the PM methods or communication media of interest, we compare the time of project stage completion across conditions. In terms of time, the results in Table 2 show that teams using the Waterfall method, regardless of the communication media, take longer than teams using an Agile method, which is expected based on our parameters and what we know of Waterfall and Agile teams method (Cusumano et al., 1995) and therefore serves as an accuracy check for our model. The fastest teams were those using Agile in a rich media scenario (speed $\mu_{A,R} = 254.29$). The slowest teams were those using Waterfall in a lean media scenario that took almost two and a half times longer than the fast group (speed $\mu_{W,L} = 636.21$). The results also show the number of meetings was strongly driven by the media scenario with lean media requiring the most meetings (meetings $\mu_{W,L} = 256.51$ and meetings $\mu_{A,L} = 256.44$). This also makes sense as lean media provides less information and therefore requires more meetings compared to rich media (Boyle et al., 1994). Thus, we proceed to utilize this model to uncover the interactive effects of organizational inputs on inefficiencies from members seeking knowledge.

<table>
<thead>
<tr>
<th></th>
<th>Speed (hours)</th>
<th>SNK</th>
<th>Meetings (meeting-count)</th>
<th>SNK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Waterfall - Rich Media</td>
<td>497.89</td>
<td>e</td>
<td>128.81</td>
<td>a</td>
</tr>
<tr>
<td>Waterfall - Lean Media</td>
<td>636.21</td>
<td>f</td>
<td>256.51</td>
<td>c</td>
</tr>
<tr>
<td>Waterfall - Mix Media</td>
<td>448.00</td>
<td>d</td>
<td>171.30</td>
<td>b</td>
</tr>
<tr>
<td>Agile - Rich Media</td>
<td>254.29</td>
<td>a</td>
<td>128.90</td>
<td>a</td>
</tr>
<tr>
<td>Agile - Lean Media</td>
<td>422.46</td>
<td>c</td>
<td>256.44</td>
<td>c</td>
</tr>
<tr>
<td>Agile - Mix Media</td>
<td>286.96</td>
<td>b</td>
<td>171.47</td>
<td>b</td>
</tr>
<tr>
<td>F-stat</td>
<td>22708.91</td>
<td>**</td>
<td>28220.60**</td>
<td>**</td>
</tr>
<tr>
<td>Df</td>
<td>5.00</td>
<td></td>
<td>5.00</td>
<td></td>
</tr>
</tbody>
</table>

+ $p < 0.10$, * $p < 0.05$, ** $p < 0.01$

Note. Mean values reported in table. Those means assigned the same letter were not significantly different.

**Table 2. Preliminary Tests**

**Output Tests**

Table 3 presents the MANOVA and SNK results of output comparisons across datasets. Based on the results we provide insights to our RQs.

Our first RQ asked about the number of team members exceeding their knowledge need. The results indicate that teams were almost equal in terms of the number of knowledge seekers over their knowledge need. The test reports a marginally significant difference ($F$-stat $= 2.09, p = 0.06$) between Agile teams in the rich media scenario (members $\mu_{A,R} = 7.75$) and Agile teams in the lean media scenario (members $\mu_{A,L} = 7.91$).

When looking at the maximum amount of knowledge overage by knowledge seekers that exceed their own need (RQ(2)), we found many differences across PM method and media conditions. The members with the most knowledge beyond their need were on Agile teams in the rich media scenario (average $\mu_{A,R} = 24.27$) and Agile teams in the mix media scenario (average $\mu_{A,M} = 24.10$). These members gained almost one and a half times more knowledge than members with the lowest amount of overage on Waterfall teams in the lean media scenario (average $\mu_{W,L} = 17.26$). Taken together, these results indicate that overage is reduced by utilizing a Waterfall rather than an...
Agile method, and additionally through the use of lean media.

Finally we assessed the maximum time wasted by team members that had completed gathering knowledge to meet their initial needs (RQ3). The results indicate that the highest waste was by Agile teams in the lean media scenario (waste $\mu_{A,L} = 7.67$ hours). This amount of time is 177% more than by teams with the least waste, Waterfall teams in the rich media condition (waste $\mu_{W,R} = 4.32$ hours). While the amount of time represents only a fraction of the project duration, and that fraction is less on Waterfall projects because they tend to be longer projects than Agile projects, the important trend is that the wasted time by knowledge seekers is reduced when rich media is used.

DISCUSSION

The pattern of results does provide support that our model represents the way PM methods and communication media might be expected to unfold in two ways. First, in terms of speed, the results indicate that Agile teams finish the project faster than Waterfall teams. Since the tenet of Agile is to use short iterations to quickly iterate and produce a viable outcome (Henderson-Sellers et al., 2005), then the results meet expectations. Second, in terms of meetings, the use of rich media produces the fewest meetings, followed by mixed media, and lastly lean media. Given that rich media allows for the most knowledge to be shared in any one meeting (Daft et al., 1986), it makes sense to see the progression of meeting counts increase in the order shown. In summary, the pattern of results for organizational inputs suggests that communication activities are unfolding as we anticipate from projects using these PM methods and media. Next we look at how these inputs interactively affect knowledge seeking to provide insights for each RQ.

RQ(1) The Number of Team Members Exceeding Their Knowledge Need

Results indicate that teams in all conditions were almost equal in terms of the number of knowledge seekers exceeding their knowledge need, although slightly more knowledge seekers exceeded their knowledge need on Agile teams in the lean media scenario compared to Agile teams in the rich media scenario. Because members receive more knowledge with rich media (Chidambaram et al., 1993), perhaps members in the rich media scenario pace their knowledge gathering better than those using lean media. As members gain smaller amounts of knowledge using lean media, they may be gathering knowledge in small increments and this may lead more members to go over at the same time. Thus, for Agile teams, we recommend they rely more on rich than lean media to gather their knowledge because rich media allows for fewer members exceeding their knowledge need and better-paced knowledge gathering as they gain more knowledge at a time via rich versus lean media (RQ2). This insight corroborates other research promoting collocation for Agile teams despite organizations’ push for distributed teams to reduce development timelines and leverage cheaper software development resources across the world (Green et al., 2010).

RQ(2) The Maximum Amount of Knowledge by Which a Team Member Exceeds His/Her Knowledge Need

Our results suggest that knowledge overage is reduced by utilizing a Waterfall rather than Agile method, and through the use of lean media. Because lean media provides smaller amounts of information (Boyle et al., 1994), it may prevent excessive knowledge overage because members do not gain large amounts at a time using lean media; thus, lean media can help mitigate knowledge overload as it can be more difficult to limit the amount of knowledge using rich media since you cannot cutoff the knowledge exchange. For example, using email as a lean medium, a member can delete the email without reading or processing the information, but when talking to someone face-to-face, the member cannot stop the communication as easily since it is a live interaction.

Furthermore, while it seems at first glance the results prefer teams use Waterfall methods with lean media, we must recall that Agile teams are purpose-built to respond quickly to change (Henderson-Sellers et al., 2005). Thus, due to Agile’s iterative nature, members may be gaining more knowledge each iteration which compensates for the rate of change these teams incorporate into projects. Agile teams only plan for two-week iterations rather than the entire

<table>
<thead>
<tr>
<th>F-stat</th>
<th>df</th>
<th>$p$-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.09+</td>
<td>5.00</td>
<td>$+ p &lt; 0.10$</td>
</tr>
<tr>
<td>209.57**</td>
<td>5.00</td>
<td>$* p &lt; 0.05$, $** p &lt; 0.01$</td>
</tr>
<tr>
<td>674.11**</td>
<td>5.00</td>
<td>$** p &lt; 0.01$</td>
</tr>
</tbody>
</table>

Note. Mean values reported in table. Those means assigned the same letter were not significantly different.

Table 3. Output Comparison Tests

eProceedings of the 8th International Research Workshop on Information Technology Project Management (IRWITPM)
Milan, Italy, December 14th, 2013 53
project like Waterfall teams, and they incorporate changes into such short-term planning so their knowledge needs will be greater and not necessarily an overage as they may use that additional knowledge when incorporating a later change. As such, Agile teams might consider a balance of rich and lean media to ensure they gain enough knowledge and are prepared for knowledge needs due to incorporating change regularly into their iterations.

RQ(3) The Maximum Amount of Time a Team Member Wastes After Filling His/Her Knowledge Need

The final RQ assesses the maximum time wasted by team members that had completed gathering knowledge to meet their initial needs, with the highest waste from Agile teams in the lean media scenario. We recommend teams use rich media to prevent waste as lean media does not provide as much useful knowledge as rich media. But while the results also suggest using a Waterfall method to prevent waste, we encourage teams to consider the length of overall project duration. Waterfall projects are much longer than Agile projects, so members waste less percentage of time on Waterfall projects than on Agile projects. Agile teams in the long-run waste less time gathering knowledge, freeing those team members to contribute to other value-add tasks. Thus, there is a trade-off between gathering too much extra knowledge which might help Agile teams prepare for change (RQ2) versus wasting time doing nothing. But if a member is doing nothing, perhaps the team can redirect that member to more value-add activities, thereby using time more efficiently.

LIMITATIONS

Our investigation utilized simulation procedures to assess knowledge sharing on projects. The model met expectations, and allowed us to examine 6000 project teams across six different experimental conditions that would have been cost and time prohibitive in other settings. To develop the simulation we relied on published parameters, theorized relationships, and insights from multiple studies to construct our simulation model. As such, we cannot validate our simulation on an independent set of data and settle for a pretest approach (Boudreau, Gefen, and Straub, 2001). The purpose of simulations is to create abstractions of reality (Harrison et al., 2007), and the model produced herein followed procedures (Rand and Rust, 2011) meant to represent a general model of theoretical relationships. As such, we were concerned with face validity, whether the certain constructs in the model produced expected effects (Boudreau et al., 2001), which we demonstrate through our preliminary analysis of project length for different communication media and PM methods. Given our limited validation applied, however, our results provide speculative initial insights about knowledge sharing over time on projects. Future researchers may seek additional validation using empirical data following established guidelines (e.g., Straub, Boudreau, and Gefen, 2004) to make more definitive implications.

CONCLUSION

In summary, our investigation of knowledge sharing provides at least three contributions. First, we consider inefficiencies to provide insights about the dark side of knowledge sharing under different organizational inputs. Second, PM methods like Agile and Waterfall provide the temporal structure for managing team interactions, and using different media types is often a mechanism for knowledge management directed in the project communication plan (Schwalbe, 2010). Through our study we suggest relative differences in PM methods and media use for diffusing the effects of knowledge sharing inefficiencies. Third, this paper describes the development of an agent-based simulation model of project management. As such, we provide a methodological contribution to the study of project team environments, knowledge sharing, and process outcomes. The utility of such an approach is to guide future research about complex parameters like knowledge sharing. Thus, this paper is the foundation for the authors to continue more complex study via simulation and ultimately with actual project teams in industry.

ACKNOWLEDGMENTS

We thank Randika Ekanayake for helping with initial set-up of the simulation and data collection.
REFERENCES


Leadership, Regulatory Focus and Project Performance

Chia-Yu Lai
Department of Information Management, National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C., chiayu06@gmail.com

Jack Shih-Chieh Hsu
Department of Information Management, National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C., jackshsu@gmail.com

ABSTRACT

Leadership is one critical factor of effective teamwork, such as information system (IS) projects. The mission of project leaders is to motivate followers and create an effective working environment that allows project teams to effectively meet the predefined goals. However, based on regulatory focus theory, a team may strive to the optional situation (promotion focus) or try to avoid not meeting the minimum requirements (prevention). The aim of this paper is to explore the effect of leadership styles (transformational and transactional) on the regulatory focus of one team (promotion and prevention), and investigate the relationship between regulatory focus and project team performance. Based on data collected from 154 IS professionals, we found that transformational leadership is associated with promotion focus and transactional leadership leads to prevention focus. Furthermore, while promotion focus orientated teams can perform effectively, prevention focus orientated teams are less efficient. Implications toward academia and practitioners are provided.

Keywords

Project management, leadership, regulatory focus theory, project performance

INTRODUCTION

Despite information technology being a crucial business strategy today, the high failure rates of information systems (IS) projects keep challenging contemporary organizations. According to Standish group, more than half IS projects exceeding budget, are behind schedule, or are unable to meet user’s requirements (The Standish Group, 2012). Researchers have identified various factors, including technical and managerial issues, as major courses of such high failure rate (Carson, Tesluk, & Marrone, 2007; Scott & Vessey, 2002). Among which, leadership should play a role since most IS projects are accomplished by teams instead of individuals. A lack of leadership or employing an ineffective leadership style is one top impediment for IS project success (Sumner, 2000). However, in spite of its importance, project management literatures largely ignore the importance of project managers and their leadership style while attempting to understand the success factors or project (Turner & Müller, 2005).

From a managerial perspective, the mission of an ISD project team is to reach a set of goals. The extent to which project teams can accomplish predefined goals effectively and efficiently is one critical index of team performance. While the importance of goal reaching has been identified, past studies largely focus on the ways to better reach those predefined goals (DeShon, Kozlowski, Schmidt, Milner, & Wiechmann, 2004; Locke & Latham, 2006), but ignore that goal reaching may be achieved by how project teams are oriented toward the goal. According to regulatory focus theory (Higgins, 1997, 1998), goals can be classified into “ideal” and “ought to be” two types and actors are driven to lean on achieving ideal situation (promotion focus) or avoiding not meeting ought to be status (prevention focus). In an ISD context, promotion focus oriented teams tend to strive to ideal goals (e.g. high quality system or accomplish project as early as possible). On the other hand, prevention focus oriented teams tend to pay more attention to preventing performance from being worse than the minimum expectation (e.g. try not missing any main functions or not exceeding budget and deadline).

The regulatory focus orientation of one team reflects what the team attempts to achieve and serves as a motivation principle to guide behaviors (Higgins, 1997, 1998). It is then reasonable to suspect that project performance may be determined by the regulatory focus orientation of project teams. Furthermore, given that one major function of leaderships is to motivate a team to move toward a set of predefined goals, it is also reasonable to suspect that the employment of specific leadership styles may lead project teams to be promotion or prevention focus (Neubert et al., 2008). However a lack of systematic study leaves these questions unanswered.
The purpose of this study is then twofold. First, we intend to understand the correlation between regulatory focus and project performance. Exploring the relationship helps pinpoint the importance of regulatory focus in ISD context. Second, we attempt to explore the impact of leadership on the forming of regulatory focus of ISD project team. Understanding whether different leadership styles lead to different regulatory focus allows us to further understand the function of leadership in a project setting. The remainder of the paper is organized as follows: first, we review relevant literatures about leadership and regulatory focus theory. The theoretical gaps that our study addresses are also identified. Next, a theoretical model linking leadership styles, regulatory focus, and project performance is then constructed. Meanwhile, hypotheses were also built. In the third section, we describe how required data were collected. In the fourth section, data analysis and hypotheses test were provided. Finally, we conclude with implications of our work for both academic and practical areas.

LITERATURE REVIEW AND HYPOTHESES DEVELOPMENT

In this section, we first reviewed transformational and transactional leadership and their roles in information systems projects. Next, we introduce the regulatory focus theory. Finally, we build the link between leadership styles leading to different regulator focus and hypothesize the relationship between regulatory focus and project performance.

Transactional and Transformational Leadership

In organization, leadership is critical in shaping workers’ perceptions toward organization, behaviors associated with organizational change, acceptance of innovations, and motivation to achieve goals. According to Yukl and Heaton (2002) “leadership reflects the assumption that it contains a social influence process where intentional influence is employed by one person over other people to establish the activities and relationships in a group or organizations”. Leadership has received considerable attention in management over decades and is known to be a critical success factor for project success (Thite, 2000; Turner & Müller, 2005; Yang, Huang, & Wu, 2011).

Leadership can be classified into transformational and transactional two types (B. M. Bass, 1985). The major difference between two leadership styles is that transactional leadership is based on exchanges between leader and follower and transformational leadership goes beyond the cost-benefit exchange of transactional leadership by motivating and inspiring followers to perform beyond expectations. While transformational style leaders tend to inspire and motivate followers through a vision, transactional style leaders focus more on reinforcement and exchanges, accompanied with monitoring. Specifically, transformational leadership goes beyond just the simple exchange of rewards for compliance motivating and inspiring followers to perform beyond expectation through visions and charisma leadership. Transformational leaders create new visions, seek new ways of working, search for opportunities in the face of risk, prefer effective answers to efficient answers, and are less likely to stay the status quo (Avolio, Bass, & Jung, 1999; B. M. Bass, 1985). Transformational leaders also provide inspiration and intellectual stimulation to motivate followers by creating high expectation and achievements, and challenge followers with new thinking and ideas. Prior research and meta-analysis have indicated that transformational and transactional leadership have different effects on individual performance (Bono & Judge, 2004; Eagly, Johannesen-Schmidt, & Van Engen, 2003; Howell & Avolio, 1993). Empirical evidence has revealed that transformational leadership predicts positive performance outcomes, whereas the impact of transactional leadership has obtained mixed results, which have reported positive relationships and negative relationships (B. M. Bass, Avolio, Jung, & Berson, 2003; Hater & Bass, 1988; Howell & Avolio, 1993; Howell & Hall-Merenda, 1999).

On the other hand, transactional leadership (Turner & Müller, 2005) aims at monitoring and controlling followers through rational or economic means. Transactional leader who can also interact with followers by exerting management-by-exception to focus on mistakes and ineffective performance may punish followers or intervene after standards have not been met and have preference for risk avoidance. According to Hater and Bass (1988), the focus of passive management-by-exception is that leaders keep passive until problems occur and need to correct and then intervene with criticism or disapproval. In an active way of management-by-exception, the leaders actively monitor followers’ performances to predict mistakes or problems before they become serious. In either case, negative feedback, punishment, and discipline are the possible outcomes (B. M. Bass & Avolio, 1993; B. M. Bass et al., 2003). Based on the leader’s level of engagement with followers and activity level, B. M. Bass (1985) further characterized two types of transactional leadership: contingent reward and management-by-exception. Transactional contingent reward leadership clarifies expectations and negotiates reward agreement when goals are achieved. Rewards and recognition are provided when the followers successfully accomplish their tasks and assignments (Avolio et al., 1999; B. M. Bass, 1985). However, even though leadership has been explored and emphasized in
management, it drew only few little attention in project management literature (Avolio et al., 1999; Kendra & Taplin, 2004; Müller & Turner, 2007; Turner & Müller, 2005). This highlights a need to explore the relationship between relationship and teamwork effectiveness in project management area.

**Regulatory Focus Theory**

Regulatory focus is a social cognitive explanation of motivation that includes deliberated consideration of needs, goals and consequences. Regulatory focus theory was developed through extending the “pleasure approaching and pain avoiding” concept in psychology (Higgins, 1997, 1998, 2000). This theory proposes that self-regulation exerted differently when serving primarily different needs, such as the needs of nurturance and security. Two types of regulatory focus are then identified (Brockner & Higgins, 2001). First, promotion focus is more concerned with accomplishments and advancement with presence of positive outcomes and second, prevention focus is concerned with safety and responsibility with presence of negative outcomes.

Each regulatory focus has different results for cognition, decision making, and emotions, as well as for individuals’ behavior and performance. The purpose of the promotion focus is to pursue advancement and change and to explore the opportunities of creativity and novelty (Kark & Van Dijk, 2007). Therefore, individuals with promotion focus orientation are more likely to use approaches as a goal achievement strategy, are more sensitive to rewards, tend to be more creative in problem-solving methods, and reveal more willingness to undertake risks. On the other hand, the purpose of the prevention focus is to ensure one’s safety and security, to retain routines, and to stay the status quo. Thus, people with prevention focus orientation are more likely to use avoidance as a goal achievement strategy, are more sensitive to punishment, and are more anxious to take risks (Brockner & Higgins, 2001; Higgins, 1997, 1998).

Recent studies on prevention-promotion effects suggest that conditionally operating the regulatory focus orientations are significant predictors of attitudes and behaviors (Cesario, Grant, & Higgins, 2004; Cunningham, Raye, & Johnson, 2005; Lee & Aaker, 2004). While past work on regulatory focus theory has been conducted at individual level of analysis, recent team-level studies have embraced this prevention-promotion focus concept (Florack & Hartmann, 2007; Levine, Higgins, & Choi, 2000; Seibt & Förster, 2004). For example, collective regulatory focus can be part of the identity of a group (Faddegon, Scheepers, & Ellemers, 2008); people who work together overtime developed shared regulatory focus on problem solving strategy(Levine et al., 2000); and regulatory generates effect in a group after a period of time (Florack & Hartmann, 2007). In this study, we apply regulatory focus theory in the team level and argue that the shared regulatory focus orientation within the team is a result of leadership styles exercised by the leaders. Based on regulatory focus theory, in the context of IS projects, individuals may attempt to achieve project goals using either a promotion focus which maximizes positive outcomes (e.g., project success, accomplishment), or a prevention focus which minimizes negative outcomes (e.g., escalation, over-budget). We argue that project leaders have the potential to shape team members’ regulatory focus, as suggested by Brockner and Higgins (2001) and Kark and Van Dijk (2007).

**Hypotheses development**

A research model was developed, as shown in Figure 1. We argue that each leadership style leads to specific regulatory focus of the team. In addition, project performance is a function of regulator focus style of the project team. In the following hypotheses were developed.

![Research Model](image-url)

**Figure 1 — Research Model**

*eProceedings of the 8th International Research Workshop on Information Technology Project Management (IRWITPM)*

*Milan, Italy, December 14th, 2013*
From leadership to regulatory focus

Due to project leaders’ potency to administer rewards or punishment, project team members are motivated to comply with leaders’ behavior and communication in order to receive rewards or avoid punishment, making leaders even more influential as role models (Treviño, Brown, & Hartman, 2003). Prior research also has proved that leaders can influence the regulatory focus of followers (Neubert, Kacmar, Carlson, Chonko, & Roberts, 2008; Neubert, Wu, & Roberts, 2013). Through self-regulation, project team members can adapt and align their perception and behaviors with the expectations of leaders (Brockner & Higgins, 2001; Higgins, 2000).

The personalities of transactional leaders tend to be more practical and less idealistic (Bono & Judge, 2004). Transactional leaders reward subordinates when their performance is better than expected and, in contrast, punish subordinates when their performance is worse than expected. In addition, transactional leaders pay attention to exceptional conditions, most likely when performance is lower than expected or when members cannot fulfill their responsibilities. This is reasonable because employees are expected to have sufficient performance and, therefore, transactional leaders in general highlight the minimum accepted level performance that each member or the whole team needs to achieve. As an outcome, project team members are affected by leaders and tend to focus on the responsibilities and duties of the project. Therefore, project teams with transactional leaders tend to be more prevention focus and focus on their responsibilities or obligations that they ought to do.

\[ H1: \text{Transactional leadership will be positively associated with the prevention focus.} \]

On the other hand, transformational leaders transform the self-concept of their team members, affective influence and ideal appeal to or reveal the importance of the visions, achievement and involvement (B. M. Bass et al., 2003; Kark & Van Dijk, 2007). Transformational leaders inspire the whole team and make them believe that the ideal situation is achievable and they are willing to devote themselves on achieving the goals. Transformational leaders may affect their followers to be more promotion focus because they encourage followers to attain their ideal states. We then argue that project teams with transformational leaders tend to be more promotion focus and propose that

\[ H2: \text{Transformational leadership will be positively associated with the promotion focus.} \]

From regulatory focus to project performance

According to regulatory focus theory (Higgins, 1997, 1998), the self-regulatory focus is a fundamental motivational principle influencing multiple human activities and behaviors. Individuals with a prevention focus tend to be more cautious in their behavior and motivated by the perception of obligation or duty, whereas individuals with a promotion focus are more likely to be inspired in their behavior and motivated by the perception of achievement (Kark & Van Dijk, 2007; Scholer & Higgins, 2008). Each regulatory focus also contributes to difference consequences for individual’s behaviors and performance (Higgins, 1998).

Speed and accuracy are two important indicators of work performance, as individuals who follow rules and procedures accurately are more likely to work safely and effectively, whereas those who can complete large amount of work more quickly are likely to be more efficient at work. The previous studies have empirically examined that individuals with promotion regulatory focus could enhance their achievement by accelerating the speed to accomplish the work, while individuals with prevention regulatory focus might tend to work in a safe manner by emphasizing on accuracy (Forster, Higgins, & Bianco, 2003; Kark & Van Dijk, 2007; C. Wallace & Chen, 2006). Thus, regulatory focus theory provides clear explanation for why and how individual differ in contexts and situations directed at efficient and effective outcomes at work. Based on theory and relevant findings reported in the IS and project literature, we expect that promotion and prevention focus have effects of effectiveness and efficiency on project outcomes.

The prior project management literature defined project success as achieving project goals, within budget and schedule, and meeting user’s requirements (Lewis, 1999). IS project performance can be conceptualized in terms of project effectiveness and efficiency (L. Wallace, Keil, & Rai, 2004). Project effectiveness refers to the extent to which a project accomplishes all tasks and satisfactorily fulfills user’s needs. Project efficiency refers to the extent to which a project is delivered on schedule and within budget. It is imperative to study both aspects of project performance, because there is a potential conflict between the efficiency of the project and its quality (Nidumolu, 1995).
Efficiency and effectiveness are two distinct measures of project performance (Henderson & Lee, 1992). Project effectiveness shows well the project is accomplished. Hence, we measure the project effectiveness of project performance on amount of completed work, quality of work produced, and effectiveness in meeting project goals. On the other hand, project efficiency refers to the extent to which a project is delivered on schedule and within budget (L. Wallace et al., 2004).

With the rapid changes of today’s information technology of strategic business, user requirements have become increasingly difficult to predict and control (Gorschek et al., 2007). Teams with a promotion focus are motivated to achieve the ideal project goal and seek chances for growth or advancement. They also tend to recall information and memory related to the rewards or positive outcomes in terms of benefit or success (Higgins, Shah, & Friedman, 1997). When teams lean on promotion-focused, project team members are inspired to perform beyond standard expectations with a commitment through a vision and competence influenced by the leader (Brockner & Higgins, 2001; Higgins, 1997; Yukl & Heaton, 2002). Project team members are motivated and have stronger self-efficacy to pursue more challenging goals (Waldman, Ramirez, House, & Puranam, 2001). On the effectiveness side, in addition to solely accomplishing the predefined tasks, project teams tend to move forward and try to develop a high standard system. On the efficiency side, rather than being not delayed or over budget, they also accomplish the predefined tasks earlier than schedule and within budget. Thus, we expect that project teams tend to be more effective and efficient when they are more regulatory focus oriented.

In organization, quantity and quality are two important indicators of work performance, as individuals who pay attention on the final outcome are more likely to work effectively, whereas those who focus on completing large amount of work more quickly are likely to be more efficient at work. The previous studies have empirically illustrated that people with promotion regulatory focus may enhance their achievement by accelerating the speed to accomplish the work or increasing the quality of outcome. On the other hand, individuals with prevention regulatory focus might tend to work in a safe manner by not meeting the minimum requirements (Förster et al., 2003; Kark & Van Dijk, 2007; C. Wallace & Chen, 2006). As an outcome, prevention focus individuals care more about whether they can achieve the goal without violating any work rule but not how efficient they can be.”

**H3:** The promotion focus will have a significant positive effect on effectiveness of project team performance.

**H4:** The promotion focus will have a significant positive effect on efficiency of project team performance.

On the other hand, we predict that prevention focus will be negatively associated with project performance. When teams are prevention focused, members in the teams are responsive to seeking to attain the goals or standards associated with the obligation and duties and strategically to try to avoid behaviors that mismatch a goal or standard (Brockner & Higgins, 2001). The behaviors with prevention-focus indicated when the project team members face threatening or aversive conditions that arouse the feelings of fear and anxiety to avoid or minimize negative outcomes (Scholer & Higgins, 2008). Prevention focus also is considered as more conservative, risk-averse and less creativity and unwilling to change(Friedman & Förster, 2001). Therefore, project teams with strong prevention focus tend to avoid any changes in the system development process. For example, they tend to stick on the standard procedures of selected method. If the selected method is not appropriate or not allow team members to react to external changes effectively, negative outcome can then be expected. One critical feature of software development is that users’ requirements are dynamic and change frequently (Harker, Eason, & Dobson, 1993). Since teams with prevention focus are more attentive on not doing things wrong but not how to do things correctly, we expect that efficiency will be low. As the requirement changes, the team needs to redevelop some functions because those functions are deviated from users’ needs. Since prevention focus teams still attempt to achieve the minimum goal, they tend to redevelop those inadequate functions. As an outcome, extra time and costs are unavoidable. It is therefore reasonable to believe that project team members with a prevention focus may reduce the efficiency of project performance. Thus, based on the theory and several consistent findings in the literature, the following hypotheses are suggested:

**H5:** The prevention focus will have a significant negative effect on efficiency of project team performance.
RESEARCH METHODOLOGY

Data Collection and Sample

Based on our research model, a survey approach was conducted to collect required data. We focused on leading Taiwan companies because they often have their own IS department for IS system development and maintenance. Online questionnaires were mailed to the IS professionals, who were invited to voluntarily and anonymously file our survey based on their most recent experience in an IS project. All respondents were assured that their responses would be kept confidential and used solely for academic purposes. The questionnaire was sent by e-mail. As electronic surveys allow the transmission of more information, they support a better interaction between the researchers and the respondents, and they contribute to a better quality of information, to a faster response cycle and to a reduction in research costs (Klassen & Jacobs, 2001). A total of 1000 questionnaires were sent and 154 were returned, which results in a 15.4 % response rate. Table 1 shows the demographic information of the respondents.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Categories</th>
<th>#</th>
<th>%</th>
<th>Measure</th>
<th>Categories</th>
<th>#</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gender</td>
<td>Male</td>
<td>89</td>
<td>58%</td>
<td>Educational</td>
<td>&lt; college</td>
<td>5</td>
<td>3%</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>60</td>
<td>39%</td>
<td>background</td>
<td>Bachelor</td>
<td>73</td>
<td>47%</td>
</tr>
<tr>
<td></td>
<td>Missing</td>
<td>5</td>
<td>3%</td>
<td></td>
<td>Master</td>
<td>70</td>
<td>46%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Missing</td>
<td>6</td>
<td>4%</td>
</tr>
<tr>
<td>Age</td>
<td>21-30</td>
<td>22</td>
<td>14%</td>
<td>Position</td>
<td>SA</td>
<td>48</td>
<td>31%</td>
</tr>
<tr>
<td></td>
<td>31-40</td>
<td>102</td>
<td>66%</td>
<td></td>
<td>Senior member</td>
<td>61</td>
<td>40%</td>
</tr>
<tr>
<td></td>
<td>More than 40</td>
<td>23</td>
<td>16%</td>
<td></td>
<td>Other specialties</td>
<td>40</td>
<td>23%</td>
</tr>
<tr>
<td></td>
<td>Missing</td>
<td>5</td>
<td>3%</td>
<td></td>
<td>Missing</td>
<td>5</td>
<td>3%</td>
</tr>
<tr>
<td>Industry type</td>
<td>Manufacturing</td>
<td>86</td>
<td>56%</td>
<td>Size of</td>
<td>1-5</td>
<td>13</td>
<td>8%</td>
</tr>
<tr>
<td></td>
<td>Telecommunication</td>
<td>2</td>
<td>1%</td>
<td>information</td>
<td>6-10</td>
<td>18</td>
<td>12%</td>
</tr>
<tr>
<td></td>
<td>Technology</td>
<td>9</td>
<td>6%</td>
<td>department</td>
<td>11-15</td>
<td>17</td>
<td>11%</td>
</tr>
<tr>
<td></td>
<td>Governance</td>
<td>23</td>
<td>15%</td>
<td></td>
<td>16-20</td>
<td>33</td>
<td>22%</td>
</tr>
<tr>
<td></td>
<td>Other</td>
<td>27</td>
<td>17%</td>
<td></td>
<td>21-40</td>
<td>19</td>
<td>12%</td>
</tr>
<tr>
<td></td>
<td>Missing</td>
<td>7</td>
<td>5%</td>
<td></td>
<td>More than 40</td>
<td>54</td>
<td>35%</td>
</tr>
</tbody>
</table>

Table 1. Demographic information

Construct and measurement

Transformational leadership was measured with four items selected from Bass and Avolo’s (1990) Multifactor Leadership Questionnaire (MLQ) Form 5X (B. M. A. Bass, B. J., 1990; Vera & Crossan, 2004). These four items are corresponding to four scales of transformational leadership: charisma, inspirational motivation, intellectual stimulation and individualized consideration. Although past studies treat transformational leadership as a second order reflective construct (B. M. Bass, 1985; B. M. Bass & Avolio, 1993), we select one item from each first order component to reduce the effort that respondent needs to enter on answering the question. Transactional leadership was also measured with four items select from MLQ. Two items focus on contingent reward, which refers to articulating explicit agreements with expectations from project members and how they will be rewarded for their efforts and commitment (B. M. Bass & Avolio, 1993). Another two items were focus on management-by-exception, which refers to monitoring project members’ performance and taking corrective action as needed (Barling, Weber, & Kelloway, 1996).

Both promotion and prevention focus were measured with items selected from the Regulatory Focus Questionnaire (RFQ) developed by Higgins et al. (2001). All items were wording to reflect the team level nature of this study. The final instrument contains 14 questions—7 for promotion focus and 7 for prevention focus (Higgins et al., 2001; Lockwood, Jordan, & Kunda, 2002).

Project performance represents the degree to which a project achieves targeted goals effectively and efficiently. The measure for project effectiveness and project efficiency were adapted from Rai and Al-Hindi (2000). Project effectiveness refers to the extent to which a project accomplishes all project tasks and satisfactorily fulfills user’s needs. Project efficiency refers to the extent to which a project is delivered on schedule and within budget (L. Wallace et al., 2004). Except for efficiency that is measured with the percentage of delay, all items were rated using a 5-point Likert-type scale ranging from strongly disagree (1) to strongly agree (5).

Lai & Hsu

Leadership, Regulatory Focus and Project Performance

eProceedings of the 8th International Research Workshop on Information Technology Project Management (IRWITPM)

Milan, Italy, December 14th, 2013
Validity and reliability information are provided in Table 2 and 3. Reliability is assured since both composite reliability and Crobach’s alpha are high and no indicator has loading value lower than 0.5. Convergent validity is assured with acceptable loading values and averaged variance extracted (AVE). Discriminant validity is assured since correlations of any paired indexes are all lower than 0.9 and are smaller than the square root of AVE. In order to exclude the potential variance resulted from common method used, Harman’s single factor analysis was performed. Common method should not be an issue since more than two factors are extracted. Furthermore, we also compared the first and last quarter sample. Insignificant results indicate that non-response bias may not be an issue in our study.

<table>
<thead>
<tr>
<th>Construct</th>
<th>Measure</th>
<th>Loading</th>
</tr>
</thead>
</table>
| Transformational Leadership | α: 0.89  
| CR: 0.92  
| AVE: 0.75           | Encourages us to express our ideas and opinions.                        | 0.826   |
|                         | Inspires loyalty to the organization.                                   | 0.825   |
|                         | Enables us to think about old problems in new ways.                    | 0.897   |
|                         | We can count on him/her to express his/her appreciation when we do a good job. | 0.858   |
| Transactional leadership | α: 0.70  
| CR: 0.80  
| AVE: 0.51           | There is a close agreement between what we are expected to put into the group effort and what we can get out of it. | 0.502   |
|                         | Whenever we feel like it, we can negotiate with him/her about what we can get from what we accomplish. | 0.654   |
|                         | It is all right if we take initiatives but he/she does not encourage us to do so. | 0.834   |
|                         | Only tells me what we have to know to do my job.                        | 0.806   |
| Prevention focus        | α: 0.89  
| CR: 0.92  
| AVE: 0.61           | We frequently think about the person we are afraid of and we might become in the future. | 0.806   |
|                         | We often worry that we will fail to accomplish our project goal.        | 0.833   |
|                         | We often image ourselves experiencing bad things that we fear might happen to us. | 0.822   |
|                         | We frequently think how we can prevent failures in our project.         | 0.847   |
|                         | We see ourselves as someone who is primarily striving to become the selves “ought” to be—to fulfill our duties, responsibilities and obligations. | 0.826   |
|                         | In general, we focus on preventing negative events in our project.       | 0.602   |
|                         | We are anxious that we will fall short of our responsibilities and obligations. | 0.702   |
| Promotion focus         | α: 0.92  
| CR: 0.94  
| AVE: 0.68           | In general, we focus on achieving positive outcomes in our project.     | 0.84    |
|                         | We typically focus on the success we hope to achieve in the future.     | 0.843   |
|                         | We frequently image how we will achieve our hopes and aspirations.      | 0.887   |
|                         | We often image ourselves experiencing good things that we image might happen to us. | 0.666   |
|                         | Our major goal in project right now is to achieve our project ambitions. | 0.843   |
|                         | We see ourselves as someone who is primarily striving to reach our “ideal self”- to fulfill our hopes, wishes and aspirations. | 0.885   |
|                         | Overall, we are more oriented toward achieving success than preventing failure. | 0.777   |
effectiveness
\( \alpha: 0.089 \)
CR: 0.93
AVE: 0.82

<table>
<thead>
<tr>
<th>Ability to meet project goals.</th>
<th>0.927</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected amount of work completed.</td>
<td>0.915</td>
</tr>
<tr>
<td>High quality of work completed.</td>
<td>0.874</td>
</tr>
</tbody>
</table>

efficiency
\( \alpha: 0.77 \)
CR: 0.86
AVE: 0.68

<table>
<thead>
<tr>
<th>The extent of project delay.</th>
<th>0.787</th>
</tr>
</thead>
<tbody>
<tr>
<td>The extent of cost overruns.</td>
<td>0.856</td>
</tr>
<tr>
<td>The extent of incomplete project.</td>
<td>0.823</td>
</tr>
</tbody>
</table>

"Note: All factor loadings are significant; Cronbach's alpha values are all higher than 0.7; Composite reliability values are all higher than 0.6; and averaged variance extracted are all higher than 0.5 (Hair & Black, 2006)"

Table 2. Reliability and validity

<table>
<thead>
<tr>
<th>Construct</th>
<th>Mean</th>
<th>Std. dev.</th>
<th>M3</th>
<th>M4</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Transformational</td>
<td>3.29</td>
<td>0.98</td>
<td>-0.37</td>
<td>-0.77</td>
<td>0.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Transactional</td>
<td>3.05</td>
<td>0.83</td>
<td>0.48</td>
<td>0.28</td>
<td>-0.38</td>
<td>0.91</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Promotion</td>
<td>3.64</td>
<td>0.83</td>
<td>-0.32</td>
<td>-0.46</td>
<td>0.34</td>
<td>-0.07</td>
<td>0.78</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Prevention</td>
<td>2.94</td>
<td>0.93</td>
<td>0.43</td>
<td>-0.41</td>
<td>-0.26</td>
<td>0.74</td>
<td>-0.07</td>
<td>0.82</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Effectiveness</td>
<td>3.80</td>
<td>0.87</td>
<td>-0.63</td>
<td>0.53</td>
<td>0.07</td>
<td>0.17</td>
<td>0.62</td>
<td>0.18</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>6. Efficiency</td>
<td>3.27</td>
<td>0.67</td>
<td>1.49</td>
<td>1.04</td>
<td>-0.16</td>
<td>0.60</td>
<td>0.08</td>
<td>0.71</td>
<td>0.32</td>
<td>0.87</td>
</tr>
</tbody>
</table>

M3: Skewness
M4: Kurtosis
The diagonal line of correlation matrix represents the square root of AVE

Table 3. Descriptive analysis and correlation matrix

Analysis and discussion

According to (Hair, Ringle, & Sarstedt, 2011), PLS should be adopted when the purpose of study is theory building instead of confirmation. Since the purpose of our goals is to integrate two different theories in one model and to predict team performance with different leadership styles, we adopt PLS instead of covariance-based SEM. The validation of measurement includes item reliability and discriminant validity analysis. As shown in Figure 2, all proposed hypotheses were found supported. Transformational leadership has strong effect on promotion focus (\( \beta = 0.70; p < 0.01 \)) when the effect from transactional leadership is controlled. Transactional leadership has strong effect on prevention focus (\( \beta = 0.62; p < 0.01 \)) when the effect from transformational leadership is controlled. Therefore, both H1 and H2 are supported. In addition 56.5% variance of promotion focus is explained by transformational leadership while 37% variance of prevention focus is explained by transactional leadership. For the links from regulator focus to project performance, while effectiveness is strongly affected by promotion focus (\( \beta = 0.73; p < 0.01 \)), efficiency is found determined both promotion focus (\( \beta = 0.23; p < 0.01 \)) and prevention focus (\( \beta = -0.34; p < 0.01 \)). Therefore H3, H4, and H5 are supported. In addition, the variance explained by regulatory focus is 54.5% for effectiveness and 17% for efficiency. We conducted three Sobel tests to clarify the mediating effects of regulator focus on the relationship between leaderships and project performance indexes. Two of them are found significant: (1) promotion focus mediates the effect of transformational leadership on project effectiveness, and (2) prevention focus mediate the effect of transactional leadership on project efficiency.

Aligning with our expectation, all hypothesized links were found supported. Project teams tend to have strong promotion focus when managers exercise transformational leadership. On the other hand, project teams tend to be more prevention focus when managers are transactional oriented. Even though each type of leadership generates specific effect, R-square values reveal that transformational leadership has stronger predicting power on promotion focus. For the relationship from promotion focus to project performance, high coefficient and R-square value indicate that promotion focus has very strong effect on effectiveness. However, even though both types of regulatory
focus were found to have effect on efficiency, low R-square value indicates that efficiency of project performance is determined by other variables, in addition to regulatory focus. Furthermore, the negative impact of prevention focus on efficiency reflects one interesting phenomenon. Project teams are less likely to be efficient if their goal is trying not to be inefficient.

![Path analysis diagram]

**Figure 2. Path analysis**

**THEORETICAL IMPLICATIONS**

Taken together, our research model and results contribute to the literature in several important ways. First, our study makes a theoretical contribution to the existing body of knowledge on IS project performance by establishing a connection between leadership styles and regulatory focus orientations. Specifically, the study offers new sights regarding how leadership of IS project managers influence motivation of the whole team, which in turns affects IS project performance. Previous leadership research had already established the impact of leadership on project performance directly (Anantatmula, 2010; Keller, 2006; Munns & Bjeirim, 1996; Thite, 2000), or indirectly through affecting teamwork climate (Eisenbeiss, van Knippenberg, & Boerner, 2008; Rickards & Moger, 2000). Through integrating regulatory focus theory and leadership perspective, we proposed and confirmed that transformational leadership style has positive effect on evoking the promotion focus and transactional leadership results in prevention focus. Furthermore, while promotion focus promotes project performance, prevention focus hinders project teams to efficiency. Thus, we provide another explanation for the effects of leadership on performance.

Second, we contribute to regulatory focus research by successfully applying this theory to team level. Even though some studies point out that regulatory focus may be applied to team level and few studies really study regulatory focus in team level, they are either conceptual or experimental based studies. How the whole team is motivated by leaders hasn’t been explored. We addressed this gap in the project management literature and empirically constructed the links between two distinct leadership styles (e.g., transactional and transformational) and two motivational forces of regulatory focus (e.g., promotion and prevention). This is a pioneering study in testing the regulatory focus of teams. This is critical because the increasing reliance on the team-based structures in the context of IS project. Our study extends earlier work by Kark and Van Dijk (2007), who explain various leader behaviors can arouse a promotion focus or prevention focus at group level.

Third, most previous regulatory focus studies focus on the impact of regulatory on individuals’ behavior, attitude, or affective status. We extend this research stream by showing its impact on performance. Our results show that promotion focus has positive impact on team performance and prevention focus has negative impact on team performance. However, this study also raises more questions than it has answered. For example, it would be interesting to know whether teams with promotion focus orientation can always perform better and teams with prevention focus orientation always perform worse. Future studies may explore possible moderators on the relationship between regulatory focus and performance.
PRACTICAL IMPLICATIONS

Previous research have indicated that transformational leadership substantially influences followers motivation and performance very differently from transactional leadership (Dvir, Eden, Avolio, & Shamir, 2002; Howell & Avolio, 1993). However, in the specific context of IS project, the literature provides little evidence and insights for project management. Specifically, the manager’s leadership role is of critical importance in motivating people and creating an effective working environment in order to meet variety of challenges in today’s rapid-fire information technology innovations. The results of our study highlight the importance for project managers to choose the preferred leadership style. Specifically, our study advises project leaders to employ transformational leadership style to evoke the promotion focus motivations among the project team members. It is particularly essential to focus on developing the promotion focus orientations through a vision to influence team members’ perception and behaviors. Meanwhile, project leaders should avoid being a transactional leader who only focuses on punishment because doing so drive the whole team to be prevention focus. On the other hand, higher managers should pay more attention to the selection of project managers. For those who only exercise transactional leadership should avoid taking charge of projects with strong demand for being efficient.

CONCLUSION

In this paper, we aim to address the theoretical gap between leadership and motivation in the context of information system project management. To achieve this objective, we first draw from two distinct leadership styles - transactional and transformational leadership (Avolio et al., 1999; B. M. Bass, 1985), as well as from the theory of motivational orientation-regulatory focus (Higgins, 1997, 1998) to develop a conceptual framework to advance further studies on the underlying mechanism that enable project leaders to influence the motivation of their project team members and ultimately, their behaviors in terms of project performance.

We empirically tested our proposed model with 154 IS projects. The results confirmed our expectations that transformational leadership leads to promotion focus and transactional leadership leads to prevention focus. While teams with promotion focus orientation tend to perform better, teams with prevention focus are less efficiency. Several implications toward academia and practitioners can be drawn from this study. However, applying the results of this study should pay attention on the following issues. First, this is a cross-sectional study based on regional data. Longitudinal study may be needed to examine whether regulatory focus orientation also affects leaderships. Project leaders may be forced to adopt certain types of leadership when project teams exhibit explicit regulatory focus. In addition, data from other countries may be needed to assure the external validity of the found results. Second, response from one member in each team was used to represent the whole team. However, in order to understand the teamwork process better or assure the validity of collected data, opinions from all members should be obtained.

ACKNOWLEDGMENTS

We thank all authors, committee members, and volunteers for their hard work and contributions to the workshop. The layout of this format was adapted from a workshop document created by Ping Zhang. The content was adapted for the IRWITPM proceedings template. The references cited in this paper are included for illustrative purposes only.

REFERENCES


Lean Software Development: Evaluating Techniques for Parsimonious Feature Selection of Evolving Information Systems Products

Adarsh Kumar Kakar  
Alabama State University  
akakar@alasu.edu

Joanne E. Hale  
Alabama State University  
jhale@cba.ua.edu

David P. Hale  
Alabama State University  
dhale@cba.ua.edu

ABSTRACT

Lean software development is a product development paradigm with focus on creating value for the customer and eliminating waste from all phases of the development life cycle. Applying lean principles, empirical studies were conducted focusing on identifying and assessing methods that parsimoniously select features from a given set of user feature requests. The results of the studies show that the Kano survey method has potential. It demonstrated efficacy in not only identifying the feature subset, from a given set of feature requests, that maximizes value to the users but also in eliminating waste by identifying the subset of features which does not provide significant value to the users when implemented into the software product. The design and results of one study is elaborated in this article. The findings obtained in the study have useful implications for practice and opens up new avenues of research for evolving market-driven software products.

Keywords

Lean software development, feature selection, user satisfaction

INTRODUCTION

Although the lean concept has its origins in manufacturing, specifically the Toyota production system, the principles of Lean manufacturing can be used as a framework, or a guideline to address issues of software development (Kumar, 2009). Applying lean principles to selecting which features should be implemented in a software product can result in tremendous downstream benefits such as reduced cost and complexity of the product.

Of the seven wastes (Table 1) identified in Lean manufacturing one of the most important is avoiding waste of overproduction in the system which in the context of software development translates to adding extra (wasteful) features into the product (Kumar, 2009; Jailia et al., 2011). According to data reported at the 2000 International Software Engineering Research Network (ISERN) Workshop, individuals working alone used only 12 to 16 percent of Microsoft Word and PowerPoint features, whereas a 10-person group used 26 to 29 percent of these features (Basili and Boehm, 2001). Yet, software products must be incessantly adapted (evolved) to match any changes in the real world by adding new features because most software in regular use in businesses and organizations cannot be completely specified (Lehman and Ramil, 2002). Although these adaptations are an economic necessity they must still conform to the lean principle of not adding anything that does not create value for the customer.

<table>
<thead>
<tr>
<th>Serial Number</th>
<th>Seven Wastes of Manufacturing</th>
<th>Seven Wastes of Lean Software Development</th>
</tr>
</thead>
</table>
Table 1. The Seven Wastes (adapted from Kumar (2009))

<table>
<thead>
<tr>
<th></th>
<th>Overproduction</th>
<th>Extra Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Inventory</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Extra Processing Steps</td>
<td>Extra Steps</td>
</tr>
<tr>
<td>4</td>
<td>Motion</td>
<td>Finding Information</td>
</tr>
<tr>
<td>5</td>
<td>Defects</td>
<td>Bugs not caught by tests</td>
</tr>
<tr>
<td>6</td>
<td>Waiting</td>
<td>Waiting for decisions including from customers</td>
</tr>
<tr>
<td>7</td>
<td>Transportation</td>
<td>Handoffs</td>
</tr>
</tbody>
</table>

Adding features that do not add value to the product has adverse implications for both the user and producer of the software product as shown in Table 2.

Table 2. Adding Non-Valued Product Features

<table>
<thead>
<tr>
<th>User</th>
<th>Producer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users have to expend resources in terms of memory and computing power for running additional features that add no value to their work (Basili and Boehm, 2001)</td>
<td>Producers have to utilize their scarce resources in building features that have no positive business outcomes as customers do not fund upgrades of market-driven products (Karlson et al., 2007)</td>
</tr>
<tr>
<td>Overloading the product with features causes “feature fatigue” i.e. the more features a product boasts, the harder it is to use (Thompson, Hamilton and Rust, 2005)</td>
<td>Building new features makes the product complex and more difficult to maintain (Mens et al., 2005)</td>
</tr>
<tr>
<td>May degrade quality and make products unreliable (Mens et al., 2005)</td>
<td>Increases time-to-market as even providing features that do not add value to the user requires additional time to implement</td>
</tr>
</tbody>
</table>

However, selection of only those features which add value pose special challenges for market-driven software products compared with in-house development or software developed for single customer. Producers of market-driven software products have to deal with anonymous users, requirements overload due to large number of feature requests, time-to-market pressures and lack of day to day interaction and negotiation with the user base making the traditional requirement engineering techniques impractical to use (Karlsson et al., 2007). Keeping this context in view, this study reviews and identifies promising methods for feature selection from non-software product domains, such as product development and quality literatures, as well as suitable methods from requirement engineering literature for prioritization of software requirements. It then assesses which of these methods demonstrate greater efficacy in identifying features which add value to the software product and features which do not.

LITERATURE REVIEW

Lean Approach to Software Feature Selection
The first lean principle states “Specify value from the standpoint of the end customer” (Jailia et al., 2011). Software products are developed with the goal of satisfying user needs (Alves, 2003) and user satisfaction is a measure of value provided by the software product (Calisir and Calisir, 2004). It is one of the most prevalent measures of software success and use (Delone and McLean, 1992; Zviran and Erlich, 2003). We therefore chose user satisfaction as dependent variable in the study.

Further, as “the set of requirement selected for implementation is a primary determinant of customer satisfaction” (Karlsson and Ryan, 1996), the feature subsets selected by different methods from a given set of user feature requests was used as an independent variable. The efficacy of feature selection methods in identifying features that add value to the user will thus be determined by which method/s provides a feature subset that delivers maximum satisfaction to the user. The efficacy of the methods in identifying waste will be determined by which among the complementary sets (set of user feature requests – features subset that add value to the user) delivers minimum user satisfaction.

**Exploration of Feature Selection Methods**

This section continues by providing reviews of feature selection methods from requirements engineering literature. Further, this section reviews non-software product development and product quality literatures to identify promising methods of selection of software product features for comparison with the traditional requirements engineering techniques.

**Feature Selection Methods from Requirements Engineering Literature.** Several techniques have been used for prioritization of requirements. Table 3 (adapted from Berander and Andrews, 2007) provides often cited examples of requirement engineering techniques including the Grouping methods such as Priority groups method and the non-grouping (ranking) methods such Planning Game method, 100 points method, Priority Groups method, Theory W method, AHP method, Binary Search Tree method and Value-Oriented Prioritization method.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority Groups (Wiegers, 1999)</td>
<td>Requirements are classified into a small number (often 3) of priority categories, such as High (critical), Medium (regular), and Low (nice to have). Individual results may be aggregated by majority, plurality, or consensus.</td>
</tr>
<tr>
<td>Planning Game (Beck, 2001)</td>
<td>The development team sorts the requirements by value, risk, and effort. Based on the relative assessments, the scope of the next release is set.</td>
</tr>
<tr>
<td>100 Points (Leffingwell and Widrig, 2003)</td>
<td>Each stakeholder is given a total of 100 points that can be allocated (or “spent”) on the requirements. Requirement priority is then determined by sorting the requirements by total points spent by all participants.</td>
</tr>
<tr>
<td>Theory W (Boehm and Ross, 1989)</td>
<td>To ensure that every stakeholder wins, each ranks the requirements and notes which are most important and which they would be willing to remove. The stakeholder groups then negotiates the prioritized list.</td>
</tr>
<tr>
<td>AHP (Saaty, 1980)</td>
<td>Built to address multi-criteria decision-making situations, AHP conducts a comprehensive comparison of the value and cost of each requirement pair.</td>
</tr>
<tr>
<td>B-Tree Prioritize (Heger, 2004)</td>
<td>Uses an algorithm for arriving at the priority list of requirements from a given candidate set of requirements by economizing on the number of comparisons.</td>
</tr>
<tr>
<td>Value-Oriented Prioritization (Azar, Smith and Cordes, 2007)</td>
<td>In this method after identifying the core business value categories, company executives rank each value on a relative scale. Thereafter all requirements are identified a weight in each value category and a ranked list of requirements is generated.</td>
</tr>
</tbody>
</table>

Table 3: Methods from Requirements Engineering Literature

**Other Feature Selection Methods.** Table 4 provides a list of techniques from Quality and Product Development Literature. The basis for many of these techniques can be traced to the three factor model (Kano, 1984) with the following definition for the three factors:
**Basic factors:** are prerequisites and must be satisfied first, at least at threshold levels, for the product to be accepted. The customer takes Basic product features for granted, and therefore does not explicitly ask for them. Basic features are critical when they are not met, but users remain Indifferent if they are provided for in the product.

**Performance factors:** are product features that the customer deliberately seeks to fulfill. They are uppermost in her consciousness. Fulfilling these requirements leads to customer satisfaction and not fulfilling them leads to dissatisfaction.

**Excitement factors:** are those product features that the customer did not expect. They surprise the consumer by adding unexpected value to the product thereby delighting her. Not fulfilling excitement requirements do not lead to consumer dissatisfaction.

In addition to the techniques based on the three factor theory we included a determinant-attribute approach (Myers and Alpert, 1968) which has been used for feature selection in industries as diverse as construction materials (Sinclair and Stalling, 1990) to health care systems (Lim and Zallocco, 1988). As opposed to in-house developed and software products developed for single customers, which often have restricted product choices, software product evolution exists in an open environment, in which marketplace alternatives exist. As shown in Table 3, Myers and Alpert (1968) present a means to incorporate marketplace issues such as alternative products into the Dual Questioning Technique.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Penalty-Reward Contrast Analysis</td>
<td>Classifies each product feature requirement into Basic, Performance and Excitement categories by analyzing the impact of high and low feature level satisfaction on overall product satisfaction using regression analysis with two set of binary dummy variables for each product feature.</td>
</tr>
<tr>
<td>Importance Grid</td>
<td>Classifies each product feature requirement in Basic, Performance and Excitement categories. Users explicitly express preferences using 5 point Likert-like scale and implicitly (using partial beta coefficients).</td>
</tr>
<tr>
<td>Direct Classification Method</td>
<td>Classifies each product feature requirement directly into Basic, Performance and Excitement after the theory underlying this categorization is explained to the respondent.</td>
</tr>
<tr>
<td>Kano Survey Method</td>
<td>Classifies each product feature requirement into Basic, Performance, and Excitement categories based on two questions 1. the functional question “How do you feel if this feature is present?” and 2. the dysfunctional question “How do you feel if this feature is NOT present?”. Users’ response to these questions on a five point Likert-like scale.</td>
</tr>
<tr>
<td>Dual Questioning Method</td>
<td>Classifies each product feature based on users explicit expression of Importance to Not Important on a 5 point scale with 5 being Extremely Important and 1 being Not Important and Difference among compared products (4 – Very Different;, 1- Very Similar)</td>
</tr>
</tbody>
</table>

**Table 4. Methods from Quality and Product Development Literature**

**Selection of Feature Selection Method**
Five feature selection methods were selected for evaluation. The rationale behind the selection of these five methods for evaluation from among the methods described in the previous sections is described in the following paragraphs.

**Binary Search Tree.** Racheva, Daneva and Buglione (2008) reviewed a number requirements prioritization techniques and classified them into two main categories: techniques used to prioritize small number of requirements (small-scale) and techniques that scale up very well (medium-scale or large-scale). Bebensee, van de Weird and Brinkkemper (2010) observed that as software products are developed for the market rather than a single customer, one can expect a larger number of feature requests from users. Hence techniques that scale up well are most appropriate for software products. They found that the Binary Search tree method scales up well for software products with medium-scale requirements. Another study by Ahl (2005) investigating the five ranking techniques of requirements prioritization - AHP, Binary Search Tree, Planning Game, 100 Points Method and PGcAHP (Planning Game combined with AHP) - found that Binary search tree was superior to all other methods on many counts including accuracy of results and scalability. Binary search tree was therefore chosen from among the non-grouping (ranking) techniques as the first technique to be assessed in the study.

**Priority Groups Method.** Medium-scale or large-scale prioritization techniques might be based on relatively complex algorithms or at least due to the large amount of requirements need tool support (Racheva, Daneva and Buglione, 2008). However, sophisticated prioritization techniques are found to have limited ability to support requirements prioritization in market-driven product development with professionals in industry preferring simple tools instead (Lehtola and Kauppinen, 2006; Berander and Andrews, 2006). The Priority groups method is one such simple classification technique which ranks requirements into three priority categories, High, Medium and Low (Wiegers, 1999). It is an IEEE recommended method (Silitti and Succi, 2006) and among the most traditional and best known (Lehtola and Kauppinen, 2006). Priority Groups technique was therefore chosen as the second technique for comparison.

**Kano Survey Method.** A review of the advantages and disadvantages of techniques for feature selection based on the three factor theory such as the Direct Classification method, Importance Grid method, Penalty-Reward contrast analysis method and Kano survey method by Mikulic and Prebez (2011) suggests that the Kano method was the most suitable. It was found to be both a valid and a reliable method for categorizing feature requests according to the three factor theory. Another study by Witell and Lofgren (2007) comparing Direct Classification method, Importance grid method, Kano survey method and a variant of the Kano survey method which used a 3 level questionnaire rather than the 5 level questionnaire of the Kano survey method (see Table 4) came to the same conclusion and recommended that practitioners continue to use the Kano survey method. For this study the Kano survey method was therefore chosen from among the various techniques based on the three factor theory as the third technique for evaluation.

**Dual Questioning Method.** One of the limitations of the techniques listed above is that they do not take in consideration market factors such as the availability of the features being assessed in competitive products. As this study is exploring a suitable technique for market-driven software products, it will also investigate the potential of the determinant attribute approach (Myers and Alpert, 1968) using the dual questioning technique as the fourth technique for evaluation.

**Hybrid Method.** In addition a fifth technique which is a combination of Dual questioning method and the Kano survey method is suggested for comparing its efficacy in feature selection. Although the three factor theory allows producers to make a strategic choice through classifying product feature requests into the three categories, it does not rank features within a category. In addition, it does not take in consideration market factors such as the availability of these features in competitive products. In the hybrid method, detailed in the experimental treatments section, the Dual questioning approach is expected to complement the Kano technique by providing a method for ranking the features within each category, keeping competition in view, after they have been categorized using the Kano method. This we expect will be relevant for producers of market-driven software products. It will provide them with additional information to select a lean set of features that give maximum user impact for the resources invested while simultaneously keeping the strategic options open for the management.

**METHOD**
An Experimental method was used in the study. Experimental research offers a methodical way of comparing differences in the effect of treatments (features selected using various feature selection techniques) on the dependent variable (user satisfaction).

**Experimental Setting**

Gmail is an exemplar of an evolving software product from Google. Since it was first introduced in April 2004, Gmail has today evolved to become a leading web based email platform by introducing innovative features. Ten randomly selected user feature requests were chosen in late October 2012 as the test instrument for this study. The pilot study used fifteen feature requests in the test instrument. But because subjects, during the debriefing session of the pilot study, gave feedback of cognitive overload, it was decided to include only 10 feature requests in the actual study. A sample feature in the set included “Allow sending emails/ replies to emails at a later time or date. Presently if the user has to send an email or a reply to email at a later date she can only save the draft and remember to send it when the date arrives.” For the full set of 10 feature sets included in the study please see APPENDIX.

**Subjects**

A young adult (ages 19-24) cohort was used as subjects because users in this age group are recognized as early adopters of the latest technologies and responsive to innovations (new features in our experimental context) (Ehrenberg et al., 2008). Two groups of subjects participated in the experiment. In the first round 139 students participated out which 122 valid responses were obtained. The valid responses from 69 females outnumbered the valid responses from 53 male subjects. In the second round we collected data from another group of 62 subjects, of which 59 responses were found valid from 30 female and 29 male subjects. The average age of the subjects was 21.3 years with the female subjects averaging 21.3 years and the male subjects averaging 21.2 years.

Actual users of Gmail were involved in the experiment because features should be important from the users’ perspective not the developer’s (Fellows and Hooks, 1998). The subjects were recruited from a state university as all subjects were required to use the university Gmail. All subjects were trained on the methods of feature selection used in the experiments and their consent taken before conducting the study. The 10 Gmail feature requests were read out aloud and subject response taken on whether they have understood the user requirements.

The sample size for the experiment was determined based on the effect size found during the pilot study. The pilot study was conducted with 49 subjects who were users of Astrid Task Manager a mobile app. Assuming a power of 0.8, alpha=0.05 (one tail) and a medium effect size obtained in the pilot, a look up of Cohen’s power primer (Cohen, 1992) gave the sample size. To account for mortality rate, as two rounds of experiments with a gap of one week between them were conducted in the study, and the possibility of invalid responses from the subjects, the sample size obtained from Cohen’s table was inflated by 20 % to get the required sample size of 54 subjects. As an exercise of abundant caution data was obtained from 139 subjects in the first round but the analysis was restricted to 122 valid responses. The data in the second round was collected from 59 subjects and analysis restricted to 52 valid responses.

**Experimental Treatments**

The requirement prioritization methods used by the subjects in arriving at the critical feature subset from the set of 10 Gmail user feature requests are described below.

**Binary search tree method treatment.** The Binary Search Tree Method has been used previously for software product feature prioritization. It provides a ranked list of requirements according to user preference. Prioritizing software requirements using this technique involves subjects constructing a binary search tree consisting of nodes equal to the number of candidate requirements. First a single node holding one requirement is created. Then the next requirement is compared to this node. If it is of lower priority than this node then it is assigned to the left of this node else it is assigned to the right of this node. This process continues until all requirements have been inserted into the binary search tree. The node at the extreme left of the binary search tree is of the lowest priority while the node at the extreme right is of the highest priority. If the nodes in a binary search tree are traversed in *in order*, then the requirements are listed in a ranked order of priority. Thus using the binary search tree approach involved subjects selecting the requirements one at a time and creating a binary search tree and then traversing the binary search tree in order to generate a ranked list.
Priority groups method treatment. The Priority Groups Method has been used previously for software product feature prioritization. It is based on grouping requirements into different (highest to lowest) priority groups, with clear and consistent definitions of each group. Although the number of priority groups may vary the use of three groups (High, Medium and Low) is the most common (Leffingwell and Widrig, 2003). The description for these groups is as follows (Wiegers, 1999):

- **Definition: High priority requirements** are mission critical requirements; required for next release
- **Definition: Medium priority requirements** support necessary system operations; required eventually but could wait until a later release
- **Definition: Low priority requirements** are a function or quality enhancement; would be nice to have someday if resources permit

Subjects used this description to categorize each Gmail feature request into one of the three groups.

Kano survey method treatment. The Kano Survey Method involved subjects responding to two questions for every product feature request: the functional question "How do you feel if this feature is present?" and dysfunctional question "How do you feel if this feature is NOT present?" The first question concerns the reaction of the user if the product includes that feature, the second concerns his reaction if the product does not include that feature. The user has to choose one of the five possible options for the answers for both the functional and dysfunctional question:

1. I like it this way
2. I expect it this way
3. I am neutral
4. I can live with it this way
5. I dislike it this way

Asking both functional and dysfunctional questions helps product managers assess user priorities. If the user expects some feature to be present, but can live without the feature, it is not a mandatory feature. Based on the user responses to the questions in both functional and dysfunctional form for each of the user’s requirements, the quickest way to assess the questionnaires is to map each response in Table 5 and determine the requirement category to which it belongs.

Dual questioning technique treatment. In the Dual Questioning Technique consumers are:

1. asked which features they consider important and then
2. asked how they perceive this feature as differing among the competitor products

Features ranked high in rated importance (5 - Extremely Important 1 – Not Important) but not thought to differ much (4 – Very Different, 1 - Very Similar) among the various products may not be the most determinant factor. The product of attribute importance and difference among products determines the ranking of feature requests. Attributes that are ranked high in importance and difference ratings among products in the same product category are considered more determinant than attributes that are ranked low in importance and difference ratings among products.

Experimental Design and Procedure

Round 1. Two groups of subjects took part in the experiment. In round 1 each subject in the first group of 139 subjects provided their requirement prioritization of the 10 feature requests by users of Gmail through a paper-based instrument that included questions related to the Binary Search Tree Method, Priority Grouping Method, Kano Survey Method and Dual Questioning Technique using the methods detailed in the previous section. The data obtained from the subjects in Round 1 was used to select a subset of features that added value to the software product and a subset of features that do not. The details of how the two subsets were created for each of the five methods for comparison of their efficacy are detailed in the method of analyses section.
Table 5. Categorization of Subject Responses

<table>
<thead>
<tr>
<th>Functional Question</th>
<th>Dysfunctional Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>Like</td>
<td>Q</td>
</tr>
<tr>
<td>Expect</td>
<td>E</td>
</tr>
<tr>
<td>Neutral</td>
<td>E</td>
</tr>
<tr>
<td>Live with</td>
<td>E</td>
</tr>
<tr>
<td>Dislike</td>
<td>P</td>
</tr>
<tr>
<td>Like</td>
<td>R</td>
</tr>
<tr>
<td>Expect</td>
<td>I</td>
</tr>
<tr>
<td>Neutral</td>
<td>I</td>
</tr>
<tr>
<td>Live with</td>
<td>I</td>
</tr>
<tr>
<td>Dislike</td>
<td>B</td>
</tr>
</tbody>
</table>

B-Must have requirements or Basic Features
P-Performance requirements or Expected Features
E-Excitement requirements or Augmented Features
R-Reverse, i.e. wrong features, that would make the consumer experience worse
Q-Questionable, i.e. the consumer answers is inconsistent
I-Indifferent, i.e. the consumer does not care about this feature

Round 2. In Round 2 the data on user satisfaction with the current version of Gmail and after implementing the two feature subsets obtained from round 1 was captured from the second group of 68 subjects. Perceived user satisfaction was used as a dependent variable because the producer would want to know the impact of the feature subsets before rather than after implementing the features. Subjects rated their satisfaction for each of these experimental conditions using a single item 7 point scale (Andrews and Withey, 1976) with a neutral midpoint of 4, terrible at one end of the scale (1) and delighted at the other end of the scale (7): 1 - Terrible 2 – Unhappy 3 – Mostly Dissatisfied 4 – Neither Satisfied nor Dissatisfied 5 – Mostly Satisfied 6 – Pleased 7 – Delighted. Single-item measures offer advantages of being short, flexible and easy to administer (Pomeroy, Clark and Philip, 2001). They are also less time consuming and not monotonous to complete (Gardner et al., 1998), thus reducing response biases (Drolet and Morrison, 2001a). Hence they are appropriate for use in large scale studies (Robins, Hendin, and Trzesniewski, 2001).

Control Procedures. Control procedures were used to mitigate effects due to extraneous variables. The extraneous variable i.e. “user segment” of Gmail users was controlled through the use of a homogeneous sample of student subjects. The “sequence effect” of manipulating different treatments a counterbalancing design using Latin squares (Sheehe and Bross, 1961) was used to get subject responses for different methods of feature selection. Every fifth subject got the same sequence (see Table 6).

The “individual differences” among subjects in the sample was controlled by using the repeated measure design. The measurement of dependent variable (user satisfaction) was repeated as subjects rated their responses on “user satisfaction” for each of the five methods of prioritization i.e. Priority Group, Kano survey method, Binary Search tree, Dual method and Hybrid method. All the feature requests in the survey instrument were randomly selected from actual pending feature requests of users of Gmail. They were re-worded in a simple and standard style to avoid bias. Shifts in structure, content and format may introduce unwanted sources of variability that may confound subject response.
Round 1: Feature Selection Method

<table>
<thead>
<tr>
<th>Subject</th>
<th>Priority groups</th>
<th>Kano</th>
<th>Binary Tree</th>
<th>Dual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject 1</td>
<td>Priority groups</td>
<td>Kano</td>
<td>Binary Tree</td>
<td>Dual</td>
</tr>
<tr>
<td>Subject 2</td>
<td>Kano</td>
<td>Dual</td>
<td>Priority groups</td>
<td>Binary tree</td>
</tr>
<tr>
<td>Subject 3</td>
<td>Dual</td>
<td>Binary Tree</td>
<td>Kano</td>
<td>Priority Groups</td>
</tr>
<tr>
<td>Subject 4</td>
<td>Binary Tree</td>
<td>Priority Groups</td>
<td>Dual</td>
<td>Kano</td>
</tr>
</tbody>
</table>

Table 6. Sequencing of Methods

Method of Analyses. Comparison of feature selection techniques that are structurally different from each other requires that care be taken during analysis. For example, the Kano survey method and the Priority groups method both group requirement into three different categories. But each of these methods has different categories and has a different basis for categorization. To compound the problem the binary search tree method and the dual questioning method do not group requirements into categories but produce a ranked order list of requirements with no clear direction on the cut-off point for either for selection of requirements or for the exclusion of requirements to be built into the product.

The Kano survey method identifies both a subset of features that add value to the product (Basic + Performance + Excitement features) and those that do not add value to the product (Indifferent features). Kano survey method was therefore used as the baseline. For instance, if the Kano survey method identifies ‘n’ features that are likely to add value to the product, then in the complementary subset there are '10-n' features that are not likely to add value to the product. The top ranked ‘n’ requirements identified by the Priority groups method, Binary Search tree method and Dual method were then chosen for comparison of efficacy in identifying features that provide value and bottom '10-n' features were chosen for comparison of efficacy in identifying waste. For the ranking methods such as Binary Search tree and Dual questioning method the set of ‘n’ value-add and '10-n’ non-value added feature sets could be easily derived from the rank order. For Priority groups method the ranking order was determined by High > Medium > Low and within each category by the descending order of the number of users who selected the features in that category. The number of features that add value for the Hybrid method was determined by considering only the common features identified by Kano survey and Dual questioning methods.

Aligned with the lean principles of focusing on value creation for the customer and elimination of waste, the top ‘n’ ranked features from the 5 methods would be compared for determining the efficacy of the methods in identifying the feature subset from a give set of feature requests that creates value for the users. The bottom 10-n’ features would be used to compare the efficacy of the methods in identifying waste. While one can deduce that the method/s which demonstrate greater efficacy in identifying a subset of features that create value for the user would also be the method/s that demonstrate greater efficacy in identifying the complementary subset of features that are wasteful, this may not be always the case. This is because IS product features exhibit complex inter-dependencies among themselves. The overall satisfaction with features in the feature subset may not be additive (Ruhe, 2005). Although a feature selection method in comparison with other feature selection methods may be able to identify a feature subset that provides higher perceived user satisfaction, its complementary feature subset may not necessarily generate lower perceived user satisfaction. Thus only methods which demonstrate efficacy in identifying both value added and non-valued feature subsets are likely to show promise for lean software development.

Repeated measure ANOVA was used to test the difference in user satisfaction as the same subjects take part in all conditions of the experiment in the second round. Feature selection technique was the independent variable and user satisfaction was the dependent variable. The measurement of dependent variable (user satisfaction) was repeated as each subject rated her responses on “user satisfaction” for all the five methods of prioritization i.e. Priority Group, Kano survey method, Binary Search tree, Dual method and Hybrid method. Using a standard ANOVA in this case is not appropriate because it fails to model the correlation between the repeated measures as the data violate the ANOVA assumption of independence. IBM® SPSS® Statistics Version 19 was used to run repeated measures ANOVA. ANOVA is robust against violations of normality but requires that the variances for each set of different scores and their covariances are equal. Violations of this assumption of sphericity can invalidate the analysis. The Mauchly’s (1940) sphericity test was therefore conducted to evaluate sphericity.
RESULTS AND ANALYSES

<table>
<thead>
<tr>
<th>Feature Number</th>
<th>Basic</th>
<th>Performance</th>
<th>Excitement</th>
<th>Indifferent</th>
<th>Questionable + Reverse</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>56</td>
<td>23</td>
<td>31</td>
<td>10</td>
<td>2+0</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>68</td>
<td>8</td>
<td>22</td>
<td>7+4</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>41</td>
<td>25</td>
<td>44</td>
<td>0+0</td>
</tr>
<tr>
<td>4</td>
<td>81</td>
<td>12</td>
<td>7</td>
<td>19</td>
<td>2+1</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>31</td>
<td>9</td>
<td>59</td>
<td>1+1</td>
</tr>
<tr>
<td>6</td>
<td>33</td>
<td>20</td>
<td>15</td>
<td>52</td>
<td>0+2</td>
</tr>
<tr>
<td>7</td>
<td>37</td>
<td>18</td>
<td>44</td>
<td>35</td>
<td>0+2</td>
</tr>
<tr>
<td>8</td>
<td>11</td>
<td>24</td>
<td>32</td>
<td>54</td>
<td>0+0</td>
</tr>
<tr>
<td>9</td>
<td>18</td>
<td>71</td>
<td>14</td>
<td>9</td>
<td>0+0</td>
</tr>
<tr>
<td>10</td>
<td>29</td>
<td>17</td>
<td>61</td>
<td>15</td>
<td>0+0</td>
</tr>
</tbody>
</table>

Table 7. Consolidation of subject responses from Kano survey

<table>
<thead>
<tr>
<th>Feature Number</th>
<th>Binary Search Tree Method</th>
<th>Dual Questioning Method</th>
<th>Priority Groups Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ranks 1-6</td>
<td>Ranks 7-10</td>
<td>Ranks 1-6</td>
</tr>
<tr>
<td>1</td>
<td>36</td>
<td>86</td>
<td>75</td>
</tr>
<tr>
<td>2</td>
<td>82</td>
<td>40</td>
<td>69</td>
</tr>
<tr>
<td>3</td>
<td>71</td>
<td>51</td>
<td>81</td>
</tr>
<tr>
<td>4</td>
<td>43</td>
<td>79</td>
<td>90</td>
</tr>
<tr>
<td>5</td>
<td>77</td>
<td>45</td>
<td>29</td>
</tr>
<tr>
<td>6</td>
<td>31</td>
<td>91</td>
<td>66</td>
</tr>
<tr>
<td>7</td>
<td>83</td>
<td>39</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>55</td>
<td>67</td>
<td>21</td>
</tr>
<tr>
<td>9</td>
<td>88</td>
<td>34</td>
<td>51</td>
</tr>
<tr>
<td>10</td>
<td>63</td>
<td>59</td>
<td>73</td>
</tr>
</tbody>
</table>

Table 8. Consolidation of subject responses from Other methods
Analyzing the subject responses obtained from round 1, we find that the Kano survey method (Table 7) identified (in bold) 6 features (Basic + Performance + Excitement features) that provided value to the user and 4 features which the users were indifferent to. The value added and non-value added features for other methods were determined by further analyzing the data obtained (Table 8) using the process detailed in the Method of Analyses subsection. The results of round 1 of the study are summarized in Table 9. As can be seen from Table 9 each method identifies a unique subset of features that are likely to add value to the product (features in black) and a unique subset of features that are not likely to add value to the product (in white). For a full description of each feature see APPENDIX.

<table>
<thead>
<tr>
<th>Feature Number</th>
<th>Kano Priority Group</th>
<th>Dual</th>
<th>Binary Search Tree</th>
<th>Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 9. Summary of Features selected

The descriptive statistics of the mean user satisfaction (V=value added features, NV=non-value adding features)) under different experimental treatments is shown in Table 10.

<table>
<thead>
<tr>
<th>EXPERIMENTAL CONDITIONS</th>
<th>Mean User Satisfaction (NV)</th>
<th>Mean User Satisfaction (V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current version (1)</td>
<td>4.541</td>
<td>4.541</td>
</tr>
<tr>
<td>Kano Survey Method (2)</td>
<td>4.655</td>
<td>4.984</td>
</tr>
<tr>
<td>Priority Groups Method (3)</td>
<td>4.721</td>
<td>4.679</td>
</tr>
<tr>
<td>Dual method (4)</td>
<td>4.789</td>
<td>4.936</td>
</tr>
<tr>
<td>Binary Search tree method (5)</td>
<td>4.749</td>
<td>4.656</td>
</tr>
<tr>
<td>Hybrid method (6)</td>
<td>4.771</td>
<td>4.682</td>
</tr>
</tbody>
</table>

Table 10. Descriptive Statistics
After analysis of data obtained from the first group of subjects in round 1, we collected data from the second group of subjects in round 2. Each subject rated their satisfaction with the 6 features, obtained using the 5 different methods, that added value to the user and 4 features that did not add value to the user.

To determine if there is a significant difference between satisfaction with the various feature subsets that add value a repeated measure ANOVA was performed. Mauchly’s (1940) sphericity test was conducted to evaluate sphericity. The data violated the assumption of sphericity as the probability of Mauchly’s test statistic (p=0.000) was less than 0.05. Applying the Greenhouse-Geisser (1959) correction it was found that the difference in mean scores of user satisfaction with Gmail under the experimental conditions (treatments) for subsets that add value to the Gmai are statistically significant (p=0.001). However, although an overall significant difference in means was observed one does not know where those differences occurred. The Bonferroni post-hoc test results summarized in Table 11 were therefore examined (row - column) to discover which specific means differed significantly.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.443**</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.138</td>
<td>-0.306**</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.395**</td>
<td>-0.048</td>
<td>0.257*</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.115</td>
<td>-0.328**</td>
<td>-0.022</td>
<td>-0.280*</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.141</td>
<td>-0.302*</td>
<td>0.003</td>
<td>-0.254*</td>
<td>0.026</td>
<td>0</td>
</tr>
</tbody>
</table>

p < .05 ** P < .01 ***p<.001

Table 11. Difference in User Satisfaction (V)

To determine if there is a significant difference between satisfaction with the various feature subsets that did not add value to the software product a repeated measure ANOVA was performed again.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.114</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.180</td>
<td>0.066</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.248*</td>
<td>0.134</td>
<td>0.068</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.208*</td>
<td>0.094</td>
<td>0.028</td>
<td>-0.040</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.230*</td>
<td>0.116</td>
<td>0.050</td>
<td>-0.018</td>
<td>0.022</td>
<td>0</td>
</tr>
</tbody>
</table>

* p < .05 ** P < .01 ***p<.001

Table 12. Difference in User Satisfaction (NV)

Mauchly’s (1940) sphericity test was conducted to evaluate sphericity. The data did not violate the assumption of sphericity as the probability of Mauchly's test statistic (p=0.45) was greater than 0.05. Assuming sphericity it was found that the difference in mean scores of user satisfaction with Gmail under the different experimental conditions (treatments) for subsets that add value to Gmail are statistically significant (p=0.000). However, although an overall
significant difference in means was observed one does not know where those differences occurred. The Bonferroni post-hoc test results summarized in Table 12 were therefore examined (row - column) to discover which specific means differed significantly.

**DISCUSSION AND INTERPRETATION OF RESULTS**

Looking at the columns representing Current version and titled ‘1’ in both Table 11 we see that the Kano survey method (row 2) demonstrated superior efficacy (significantly higher perceived user satisfaction) in identifying a feature subset that add value to the users of the software product compared to all other methods expect the Dual questioning method. Also among the five methods only the feature subsets identified by Kano survey method and Priority groups method demonstrated efficacy in identifying non-value added feature subsets that did not impact user satisfaction significantly (see column 1, Table 12).

Thus the results show that overall the Kano survey method demonstrates promise for lean software development. While the Dual method performed better statistically than the Binary Search tree method, Priority groups method and the Hybrid method in identifying the features that add value to Gmail, it did not demonstrate efficacy in identifying features that do not add value to Gmail. Also, while the Priority groups method performed better statistically than the Dual method, Binary Search tree method and the Hybrid method in identifying the features that do not add value to Gmail, it did not demonstrate efficacy in identifying features that add value to Gmail. Only the Kano method demonstrated efficacy in identifying features that add value to the IS product as well as in identifying features that did not add value to the IS product.

The proposed Hybrid technique using the common feature sets derived from Kano survey method and Dual method showed mixed results. On the negative side, in spite of having to ask four questions to the user for each feature, the hybrid method demonstrated lower efficacy in identifying a feature set that provides maximum value to the user compared to the Kano survey method. On the positive side the subjects using the Hybrid method on an average identified a significantly (p=0.000) lower number of features (3.41) that add value to the software product compared to all the other methods (4.35) in Round 1. On the aggregate level the comparison for the mean user satisfaction by subjects in Round 2 was made for 4 value added features for the Hybrid method compared with 6 value added features for all the other methods (i.e. 22.22 % lesser features) and yet there was no significant difference in the mean satisfaction feature set identified by the Hybrid method compared to the feature sets identified by the Dual the Priority groups and the Binary search tree methods.

**CONTRIBUTION**

Feature selection is a critical process that impacts development expenses and software products’ market potential, yet it is a vexing issue for a software development organization. Choosing the appropriate set of next-release requirements from a larger set of candidate requirements has consequences for the customer and the development organization. For the customer of a market-driven software product, the appropriate set of selected requirements must deliver the expected functionality of the application domain and differentiate the product meaningfully from its competitors. For the development organization, the appropriate set of requirements must meet the customer’s desired functional expectations and minimize the resource outlay.

In response to calls in literature this study explored techniques not common in the field of requirements engineering and compared them with some of the commonly used techniques for software products. Developers of market-driven products face special challenges and unlike in-house development, the customer of market-driven software products does not fund the product upgrade. Thus lean development principles are appropriate as their focus is to reduce waste during software development (Jailia et al., 2011). In line with lean principles, this pioneering study assessed the efficacy of various methods for reducing waste in software development due to adding extra features which do not add value to the users of the software product. Although a number of research studies have compared the efficacy of methods of requirements prioritization based on various criteria, none of them have investigated their efficacy in reducing waste from the view point of the users.

The results of this empirical study corroborate observations made in research literature that traditional requirements engineering techniques may not be best suited for market-driven products such as the software products. The Kano survey method from product quality literature demonstrated potential in accurately identifying those features that users value as well as those that users do not care about. This has implications for producers of software products.
Producers can evaluate the incremental impact of adding new features on user satisfaction. In addition, the software organization is freed from pursuing “maximum requirements coverage” to being empowered with information allowing it to meet the customer expectations while at the same time conserving organizational resources.

LIMITATIONS AND FUTURE RESEARCH

The subjects chosen for the empirical study were youth between 19-24 years of age. The rationale was to get as homogenous a group of sample as possible as the objective of the study was to control extraneous variables such as segmental difference in user preferences and mitigate alternative explanations for the results obtained. These design choices may limit the generalizability of the findings of this study. To assess the generalizability of results obtained in this study, the study may be replicated for other user segments and other software products. In addition, this study considered a mature and established software product such as Gmail. Future studies may replicate this study for software products in other stages of their life cycle, such as Introductory, Growth and Decline stages. Future researchers may also investigate the potential of the Hybrid technique in identifying critical user requirements under severe resource constraint conditions.

REFERENCES


Drolet, A. L., Morrison, D. G. (2001a) A practitioner’s comment on Aimee L. Drolet and Donald G. Morrison’s “Do we really need multiple-item measures in service research?”, Journal of Service Research, 3, 196–204.


APPENDIX. The 10 User Feature Requests Used in the Study

<table>
<thead>
<tr>
<th>No</th>
<th>Feature description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Allow sending emails/ replies to emails at a later time or date. Presently if the user has to send an email or a reply to email at a later date she can only save the draft and remember to send it when the date arrives.</td>
</tr>
<tr>
<td>2</td>
<td>Allow user to have another view of their inbox below the message they are composing. This will allow users to reference information from one or more emails, if required, while composing</td>
</tr>
<tr>
<td>3</td>
<td>Provide preview of media stored on other sites within an incoming Gmail message when the sender includes only a link. Users get tired of clicking on links to get to the videos and photos of friends</td>
</tr>
<tr>
<td>4</td>
<td>Allow sub-string, partial word and wildcard search to provide a powerful mechanism for searching relevant emails</td>
</tr>
<tr>
<td>5</td>
<td>Threaded conversations should be made optional to users. Presently it is a mandatory feature</td>
</tr>
<tr>
<td>6</td>
<td>Gmail should allow users to report spams to the appropriate authority automatically. This will discourage spammers from spamming in future</td>
</tr>
<tr>
<td>7</td>
<td>Send an email to the user’s inbox when she marks Bcc (Blind carbon copy) to herself. Currently there is no email sent to the user if she Bcc’s herself</td>
</tr>
<tr>
<td>8</td>
<td>Allow change in account name without losing contents. Currently the user password can be changed but not the account name</td>
</tr>
<tr>
<td>9</td>
<td>Open more than one Gmail account at the same time. Presently the user can only open one Gmail account at a time</td>
</tr>
<tr>
<td>10</td>
<td>Allow use of specific colors for emails received from sources specified by the user. This will allow the user to quickly focus clearly on those emails that are important to her</td>
</tr>
</tbody>
</table>
The Nature of Adherence to Planning –
Systematic Review of Factors Influencing its Suitability as Criterion for IS Project Success

Dirk Basten
University of Cologne
basten@wiso.uni-koeln.de

Ali Sunyaev
University of Cologne
sunyaev@wiso.uni-koeln.de

ABSTRACT
Derived from engineering, adherence to planning (ATP) is the central and most often used criterion for the evaluation of information system (IS) projects. Although this evaluation is questionable, as ATP does not account for all of IS projects’ particularities, a systematic evaluation of ATP’s suitability in the context of IS projects is still missing. As a first step to close this gap, we use aggregations of the project life cycle’s processes and conduct a systematic literature review to identify research dealing with these aggregations. Our results show that ATP’s suitability depends on an IS project’s context, and is not given or at least questionable in many cases. Researchers and managers should adapt the way of evaluating IS projects to avoid misleading implications.

Keywords
Adherence to planning, information system, project success, literature review.

RESEARCH CONTEXT AND QUESTION
Scholars have been discussing adherence to planning (ATP) as success criterion for projects, in general, and information system (IS) projects, in particular, over the last two decades (Atkinson, 1999; Cuellar, 2010; Pinto, 2004; Wateridge, 1995, 1998). ATP as the traditional approach to assess project success (Atkinson, 1999; Baccarini, 1999) usually inhibits three criteria. Two of them are the same in all cases: budget and schedule. The third criterion is referred to as quality (Atkinson, 1999), performance (Pinto, 2004), and specification (Wateridge, 1998), and concerns the requirements and the question as to whether the system under implementation fulfills these needs. Therefore, ATP’s definition in this article denotes a project’s conformance to budget, schedule, and specified (functional and non-functional) requirements (Agarwal and Rathod, 2006).

A project is in most cases defined in terms of cost, schedule, and performance requirements. Although IS projects in many ways differ from projects in general (Fuller, Valacich and George, 2008, pp. 12-14), ATP has been transferred to the context of IS projects (Cuellar, 2010) to measure implementation or rather project management success; this is probably due to its easy measurement (Pinto and Slevin, 1988). In companies, ATP is the central and most often used criterion for the evaluation of IS projects (Collins and Baccarini, 2004; Joosten, Basten and Mellis, 2011; Thomas and Fernández, 2008). Although project managers argue for using additional criteria, their companies, as a matter of measurability, rely on ATP for the evaluation of their IS projects (Joosten et al., 2011).

Studies describing success rates of IS projects solely or mainly rely on these criteria as well (El Emam and Koru, 2008; Sauer and Cuthbertson, 2003; Sonnekus and Labuschagne, 2003; The Standish Group International, 2009). These studies report that only about 16-50% of all IS projects are successful (measured in terms of ATP). However, these studies may be biased (Glass, 2005). The most prominent study, the Standish Group’s CHAOS Report (The Standish Group International, 2009), has been criticized for shortcomings like incomplete description of study design, lack of reporting project selecting criteria, and the insufficient definitions of successful and failed projects (Eveleens and Verhoef, 2010; Jørgensen and Molokken-Østvold, 2006). Nevertheless, scholars often cite this report to motivate further areas of research and to demonstrate the importance of successful project management (e.g., Balijepally, Mahapatra, Nerur and Price, 2009; Chiang and Mookerjee, 2004).

Despite ongoing discussions, researchers have not solved the problem of a generally accepted measurement concept for IS project success. Such a concept is needed to derive valid and meaningful implications concerning critical success factors in IS research and practice. Many researchers use ATP as a dependent variable to derive IS projects’
critical success factors (e.g., Mitchell, 2006; Yetton, Martin, Sharma and Johnston, 2000). The generalizability of their results has to be scrutinized as the dependent variable ATP only covers a limited perspective of IS project success that does not necessarily match even project managers’ perceptions (Basten, Joosten and Mellis, 2012; Furulund and Moiskkonen-Ostvold, 2007b). This difference between subjective success perceptions and the success assessment in terms of ATP leads to projects seen as successful failures or failed successes (Nelson, 2005). The phenomenon why ATP as success criterion is suitable in some IS projects, whereas it seems to be totally unsuitable in others, seems to depend on a project’s context but is still unexplained.

As can be seen, ATP is a central point in IS project success measurement in research and practice. Although many scholars have argued against the use of ATP for the measurement of IS project success as sole criterion and in general (e.g., Atkinson, 1999; Baccarini, 1999; Cuellar, 2010), others rely on ATP to assess whether projects are successful (e.g., Mitchell, 2006; The Standish Group International, 2009; Yetton et al., 2000). Without agreement between these two diverging views, there is lack of a common understanding whether ATP should be used for the evaluation of IS projects. Consequently, research is in need of a systematic approach for the assessment of ATP’s suitability in the context of IS projects. As this approach is still missing, we make a first step to close this gap by answering the following research question:

What factors affect the suitability of adherence to planning as success criterion for IS projects?

To answer this question, we need to understand ATP’s nature as criterion for IS project success, that is, what constitutes the suitability and the factors that have an impact on its suitability as success criterion. Identifying such factors from the project management literature (Project Management Institute, 2008), we conduct a systematic literature to identify and analyze research that deals with these factors or their interrelations. Through this synthesis of literature, we contribute to the theoretical understanding concerning the suitability of using ATP as success criterion in IS projects. Based on our study, practitioners should become aware of the nature of ATP as success criterion and adapt current assessments that solely use this criterion.

The remainder of this paper is the following. Next, we describe our approach to identify factors concerning the suitability of ATP as success criterion in IS projects. We then describe the process that we applied to identify articles that focus on these factors or their interrelations. Afterwards, we present and discuss our findings in the context of IS project success and project management in general. Finally, we provide implications for research and practice.

PROJECT LIFE CYCLE – IDENTIFICATION OF FACTORS

Consolidating existing definitions in project management literature (Kerzner, 2006; Nicholas and Steyn, 2012; Project Management Institute, 2008; Turner, 1993), we define an IS project as follows: A project is a unique series of multi-functional activities within several phases, with a specific objective to create a product or service within certain specifications, with a defined start and end dates, and funding limits. A project consumes resources and, due to its uniqueness, entails risks and uncertainty. An IS “can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization” (Laudon and Laudon, 2009, p. 46). We define an IS project as a project in above terms with the goal to develop, extend, or adapt an IS.

In line with the above definition, IS projects need to fulfill certain objectives (functional and non-functional requirements) within specified limits (time and budget). To evaluate ATP’s suitability as IS project success criterion, it is necessary to assess whether the assessment in terms of a project’s conformance with its planned budget, schedule, and specified requirements is conducted correctly. Important criteria to assess whether assessments based on ATP are correct are thus the realism of plans, and second, the correct tracking of a project’s progress.

As ATP origins from general project management, we chose the project life cycle processes as proposed by the Project Management Institute (Project Management Institute, 2008, pp. 45-65) as starting point for our analysis. The processes (e.g., effort estimation) and their results (e.g., status reports) are the factors affecting the project management data (time, budget, and requirements) and thus also its comparison (planned vs. actual). We analyzed those processes to identify factors that have an impact on the suitability of ATP as success criterion in IS projects. In order to abstract from the multitude of variables (processes, inputs, and outputs), two researchers independently coded the processes and their inputs and outputs by using content analysis as proposed by (Jankowicz, 2004). We
applied the bootstrapping technique to aggregate the identified variables. This approach led to requirements, estimates, plans, changes, and progress reports as factors for our study.

LITERATURE REVIEW

Following King and He (2005, p. 667), our literature review can be categorized as a narrative one. As narrative review, we focus on “verbal descriptions of past studies focusing on […] elementary factors and their roles […] regarding a hypothesized relationship”. This type of review is in line with our objective to examine the important and controversial topic of ATP’s suitability as criterion for IS project success (King and He, 2005)


Articles relating dealing with the identified factors were the search’s focus. The search included articles that were published since the beginning of 1995. The identification of articles also included a search backward and forward (Webster and Watson, 2002). Initially, we identified 63 articles of potential interest. We read these articles in more depth to decide on final inclusion. If an article provided insights into the conditions, the current status, the causes or consequences of one or more of the factors, it was included. A total of 45 articles were excluded. For the remaining 18 articles, our search backward and forward using Google Scholar led to a total of 26 articles. The literature’s categorization was concept-driven (Webster and Watson, 2002). In most articles, the effects on budget and schedule and deviations between planned and actual data are the main focus.

FINDINGS

In this section, we present the results of our literature review. While analyzing the factors identified in our first research step (cf. Project Life Cycle – Identification of Factors), we identified interdependencies between these factors as illustrated in Figure 1. We use these interdependencies to structure our findings in this section. The articles in our review are presented in Table 1.

![Figure 1. Influences on the Suitability of ATP as Criterion for IS Project Success](image)

**Figure 1. Influences on the Suitability of ATP as Criterion for IS Project Success**
Table 1. Concept-Matrix: Factors Influencing ATP’s Suitability as Criterion for IS Project Success

<table>
<thead>
<tr>
<th>Citations</th>
<th>Requirements</th>
<th>Estimates</th>
<th>Plans</th>
<th>Project Changes</th>
<th>Progress Tracking</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Abdel-Hamid, 1988)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Andersen, 1996)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Austin, 2001)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Bakker, Boonstra and Wortmann, 2010)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Brooks, 1995)</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>(Costello, 1984)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Dvir and Lechler, 2004)</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>(Dvir, Raz and Shenhar, 2003)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(El Emam and Koru, 2008)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Glass, Rost and Matook, 2008)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Grimstad, Jørgensen and Moløkken-Østvold, 2006)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Han and Huang, 2007)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Hofmann and Lehner, 2001)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Iacovou, Thompson and Smith H. Jeff, 2009)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Jørgensen, 2006)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Jørgensen and Carelius, 2004)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Jørgensen and Grimstad, 2008)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Jørgensen and Sjøberg, 2004)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Keil, Im and Mähring, 2007)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Keil, Mann and Rai, 2000a)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Keil and Park, 2010)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Keil, Tan, Wei, Saarinen, Tuomainen and Wassenaar, 2000b)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Kitchenham and Linkman, 1997)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Lederer, Mirani, Neo, Pollard, Prasad and Ramamurthy, 1990)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Moløkken-Østvold, Jørgensen, Tanilkan, Gallis, Lien Anette C. and Hove Siw E, 2004)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Parkinson, 1955)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(1) Customers’ Requirements

Contracted IS projects have internal or external customers defining the IS requirements. Although the contractor participates in the elicitation process, the requirements represent the customer’s needs. To define the requirements, the involvement of customers or rather end-users is necessary and important (Dvir et al., 2003). Requirements engineering is supposed to be most crucial to IS development, mainly due to their great influence on later IS project phases (Hofmann and Lehner, 2001). Thereby, a project’s scope and requirements are often (especially during early phases) unclear till the completion of a project. The level of uncertainty depends on the project’s characteristics (e.g., level of innovation). This can be attributed to two distinct reasons. First, the customer is often not aware of what is really required in the desired IS (Fuller et al., 2008). Second, the process of specifying requirements is in many cases problematic (Jiang et al., 2009). The difficulties of predicting what is required can be attributed to the rapid changes that are underlying IS development (Gorschek, Svahnberg, Borg, Loconsole, Börstler, Sandahl and Eriksson, 2007). This leads to high uncertainty and thus a need for risk management. Thereby, risks are influences that may affect a project’s budget and schedule (Lewis, 2005). Only if risks are controlled, a realistic assessment of a project’s effort is possible. However, just the knowledge about risks is not sufficient (Bakker et al., 2010) and it is
too expensive to control all risks that may occur in IS projects (Kitchenham and Linkman, 1997). As requirements uncertainty may lead to rework, it is one of the greatest risks for developing an IS (Han and Huang, 2007). Therefore, projects with more effort in risk assessments are more likely to conform to their plans. In this context, a high degree of uncertainty related to continuance of a project leads to negative project results (Keil et al., 2000b). It is a likely risk that too little effort is spent on project planning (Boehm, 1991) and research seems to mainly focus on how risk management is supposed to work, instead of focusing on how it is really done in practice (Bakker et al., 2010).

If a project is not directly contracted to an internal IS development department, the customer may announce the contract for bidding. The design of the bidding process can influence a project’s granted budget (Jørgensen, 2006; Jørgensen and Carelius, 2004). The number of bidders should be kept low and it should be asked for bids as late as possible to secure that requirements are as certain as possible.

2) Requirements and their Influence on Estimation Accuracy

Estimates are delivered for the effort to implement a project’s specified requirements. Depending on the resources available, the time needed to implement such requirements can be derived. Consequently, estimation accuracy at least partly depends on project requirements.

With a high certainty, estimates are inaccurate (Kitchenham and Linkman, 1997). Apart from IS professionals seeing unclear requirements as cause for estimation inaccuracy (Furulund and Moløkken-Østvold, 2007a; Jørgensen, 2004; Jørgensen and Moløkken-Østvold, 2004), experiments have shown slightly different wordings in requirement specifications (irrelevant and misleading information) leading to significantly different estimates (Jørgensen and Grimstad, 2008). In this context, customer expectations also need to be mentioned. Customer expectations can be seen as additional requirements. It has been shown that such requirements significantly influence effort estimates when known by the estimation team (Jørgensen and Grimstad, 2008; Jørgensen and Sjøberg, 2004).

Furthermore, budgets are influenced during project initiation by haggling and political actions (Lederer et al., 1990). If, for example, user representatives believe that the project will not be accepted due to too high cost, the estimated effort is reduced without adapting the project’s scope.

3) Changes of Project Requirements

The positive effect of planning quality is overridden to a large extent by goal and plan changes (Dvir and Lechler, 2004) that often occur in IS projects (Javed, Manzil-e-Maqsood and Durrani, 2004). Thereby, the relation between requirements and project changes is bi-directional.

First, project’s requirements are often unclear, ambiguous, and incomplete in the beginning (cf. section (1) Customers’ Requirements). Apart from requirements uncertainty, change requests occur due to the demand of new features. Both causes have to be regarded to secure estimation accuracy and realism in project plans. Second, problems due to initially specified requirements change the project’s scope. Thus, new, extended, or removed features have an impact on the requirements that are meant to be implemented during the project.

If a project’s scope changes, it is important to precisely assess the consequences. As professionals name change requests as one of the most important reasons for inaccuracy (Jørgensen and Moløkken-Østvold, 2004; Lederer and Prasad, 1995), it seems that related consequences are not sufficiently assessed. Not changing plans and consequently exceeding budgets may be one reason for project failure as changes in requirements and scope have been identified as one of the main reason for project cancelation (El Emam and Koru, 2008). Consequently, project changes need to be accurately considered as they influence project plans and their tracking and thus indirectly affect ATP suitability (cf. Figure 1).

4) Estimation Accuracy and Plans Realism

Even without any empirical evidence, it seems obvious that inaccurate estimates cause unrealistic plans. If estimates are inaccurate, plans that are built on these estimates will in most cases be unrealistic. For example, if activities on a project’s critical path exceed their planned duration as a consequence of underestimated effort, the plans are not realistic. In this context, activity planning can be a problem for the performance of projects if applied in early phases.
of projects (Andersen, 1996). If only insufficient information is available, project planning should start with milestones and a detailed planning for every milestone should occur right before work starts on that milestone.

(5) Project Changes and Plans Realism

Change requests are seen as cause for deviations between planned and actual effort (Jørgensen and Moløken-Østvold, 2004; Lederer and Prasad, 1995) indicating that project changes are often neither considered in the underlying requirements nor in re-estimations. Project’s initial plans are only a starting point. The adaption of plans due to changes seems to be more essential (Dvir and Lechler, 2004). If project plans are not adapted due to requirements changes, it is not realistic that the project will be completed in accordance with its plan. As can be seen, changes and according plan adaptions have a considerable impact on ATP’s suitability (cf. also Figure 1).

(6) Tracking and Reporting of Project Changes

Subsection (5) is also important for tracking and reporting a project’s current status (cf. Figure 1). If project changes are not considered in the requirements and estimates, this bias influences tracking project progress. Changes that are not considered in project plans adulterate the comparison of planned and actually realized effort. As a consequence, progress reports do not reflect the project’s actual state. Reliable reporting is supposed to be a critical success factor in IS development (Iacovou et al., 2009). For reliably monitoring and controlling progress, project management depends on updated data constantly at its disposal. Otherwise, it is not possible to detect deviations and take corrective actions accordingly.

To guarantee updated data, changes need to be communicated. Professionals lying on software projects (Glass et al., 2008) indicate that realistic values concerning the current state of a project are not always available. Additionally, project managers often do not even know about lying. In such cases, it is almost impossible to assess a project’s current state. Team members may just not know the percentage of work completed and how much longer it will take them to finish their tasks. According to the 90% syndrome (Abdel-Hamid, 1988), a project’s progress is constantly reported up to 90% and IS projects seem to be almost completed for most of their duration. In this context, it is critical to decide whether projects should be continued if the prospects for success are not obvious (Keil et al., 2000b).

(7) Reporting Progress regarding Project Plans

To report project progress according to ATP, the planned and actual state (requirements, budget, and schedule) are compared. Apart from the influence of change requests, reporting a project’s progress can only be correct if the plans are realistic (cf. points 4 and 5). Otherwise, implications for project management based on progress reports may be misleading.

Reporting project progress can be related to negative outcomes. In general, the reluctance to report bad news may lead to project failure. Reporting bad news on projects is a common topic in IS research (Keil et al., 2007; Keil and Park, 2010). Team members do not want to lose face and thus the project’s course may be in danger. Tracking project progress also has positive consequences. Team members’ motivation is presumably higher if it is obvious that the project is actually proceeding to its goals. In general, higher motivated teams may produce better results.

(8) Realism in Plans

ATP can only be suitable as IS project success criterion if project plans are realistic. If those plans are unrealistic due to inaccurate estimates or changes that are not considered, the comparison between planned and actual project data is misleading. Furthermore, unrealistic plans can have harmful consequences. If too few resources are available for project implementation, time pressure may lead to shortcuts (Austin, 2001), for instance, neglecting quality assurance. Deadlines are supposed to be the “greatest enemy of software engineering” (Costello, 1984, p. 15). It is argued that deadline pressure limits software engineers’ time to guarantee a system’s effectiveness. This shows that it is essential to carefully control the availability of resources during IS projects to guarantee ATP’s suitability (cf. Figure 1).

(9) Correctly Tracking and Reporting Project Progress
Throughout carrying out projects, it is necessary to monitor and control project work (Project Management Institute, 2008, pp. 59-64). Tracking and reporting the project’s status has a direct impact on ATP’s suitability (cf. Figure 1). Detection of budget and schedule deviations is important for correct status reports. Without controlling changes and tracking/reporting the status, the suitability of decisions made on the basis of the project’s status is questionable.

Although budget and schedule depend on a project’s scope, it is comparatively easy to control these values. Deviations between actual and planned schedule may lead to project management adding more staff. According to Brooks’ (1995) Law, this often makes the project even later due to the increased training and communication overhead and a higher need for communication in general. According to Parkinson’s (1955) Law, project plans tend to fulfill themselves. All available time will be used, so that tasks are never completed before schedule. In that way, team members try to avoid that a lesser amount of time is assigned for future tasks. This law can also be applied for IS projects. Time that is left for implementing tasks will not be used for other tasks, but to optimize efforts according to part of the system being worked on at present.

The coherence described in (8) also applies for the relation between project progress and ATP’s suitability as IS project success criterion. As the comparison of planned and actual data requires reports of the project’s actual status, ATP’s suitability is not given in cases in which reports are flawed. This effect is strengthened due to the progress report’s dependence on a plan’s realism. Thus, a plan’s realism has a direct and an indirect effect on ATP’s suitability.

(10) Effects on Project Management

The suitability of ATP depends on the aspects described above. In this context, ATP’s suitability as success criterion for IS projects affects project management. If the necessary conditions are not fulfilled (e.g., change requests are not regarded in re-estimations), the use of ATP as success criterion has negative consequences for project management. For example, projects might unnecessarily be cancelled if progress reports are wrong. Misleading project management decisions may also strengthen the effects of the commitment to failure behavior (Keil et al., 2000b).

Project escalation is a common phenomenon in IS projects (Keil et al., 2000a). Escalation of commitment to a failing course of action leads to so-called runaway systems. More resources are invested even if a cancelation would be appropriate. There are several theories that explain such behavior. One of these is described in the following. In case of self-justification theory (Keil et al., 2000a), commitment is kept high as justification for previous actions on that project. Therefore, success decisions based on ATP have to be considered carefully, as it is even worse if a project is continued under such conditions. Even if project data is accurate and realistic, this information might not be available to those who decide on a project’s continuance. Closing a project depends on the project’s current status and therefore is also an important aspect. As a project’s evaluation and decision on its continuance can have harmful effects, it is important that these decisions are made according to the actual status of the project. Thus, accurate progress reports are critical in this context. Measuring success in terms of ATP after a project’s completion is easier than during the course of the project. After project completion, it is not necessary to calculate the additionally effort needed. Then, a comparison of planned and actual data is decisive for success or failure. Thereby, it is necessary to agree upon overruns that are acceptable. The definition of ATP does not account for the degree of deviation necessary for a project to be considered a failure. Depending on the project duration, an overrun may be acceptable (except for special circumstances like legal bounds). The same applies for the criteria of budget and requirements. Success studies like the CHAOS Report (The Standish Group International, 2009) classify projects with potentially insignificant overruns as already being unsuccessful. As described above, using ATP in cases with low suitability can have harmful effects for project management and the success of projects. Thus, project management needs to regard ATP’s suitability (cf. Figure 1) to ensure an adequate management of projects.

DISCUSSION

Our study considers requirements, estimates, plans, changes, and progress reports as factors explains their interrelations with regard to ATP’s suitability as criterion for IS project success. The differences between IS projects and the uniqueness of single projects seem to be the predominant reasons for the resulting problems when deviations from the planned course of action occur. In our literature review (cf. Table 1), we identified articles showing that the maturity of the project life cycle’s processes is often not given in practice and that the resulting outcomes only seldom allow using ATP as IS project success criterion. Underlying requirements are often unclear, making plans that are drawn upon these requirements unrealistic. As stated before, the “use of estimated budget and schedule for

Basten & Sunyaev  
The Nature of Adherence to Planning
success evaluation assumes the efficiency of estimation models, which is a debatable issue for years to come” (Agarwal and Rathod, 2006, p. 360). Despite the improvements in estimation techniques’ accuracy, there is still room for improvements. Especially experience data concerning similar already completed projects may help (Furulund and Møløkken-Østvold, 2007a).

Additionally, correctly reporting a project’s current status is biased (e.g., 90% syndrome) and influenced by political actions (e.g., lying). Although change requests are quite common in IS projects, there seem to be only few studies that focus on coping with those in project plans (cf. Table 1). Despite the importance of changing plans, change requests are in many studies named as reason for deviations between the planned and actual progress. This clearly indicates that IS professionals do not adapt project plans; maybe to avoid losing face. Not losing face also seems to be a phenomenon leading to deviations between plans and a project’s actual course of action. As a consequence, reducing the effort for quality assurance can be attributed to the fact that project managers are evaluated on their ability to bring a project to an end that conforms to its budget, schedule, and specified requirements (Shenhar, Dvir, Levy and Maltz, 2001). Thus, reputation is an important influence on how projects are carried out. Existing research mainly focuses on the processes of eliciting requirements and estimating the related effort. Especially, adapting plans as a consequence of change requests is important. There seem to be only few studies that actually address the handling of deviations from a project’s planned course of action.

Not having experience data from similar projects is a high risk of not planning realistically. Famous examples from other disciplines, like the Sydney Opera House, corroborate this finding (Pinto, 2004). Such projects will probably be seen as success by most people. Simply comparing planning data and subjective perceptions shows that using ATP is not necessarily an adequate indicator for success. Furthermore, projects developed on-time and in-budget does not necessarily represent realistic planning (Grimstad et al., 2006). It is shown that only a reduction of effort and the freedom from defects may lead to development according to project plans.

Nevertheless, studies show that ATP is still a central criterion in assessing IS project success (Joosten et al., 2011; Thomas and Fernández, 2008). Despite a variety of other criteria that have been discussed and researched in literature, the predominance of ATP is still prevailing. An interesting point is the use of ATP (conformance with budget and schedule) to measure the efficiency of the development process (ratio of resources used and outcome achieved) in many studies (Aladwani, 2002; Crawford and Bryce, 2003; Shenhar et al., 2001; Thomas and Fernández, 2008). As efficiency is in general a valid index to measure process success, the focus on ATP seems reasonable if these concepts are seen as equal. However, this equalization is only valid in cases where project plans are realistic. The difference can be shown in terms of the following example. If project plans are unrealistically optimistic, it is no surprise if the corresponding projects exceed their budget and schedule to fulfill the specified requirements. Nevertheless, the use of resources can be efficient.

Although studies have shown that project planning itself contributes to project success (Dvir et al., 2003), it seems that predominantly negative consequences are analyzed. This may be due to the high rate of projects that experience budget and schedule overruns. Then, research aims to explain why these overruns occur by finding risk factors. It is assumed that avoiding these factors will lead to projects that are completed on-time and in-budget.

We conducted this study as a first step to solve the problem of diverging views concerning ATP’s suitability and its usage in IS contexts (cf. the introductory section). While a general statement on the suitability of ATP as criterion for IS projects cannot be derived, ATP’s suitability depends on a project’s context and thus needs to be assessed accordingly. If project plans are realistic, that is, the requirements are the ones that are really needed and the time and budget constraints are realistically planned, ATP can be used as success criterion. These conditions are seldom fulfilled in current IS practice. As the definition of success and failure depends on the degree of deviation between actual and planned data and this definition is not distinct, using ATP as success criterion during project closing seems unreasonable. Thereby, we need to differentiate between different contexts of IS projects. One specific group of projects may be characterized to be highly time critical. If, for example, an insurance company is legally bound to change its systems, the project cannot be seen as a success in case of schedule overruns. In such cases, the adherence to schedule criterion is at least a necessary condition for the project to be successful. The example shows that there is the definite need to partly fulfill ATP to make a project successful. Nevertheless, the time criticality of many of today’s projects (further examples are marketing announcements, first mover advantages, or contracted deadlines) shows that this dimension of ATP should not be neglected, and in some contexts may be the most decisive criterion. However, we emphasize that it is actually not a matter of how much time has been spend to realize the IS under implementation, but that the final release date is the decisive aspect. Thus, success is rather a matter of on-time
availability than development time. In this context, releasing an IS does not necessarily comprise all demanded requirements, but a workable system that can be released on time.

The suitability of ATP as IS project success criterion depends on the context it is used in. Thus, companies that use highly mature processes (e.g., highly rated according to the capability maturity model) are more likely to use ATP, as their development process might inhibit more accurate assessments of the inputs and outputs of the processes of the project life cycle. Additionally, the suitability of ATP’s suitability might even change during a single project. If requirements are more certain till the end of a project, the according plans might contain a higher realism as well. Thereby, the plans need to be adapted according to the requirements. As the IS development’s current state shows a different picture, reasons for professionals anyway using ATP all throughout the development process need to be analyzed. In this context, the finding that measuring IS project success is mainly a matter of measurability (Joosten et al., 2011) is a first step towards the solution of this problem.

CONCLUSIONS

The answer to the question as whether to use ATP as success criterion for IS projects depend on the context the criterion is used in. In this study, we provide an approach to systematically analyze the often questioned suitability of ATP as criterion for IS projects and make a first step towards explaining the phenomenon that IS project success cannot be adequately measured in terms of ATP in projects in general. Thereby, a project’s context determines the suitability. Accordingly, we provide an explanation for the diverging views in previous research on this topic. Depending on the situation, using ATP is suitable.

Although it is rather easy to control the performance indices that are related to ATP, it is rather difficult and costly to make sure that the conditions for controlling these indices are kept. This leads to the following implications: (1) Projects should not be evaluated primarily on the basis of ATP. (2) Success studies should use additional or rather other criteria to show the actual current state of IS projects. (3) Studies focusing on critical success factors have to apply different success criteria. Nevertheless, practitioners should not relinquish planning projects as plans and especially the initial estimates are needed.

We recommend using our study for future research. Especially, status reports (e.g., detection between planned and actual data) and the handling of change requests have been only marginally researched in the past. To use ATP as success criterion for IS projects, different conditions need to be fulfilled. Therefore, it is necessary to research different approaches in which these conditions can be fulfilled. Transferring our review to the context of contingency theory might be a suitable approach, as different characteristic of project contexts seem to determine the suitability of ATP as criterion for IS project success.

REFERENCES


The Planning Fallacy as an Explanation for Over-Requirement in Software Development

Ofira Shmueli  
Ben-Gurion University of the Negev  
ofirash@post.bgu.ac.il

Nava Pliskin  
Ben-Gurion University of the Negev  
pliskinn@bgu.ac.il

Lior Fink  
Ben-Gurion University of the Negev  
finkl@bgu.ac.il

ABSTRACT

Over-Requirement occurs in software development projects when a software product is specified beyond the actual needs. This study shows empirically that Over-Requirement happens partially due to the Planning Fallacy, i.e., the tendency of people to underestimate the time needed to complete a task. Underestimating the time needed to develop a software feature during project planning, we argue, may lead to including within the project scope more required and unrequired features than can be completed by the project deadline. To investigate this argument, we conducted an experiment in which participants were asked to estimate the time it would take to develop various software features in a software development project and then, given the project's duration, to recommend which of the features to include within scope. The results confirmed that the Planning Fallacy occurs in the context of software development and influences the Over-Requirement phenomenon.

Keywords

Planning fallacy, software development, over-requirement, over-specification, gold-plating, experiment.

INTRODUCTION

The phenomenon of Over-Requirement, also termed over-specification or gold-plating, occurs when a product or a service is specified beyond the actual needs of the customer or the market (Boehm and Papaccio, 1988; Ronen and Pass, 2008). In software development projects, Over-Requirement is a major risk since, once a development project is launched, it is very difficult to cut features off its scope, even when some features are not really necessary (Dominus, 2006). Despite its negative impacts, Over-Requirement is a common phenomenon reported to pertain to at least 30% of developed features (Coman and Ronen, 2009).

The purpose of this work is to show that Over-Requirement is partially due to the Planning Fallacy which, according to Kahneman and Tversky (1979), refers to the tendency of people to underestimate the time needed to complete a task even in the presence of knowledge about past failures and time overruns. However, less underestimation of task completion times has been found in situations in which past experiences are perceived as relevant to the task at hand (Griffin and Buehler, 2005) and in the case of predictions performed by an uninvolved observer (Buehler, Griffin, Lam and Deslauriers, 2012). The Planning Fallacy has been demonstrated in a wide variety of tasks, including origami folding, school work, and computer programming (Connolly and Dean, 1997; Pezzo, Litman and Pezzo, 2006).

We argue that in the case of software development, such underestimation of task completion times at the stage of project scoping may lead to including in the project more features (some of which are Over-Required) than can be completed within time. To investigate this argument, we conducted an experiment that required participants, referring to a specific to-be-developed software project, to (a) estimate development times of various software features and (b) recommend which of the features to include in the project scope given the duration of the project. The experiment results confirmed that the Planning Fallacy occurs in the context of software development and influences the extent of Over-Requirement. Moreover, for software engineers in both development and consulting roles, knowing actual previous development times may improve time estimations and decrease the magnitude of Over-Requirement. Yet, this knowledge eliminates neither the Planning Fallacy nor the inclusion of Over-Required features. In addition, whether they have knowledge of previous development times or not, software developers tend
to include in scope more features than software consultants, including Over-Required features. Finally, testing the relation between time estimation and Over-Requirement reveals that the more underestimation of development time exists, the more Over-Requirement occurs.

The main contribution of this work to both research and practice is in improving the understanding of Over-Requirement which, despite its negative influence, has barely been explored. This research proposes and empirically supports a behavioral explanation for the inclusion of Over-Required features in project scope, innovating in its approach of exploring behavioral effects in the context of software development as well as exploring their consequences on the whole software development project.

The next section of this paper presents the theoretical background for the experiment as well as this study's hypotheses. Then the following sections detail the experiment method and results. Finally, the paper concludes with a discussion of the results as well as this work's contributions and limitations.

THEORETICAL BACKGROUND AND HYPOTHESES

Over-Requirement

Boehm (1991) included Over-Requirement among the top 10 software development risks, and NASA (1992) listed Over-Requirement among the eight "don't do" warnings in the software-development context. Similarly, Over-Requirement was mentioned as one of the top 10 or 20 risks in most studies devoted to identifying, classifying, and ranking software-development risks (Baccarini, Salm and Love, 2004; Houston, Mackulak and Collofello, 2001; Khanfar, Elzamly, Al-Ahmad, El-Quwasmeh, Alsamara and Abuleil, 2008; Schmidt, Lyytinen, Keil and Cule, 2001). The major damages that make Over-Requirement risky vary from delayed launch, through excessive complexity, to demise of an entire company. The list of negative outcomes includes exceeding planned resources as time or budget (Buschmann, 2009; Coman and Ronen, 2009, 2010) and the undesirable impact on software complexity, reliability, and maintainability (Battles, Mark and Ryan, 1996; Buschmann, 2010; Coman and Ronen, 2009, 2010; Elliott, 2007; Westfall, 2005). In addition, project resources are being wasted on functionality of no value, instead of on core-business functionality (Coman and Ronen, 2009, 2010; Elliott, 2007; Westfall, 2005), and poor outcomes may affect the users and the entire organization (Coman and Ronen, 2009, 2010; Kautz, 2009; Rust, Thompson and Hamilton, 2006). Nevertheless, Over-Requirement is a common, hardly-reversible phenomenon since excessive extra features introduced during the requirement-engineering phase or later are very rarely cut off scope (Dominus, 2006; Wetherbe, 1991).

Planning Fallacy

The Planning Fallacy refers to the tendency of people to underestimate the time needed to complete a task even in the presence of experience of over-runs in similar tasks. The term for this behavioral effect was first proposed by Kahneman and Tversky (1979), who also acknowledged the generalizability of this effect to both experts and laypersons. Buehler, Griffin and Peetz (2010) emphasized later on its paradoxical nature as manifested by the combination of optimistic views about the future and realism about the past. The Planning Fallacy has been found for a wide variety of desirable and undesirable tasks, including origami folding (Pezzo et al., 2006) as well as school work, tax-form completion, and computer programming (Connolly and Dean, 1997). The Planning Fallacy seems to be a result of motivational factors, but Kahneman and Tversky (1979) suggested that it is also a consequence of the tendency of people to focus on the specific problem rather than on the distribution of outcomes of similar cases. A known psychological observation is that people tend to take credit for positive outcomes but attribute negative outcomes to external factors (Lovallo and Kahneman, 2003), providing another explanation for the tendency of people to provide overly optimistic time estimations. Overall, underestimation of completion times is a negative phenomenon because it does not shorten the actual time the task takes, unless the task is very simple and performed immediately (Buehler, Peetz and Griffin, 2010), negatively affecting the entire project. Underestimation done in early phases of the project tends to persist and leads to a sustained false optimism along later project phases (Kutsch, Maylor, Weyer and Lupson, 2011).

Decomposing a large task into small enough sub-tasks was shown to decrease time underestimation as a result of a segmentation effect (Forsyth and Burt, 2008), whether the task is holiday shopping, document formatting (Kruger and Evans, 2004), or code writing (Connolly and Dean, 1997). The more complicated the task is, the more can underestimation reduction be attained (Kruger and Evans, 2004). Motivating accuracy in time estimation by offering
monetary incentives for accurate time prediction, rather than motivating speed, might help as well (Brunnermeier, Papakonstantinou and Parker, 2008). However, while the Planning Fallacy applies to predictions about one's own tasks, predictions done by an external party are less prone to the Planning Fallacy (Buehler et al., 1995; Buehler et al., 2012).

This work investigates the Planning Fallacy in the context of software-development projects, focusing on the underestimation of development times. Time underestimation is bound to influence project scoping because underestimating the time it would take to develop software features may lead to the inclusion of more features within project scope than actually possible to complete by the project deadline. This scope overload may not only lead to time overruns but also to the Over-Requirement phenomenon, as hypothesized next.

**Hypotheses**

Experts and laypersons alike were found to exhibit the Planning Fallacy, ignoring knowledge gained in the past about the distribution of outcomes (Kahneman and Tversky, 1979). Even upon increasing the salience of past memories by asking about past experiences immediately before asking for time predictions did not result in less optimistic forecasts (Griffin and Buehler, 2005). Perceived irrelevance of past experiences was demonstrated also in the case of software engineers who, knowing that past projects tended to be late, did not use the results of past projects to inform future predictions (Buehler, Griffin and Peetz, 2010). However, a reduction of the Planning Fallacy was demonstrated under the recall relevance condition, in which subjects not only recall past completion times but also describe how the current task might be similar to past tasks (Griffin and Buehler, 2005). This study aims to find out if the role assumed by a software engineer, whether developer or consultant, makes a difference.

Software development projects are known to suffer from massive time overruns (Standish Group, 2009). The human tendency to underestimate and to produce overly optimistic development schedules was noted as the most common reason for failure of software projects (Charette, 2005; Nelson, 2007). Years of evolutionary improvement of software-development methods and tools, as well as the vast experience gained, have barely improved time estimation, probably due to the perceived irrelevance of the past based on the perception of specific projects as unique (Buehler et al., 2010). Thus, knowledge of the past does not diminish the Planning Fallacy. Yet, for software engineers, the presence of concrete and relevant knowledge concerning development times of similar software tasks in the past, as in the recall relevance condition, may reduce underestimation of the time it would take to complete a software task, while the absence of such knowledge may increase underestimation. Thus, we hypothesize that in the absence of past relevant knowledge, time underestimation increases. Hence,

**H1:** *Software engineers without past relevant knowledge of development times will provide lower total time estimations for software-development tasks than those with past relevant knowledge*

The Planning Fallacy and underestimation apply to predictions about one's own tasks. Yet, predictions done by uninvolved observers usually reduce underestimation and sometimes even result in overestimation (Buehler et al., 1995). Moreover, simply imagining the upcoming task from the perspective of another person moderates the optimistic prediction bias and reduces underestimation, helping to generate more realistic predictions by reducing cognitive and motivational processes that typically contribute to bias (Buehler et al., 2012).

In software-development projects, estimations of completion times are done usually by the software developers themselves, estimating the time it would take each of them to develop the software features. Buehler et al. (2010) have shown that, when asked to predict software-development times, developers neglect experience gained in past projects and consider the current project unique. Regarding the software features that developers are going to develop, we argue that they might be more prone to bias in estimating completion times than uninvolved observers, such as consultants. Thus, we hypothesize that the role of the person affects the degree of underestimation. Hence,

**H2:** *Software developers will provide lower total time estimations for their software-development tasks than software consultants*

Scope or feature creep in software-development projects is the tendency to include too many features within project scope (Buschmann, 2009, 2010). We argue that underestimating the development times may impact the number of features included in the scope of the project. In other words, underestimating development times may cause the illusion of having more time than there really is, resulting in the inclusion of more features within scope than can be
actually completed on time. Again, having past relevant knowledge may influence the number of features to be included, as is the software engineer’s role, whether developer or consultant. Thus, we hypothesize that the number of features included within scope increases in the absence of past relevant knowledge and in the case of software developers, consistent with the two main effects on time estimations described in H1 and H2. Hence,

**H3:** Software engineers without past relevant knowledge of development times will plan more features to be included within project scope than those with past relevant knowledge

**H4:** Software developers will plan more features to be included within project scope than software consultants

Over-Requirement in software development projects is the tendency to include unneeded features within project scope. We argue that H3 and H4 can be extended beyond the number of features included within scope also to the number of Over-Required features. This argument is based on the assumption that the underestimation of development times and the inclusion of more features within project scope are not contingent on feature importance. Again, past relevant knowledge and the individual perspective may impact the magnitude of Over-Requirement. Thus, we hypothesize that the number of Over-Required features included within scope increases in the absence of past relevant knowledge and in the case of software developers. Hence,

**H5:** Software engineers without past relevant knowledge of development times will plan more Over-Required features to be included within project scope than those with past relevant knowledge

**H6:** Software developers will plan more Over-Required features to be included within project scope than software consultants

Finally, we wish to draw a direct line between time underestimation and Over-Requirement. We wish to test the prediction that the underestimation of development times leads people to not only overestimate how much they can accomplish in a given period of time (Nouri, Jamali and Ghasemi, 2012) but also to include more Over-Required features within project scope. In other words, regardless of their past relevant knowledge and their role perspective, underestimating development times by software engineers increases the number of Over-Required features they include within scope. Hence,

**H7:** The more software engineers underestimate total development time the more Over-Required features they will plan to be included within project scope

The research model depicting the above hypotheses is presented in Figure 1.
Figure 1. Research Model

METHOD

An experiment was conducted to empirically test the research model depicted in Figure 1 and the seven hypotheses presented above. The experiment targeted advanced undergraduate Industrial Engineering & Management students majoring in the Information Systems while they took in the junior year the compulsory course “Automation and Computer-integrated Manufacturing”. By the time that this experiment took place they already experienced performing tasks similar to the task demanded of them in the experiment as part of their duties in this course and previous courses. As an incentive to participate in the experiment, each participant received an extra bonus point (1 out of 100) in the course grade. It is worth noting that engineering undergraduate students in Isreal, where the experiment took place, are older than their typical counterparts elsewhere in the world and most of them work part time while studying. In our experiment, 64% of the participants were 26 years old and above, yet only few had past work experience in industry. Since it is difficult to collect data from participants employed by industry when a controlled environment is required, relying on students as participants is acceptable in experiments exploring the behavior and the decision making of software developers and managers (Andres, 2001; Keil, Im and Mähring, 2007; Keil, Tan, Kwok-Kee, Saarinen and et al, 2000; Umapathy , Purao and Barton , 2008).

A pilot study with a sample of 15 students from the same population as in the experiment was performed to test the experimental design and manipulations. Some minor modifications were made according to feedback received from participants in the pilot study.

The main experiment was based on a factorial design of $2 \times 2$, manipulating two dichotomous independent variables: Past Relevant Knowledge (with or without knowledge regarding development times of similar features in the past) and Role (software developer or software consultant). Prior to randomly assigning the target sample to four groups, 10 participants were randomly excluded from the target population and assigned to a reference group that performed a side experiment in a parallel session without any manipulations. All other participants participated in the main experiment and were randomly assigned to one of four groups: developers with past relevant knowledge, developers without past relevant knowledge, consultants with past relevant knowledge, and consultants without past relevant knowledge. In the three-step main experiment, which took about half an hour, the manipulation of Role took place during the first step whereas the manipulation of Past Relevant Knowledge took place in the second step.

In the first step of the experiment, participants were told that they were employees of a big software company (The Company, hereinafter). Then, for manipulating the independent variable Role, half of the participants were randomly assigned to a developer role by being told that they were members of the development team in a development project and half were randomly assigned to a consultant role by being told that they were members of the consulting team in the same project. Then, a fictitious case was presented, according to which The Company is planning a software-development project for operating a robot whose purpose is to use blocks of different sizes, which arrive on a conveyor, to build three towers. Additional details as the sizes and locations of the towers were also provided, along with a list of 16 features of the planned software that the customer mentioned in early meetings with representatives of The Company. The feature list was deliberately composed of features of different importance for the project's goal as determined by two course instructors independently: five features on the list which both instructors considered unnecessary, were categorized as Over-Required, seven features as essential, and four as optional. In the second step of the main experiment, participants in a developer role were asked to estimate how long it would take them to develop each feature while, on the other hand, participants in a consultant role were asked how long it would take the developers to develop each feature. Then, for manipulating the independent variable Past Relevant Knowledge, half of the participants assigned earlier to each of the two roles were further randomly assigned to the group with (without) past knowledge by being informed (or not), for each feature, about the time it took to develop a similar feature in the past. Only in the third step of the main experiment participants were informed that the project duration is planned to be 18 hours. They were then asked to choose features to be included within the project scope, taking into consideration not only their earlier estimations of development times given while unaware of the upper bound on project duration but also the set bound of 18 hours.

For testing the different hypotheses, three dependent variables were calculated based on the answers of each participant in the main experiment: (a) Total Time Estimation, (b) Number of Features Included, and (c) Number of Over-Required Features Included. The Total Time Estimation summed all 16 time estimations for the development of features on the list. The Number of Features Included counted the number of features that the participant included
within project scope out of the full list of 16 features. The Number of Over-Required Features Included counted the number of features that the participant included within project scope out of the list of five features determined as unnecessary by both course instructors.

Participants in the reference group received a questionnaire with a similar introduction, presenting the project goal and the list of features. They were asked to express their opinion whether each feature was essential in one question, and whether each feature was unnecessary in another question. Their answers landed support to the list of unnecessary features, as determined by the course instructors. On average, 7 out of 10 members of the reference group categorized as unnecessary each of the five features determined unnecessary by course instructors, compared to only 1.3 out of 10 members of the reference group who, on average, categorized as unnecessary each of the other 11 features. The feature valuation by the reference group thus confirmed that participants in the four experiment groups, who belonged to the same target population, were by themselves capable of not only figuring out the importance of the features on the list but also of identifying Over-Required features.

RESULTS

Data collection yielded 85 gender-balanced participants, 75 in the main experiment and 10 in the reference group, and data analysis was pursued via analysis of variance (ANOVA) in SPSS. Overall, the mean of Total Time Estimation provided by participants was 23.03 hours (17.73 hours for those without past relevant knowledge and 28.18 hours for those with that knowledge), almost half of the total time of estimations we presented to participants as past relevant knowledge (43.50 hours). Since our experiment procedure uses homogenous groups with randomly assigned participants, these consistent differences across all 16 features provide evidence for treatment effectiveness since other possible explanations are controlled for. Furthermore, the mean of Number of Over-Required Features Included for all participants was 2.13, implying that participants included within scope almost half of the five unnecessary features. These findings served as a preliminary indication that the experiment was able to capture the Planning Fallacy and Over-Requirement in software development.

Table 1 shows the cell sizes for the four experimental conditions. Table 2, Table 3, and Table 4 show the estimated marginal means (and standard errors) of the three dependent variables. As can be seen in these tables, the group of consultants with Past Relevant Knowledge provided the highest mean of Total Time Estimation (30.95 hours), the lowest mean of Number of Features Included (8.42), and the lowest mean of Number of Over-Required Features Included (1.11). In contrast, the group of developers without Past Relevant Knowledge provided the lowest mean of Total Time Estimation (17.66 hours), the highest mean of Number of Features Included (13.28), and the highest mean of Number of Over-Required Features Included (3.61). Overall, the means of Total Time Estimation for developers and consultants without Past Relevant Knowledge were lower than for developers and consultants with Past Relevant Knowledge. At the same time, the Number of Features Included and the Number of Over-Required Features Included were higher. With and without Past Relevant Knowledge, as can be seen in Tables 3 and 4, the means of Number of Features Included and Number of Over-Required Features Included by developers were higher than by consultants.

<table>
<thead>
<tr>
<th>Role</th>
<th>Past Relevant Knowledge</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>With</td>
<td>Without</td>
</tr>
<tr>
<td>Developer</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>Consultant</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>Total</td>
<td>38</td>
<td>37</td>
</tr>
</tbody>
</table>

Table 1. Cell Sizes of Experimental Conditions

<table>
<thead>
<tr>
<th>Role</th>
<th>Past Relevant Knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>With</td>
</tr>
<tr>
<td>Developer</td>
<td>25.41 (2.37)</td>
</tr>
<tr>
<td>Consultant</td>
<td>30.95 (4.53)</td>
</tr>
</tbody>
</table>

Table 2. Marginal Means (Standard Errors) for Total Time Estimation in Hours
Table 3. Marginal Means (Standard Errors) for Number of Features Included

<table>
<thead>
<tr>
<th>Role</th>
<th>Past Relevant Knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>With (Standard Error)</td>
</tr>
<tr>
<td>Developer</td>
<td>11.21 (0.72)</td>
</tr>
<tr>
<td>Consultant</td>
<td>8.42 (0.65)</td>
</tr>
</tbody>
</table>

Table 4. Marginal Means (Standard Errors) for Number of Over-Required Features Included

<table>
<thead>
<tr>
<th>Role</th>
<th>Past Relevant Knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>With (Standard Error)</td>
</tr>
<tr>
<td>Developer</td>
<td>2.37 (0.40)</td>
</tr>
<tr>
<td>Consultant</td>
<td>1.11 (0.30)</td>
</tr>
</tbody>
</table>

Table 5 presents the ANOVA results for the full factorial models for the three dependent variables, aimed at testing H1 through H6. As can be seen in the first column of Table 5, there is a main effect of Past Relevant Knowledge on Total Time Estimation ($F=10.933, p<0.01$), supporting H1, while H2 is not supported because there is no main effect of Role ($F=0.805, p=0.373$). Testing H3 and H4, concerning the impact of the two independent variables on the Number of Features Included (second column in Table 5), reveals significant effects of both Past Relevant Knowledge ($F=10.754, p<0.01$) and Role ($F=15.225, p<0.001$), respectively supporting H3 and H4. The third column in Table 5 presents two more significant main effects, those of Past Relevant Knowledge ($F=4.971, p<0.05$) and Role ($F=20.129, p<0.001$) on the Number of Over-Required Features Included, respectively supporting H5 and H6. All the differences between the two conditions of Past Relevant Knowledge (with or without) and the two conditions of Role (developer or consultant) for all three dependent variables were in the hypothesized direction.

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Total Time Estimation</th>
<th>Number of Features Included</th>
<th>Number of Over-Required Features Included</th>
</tr>
</thead>
<tbody>
<tr>
<td>Past Relevant Knowledge</td>
<td>10.933 **</td>
<td>10.754 **</td>
<td>4.971 *</td>
</tr>
<tr>
<td>Role</td>
<td>0.805</td>
<td>15.225 ***</td>
<td>20.129 ***</td>
</tr>
<tr>
<td>Role × Past Relevant Knowledge</td>
<td>0.728</td>
<td>0.050</td>
<td>1.212</td>
</tr>
<tr>
<td>$R^2$</td>
<td>0.150</td>
<td>0.266</td>
<td>0.267</td>
</tr>
<tr>
<td>Adjusted $R^2$</td>
<td>0.114</td>
<td>0.235</td>
<td>0.236</td>
</tr>
</tbody>
</table>

Table 5. ANOVA Results (F Values are Shown; * $p<0.05$, ** $p<0.01$, *** $p<0.001$)

Figure 2 graphically depicts these differences for the Number of Over-Required Features Included, confirming the main effects on over-requirement. Concerning H7, a Pearson correlation test revealed a significant negative correlation between Total Time Estimation and the Number of Over-Required Features Included ($r=-0.398, p<0.01$). As hypothesized, therefore, the lower Total Time Estimation is, the more Over-Required features are included within scope.
DISCUSSION AND CONCLUSION

The findings of the experiment confirm findings of past studies that the Planning Fallacy plays a role in software development. Moreover, this study finds a relationship between the Planning Fallacy and Over-Requirement. Due to the Planning Fallacy, software developers and consultants alike tend to underestimate the time required to develop software features. Underestimation of software-development time can lead, on one hand, to optimistic time planning for the entire development project and, on the other hand, to Over-Requirement, i.e., planning more features within project scope than can be accomplished by the deadline (Nouri et al., 2012), including Over-Required features.

The effects of past relevant knowledge on the estimation of development time (H1), on the number of features included within project scope (H3), and on the number of Over-Required features included within project scope (H5) have been found significant. As part of their course duties, all participants had the experience of developing similar features. Yet, only half the participants composing two of the four groups were deliberately presented with past development times as relevant to the tasks at hand to ensure that this Past Relevant Knowledge condition holds (Griffin and Buehler, 2005). Having explicit knowledge of past development times seems to reduce time underestimation as well as scope overloading and Over-Requirement. The results thus demonstrate the positive influence of past relevant knowledge and experience on time estimations of software development and on project scoping. The results show, however, that underestimation is reduced but not eliminated by past relevant knowledge.

The role, whether developer or consultant, has been found to have a significant influence on the number of features included within scope, including Over-Required features (H4 and H6 were supported). Yet, there is no significant difference in time estimations provided by developers and consultants (i.e., H2 was not supported). Thus, despite the finding that developers do not provide significantly lower estimations (for tasks to be developed by them) than consultants (for tasks to be developed by others), developers tend to include more features and more Over-Required features within project scope than consultants. These significant differences between developers and consultants may imply other behavioral explanations for the Over-Requirement phenomenon beyond the Planning Fallacy. One
possible explanation might be the endowment effect, due to which ownership feelings that might be elevated when holding a developer's perspective, as opposed to a consultant's perspective, lead to regarding the features under development as more valuable than they really are (Thaler, 1980). Another explanation might be the IKEA effect, due to which feelings of attachment toward the features under development are gained, as a result of cognitive effort invested by developers more intensely than by consultants in analyzing and estimating the time needed to develop each feature (Norton, Mochon and Ariely, 2012). Both explanations rely on the assumption that since consultants are less involved, they are less prone to behavioral biases associated with ownership (endowment) or effort (IKEA). Nevertheless, despite holding a third-person perspective, which the Planning Fallacy regards as less biased subjects, their lesser involvement does not seem to impact the consultants' time estimations. It may very well be that the experimental manipulations failed to cause participants in the consultant role to be completely uninvolved when time estimations were requested since, for the sake of accurate estimations and for the sake of mutual baseline with developers, they were specifically asked to assess the time needed for their "friends in the development team who joined The Company together with them".

Finally, the last hypothesis (H7) was supported, demonstrating the negative relationship between the estimation of development time and the number of Over-Required features included within scope, in the sense that the lower time estimations are, the more Over-Requirement occurs. Since the experiment steps were ordered in a way that subjects were asked to determine the project's scope only after they estimated feature completion times, this finding confirms the impact of the Planning Fallacy on Over-Requirement.

The findings of this study are relevant to both practitioners and researchers. Developers and consultants in software development projects should strive, to the extent possible, to base their estimations on the outcomes of past projects of their own and of others. They should resist the inclination to consider each software development project as a unique endeavor and, instead, seek to find relevance in previous projects, which can then serve as a baseline for estimations. In addition, managers of software development projects should be aware that developers are more biased than consultants regarding the inclusion of Over-Required features in scope and that the underestimation of development time is at the root of this phenomenon.

From a research perspective, making the connection between the Planning Fallacy and the phenomenon of Over-Requirement in software development projects opens new directions for future research regarding other behavioral effects that might be relevant to Over-Requirement, such as the endowment and IKEA effects (Norton et al., 2012; Thaler, 1980). Future research might also address the two primary limitations of this study. First, although participants were advanced undergraduate students with the relevant knowledge for the task, most of them did not have real development experience in industry. Second, the experiment took about half an hour and tried to emulate reality concerning time-estimation and project-scoping tasks and the roles of developer and consultant. Yet, it is seldom possible in an experiment to account for such aspects as human interactions, organizational culture, and maturity. To address both limitations, further research can target full-time software developers or software consultants at the phase of project planning and ask them relevant questions about feature development time and features to be included in scope. Notwithstanding its limitations, this study demonstrates the implications of the Planning Fallacy for software development projects in general and for Over-Requirement in particular. Our findings confirm the notion that the lens of behavioral effects has the potential to advance the understanding of software development processes.

REFERENCES


The Structures of Computation: A Distributed Cognitive Analysis of Requirements Evolution in Diverse Software Development Environments

Sean Hansen  
Rochester Institute of Technology  
shansen@saunders.rit.edu

Amol Kharabe  
Ohio University  
kharabe@ohio.edu

Kalle Lyytinen  
Case Western Reserve University  
kalle@case.edu

ABSTRACT

Despite decades of research, effective requirements engineering (RE) remains a significant challenge for software development projects. In addition, the study of RE processes has failed to keep pace with dramatic changes in IS development practice. In this study, we frame RE as a fundamentally socio-technical computational task in which diverse social actors and artifacts collaboratively “compute” the requirements for an envisioned software system. Through the perspective of distributed cognition, we analyze the distributed RE activities of three IS development projects, representing distinct methodological approaches. We develop models of the computational structures of these projects to support a novel analytical basis for comparison and contrast of three development methodologies—structured development, agile development, and open source software development.

Keywords

Requirements engineering, distributed cognition, structured development, agile development, open source development.

INTRODUCTION

Requirements engineering (RE) refers to processes by which information systems (IS) designers discover, specify, validate, and manage the functionality that an envisioned system is expected to possess, as well as the constraints to which it must conform (Kotonya and Sommerville 1998). Despite more than three decades of research in the RE domain, requirements-oriented issues remain a primary source of IS project distress (Aurum and Wohlin 2005; Crowston and Kammerer 1998; Hickey and Davis 2003; van Lamsweerde 2000). This persistence of requirements-based challenges has been complicated by the rapidly changing nature of RE practice and IS design (Jarke et al. 2011). One of the most notable trends in this regard is that RE activities are increasingly distributed across locations, organizations, participants, and time (Hansen et al. 2009). Such broad distribution poses new challenges and presents a stark contrast to the traditional view of RE, in which a small group specifies a system in relative isolation prior to downstream development and implementation (Jarke and Pohl 1994). In addition, the software development landscape has experienced an explosion of new methods and technologies including the rise of commercial off-the-shelf (COTS) and framework-based development (e.g., SOA), agile development methods, and open source software development (OSSD). Interestingly, some of these new methods (e.g., agile development) have emerged specifically in reaction to the perceived shortcomings of older RE approaches. However, despite these novel approaches, the fundamental challenge of deciding the design needs of an artifact has continued to persist (Cao and Ramesh 2008).

In the wake of these multi-faceted changes, RE research has struggled to keep pace (Hansen et al. 2009; Jarke et al. 2011; Kaindl et al. 2002). While some research has focused on RE in geographically distributed teams (e.g., Damian et al. 2003; Grünbacher and Braunsberger 2003), or comparisons of RE processes across different development paradigms (Gacek and Arief 2004; Scacchi 2009), the extant literature includes relatively little exploration of the diverse ways in which RE is simultaneously distributed across space, organizations, artifacts, and time. While RE research maintains a strong design science tradition (e.g., multiple prescribed tools and techniques), the field has
been marked by a relative dearth of theoretical foundations and theory development (Jarke et al. 1993). Consequently, we know little about how a change in distribution of RE parameters affects RE activities and their outcomes (Jarke et al. 2011).

In the current research, we seek to address this gap both theoretically, by developing cognitive models of RE computation within distributed projects, and empirically by using rich field study data to conduct rigorous analyses of task flows in multiple development environments. Specifically, we adopt a cognitive perspective which focuses on mechanisms by which participants in the RE undertaking process information to arrive at a commonly held understanding of the needs and constraints of stakeholders and users. With reference to Cheng and Atlee’s (2009) widely cited roadmap of RE research, we assert that the adoption of a cognitive perspective on RE tasks incorporates two research strategies - leveraging other disciplines and evaluation. Furthermore, the research is intended to address the “RE research hotspot” (Cheng and Atlee 2009) of addressing emerging needs.

From a cognitive perspective, we frame RE as a fundamentally socio-technical computational task. That is, a development project is a socio-technical cognitive system, composed of diverse social actors and artifacts which collaboratively work to “compute” the requirements for an envisioned software system (Hansen et al. 2012). In this view, we understand requirements to be computed through various computational structures (i.e., processes for exchange between elements of a cognitive system), which may be expected to vary across different development methods (e.g., traditional structured development vs. open source software development). Further, we argue that RE-oriented computations can be modeled as variations in cognitive workflows, which, when successful, achieve requirements closure—i.e., a state where requirements are viewed as clear, agreed upon, and implementable. Finally, we also anticipate that, given the diversity of configurations, computational structures may vary in their effectiveness at achieving such a closure for a given design task. Thus, we seek to address the following key research questions:

- What are the characteristic modes of social, structural, and temporal distribution of RE activities in different development environments?
- How do computational structures vary across different development environments?

To address these questions, we conducted a multi-site field study of software development projects in three distinct development environments – structured development, agile development, and open sourced software development. In analyzing these project environments, we draw upon the Theory of Distributed Cognition (Hollan et al. 2000; Hutchins 1995a) to identify and model the characteristic modes of cognitive distribution within these complex work environments.

BACKGROUND AND THEORETICAL FOUNDATIONS

Requirements Engineering

Whether or not it is explicitly acknowledged by designers, RE represents a fundamental facet of any software development effort wherein several foundational questions are addressed: What do we want to create? What features and functionality should be developed? What practical objectives are we trying to address? Frameworks within the research literature have divided RE into anywhere from two to seven distinct cognitive tasks (Dorffman 1997). For the present discussion, we consider three commonly recognized RE tasks: (1) discovery, (2) specification, and (3) negotiation and prioritization, as they represent distinct and necessary cognitive outcomes for a successful computation of design requirements. RE efforts must address the ways in which software development teams organize, manage, and coordinate these three activities to achieve desired RE outcomes.

- **Discovery**: The process by which designers identify the organizational, individual, technological, or other needs that must be met by the envisioned software (Kotonya and Sommerville 1998; Loucopoulos and Karakostas 1995; Maiden 2008; Piller et al. 2004).

- **Specification**: The rendering of discovered requirements into a representational form, so that the knowledge can be validated for correctness, clarity, feasibility, and coherence, and thereafter used in downstream design activities (Borgida et al. 1985; Mylopoulos 1998; van Lamsweerde 2000). Specification forms the point of transition where articulated needs are extended with functional and technical design implications.
Negotiation & Prioritization: The identification and resolution of conflicts through exploration of the range of design possibilities – generating knowledge of design options and their consequences (Feather et al. 1997; Grünbacher and Seyff 2005; Robinson and Volkov 1998) often through the use of negotiation models (Fisher and Ury 1991). Designers must reconcile requirements due to the presence of disagreements between stakeholders (Grünbacher and Seyff 2005; Robinson et al. 2003), and prioritize the resulting requirements to determine the relative value of individual design requirements (Berander and Andrews 2005; Regnell et al. 2001).

From a cognitive perspective, all software design efforts must address the overall coordination and integration of the three tasks. However, the identification of these tasks reflects no assumptions of their sequencing, executors, means, local goals, or spatial organization. Indeed, as alluded to above, in some development environments (e.g., agile development), designers might demure at the use of the terms we propose vis-à-vis the widely-accepted language of RE. Nevertheless, we contend that these activities represent essential cognitive tasks inherent in any design process.

Distributed Cognition

Distributed cognition is a branch of cognitive science that argues that cognitive processes, such as memory, decision making, and reasoning, are not bounded by the internal mental states of an individual (Hollan et al. 2000; Hutchins 1995a; Hutchins 1995b; Magnus 2007). The development of the theory was motivated by research on teams engaged in complex, multi-person tasks. In these settings information processing is not limited to individual actors; rather, it is distributed across members of a group. Furthermore, a significant portion of the cognitive load in these environments is borne by the artifacts used by group members.

By framing cognition as “the propagation of representational state across representational media” (Hutchins 1995a: p. 118), distributed cognition extends the unit of analysis for cognitive activity from the individual to the entire group in accomplishing a given task. With this fundamental shift in perspective, the theory develops three critical assertions (Hutchins 2000) - 1) cognitive processes are distributed among members of social groups (which we label social distribution), 2) cognition employs both internal and external structures (structural distribution), and 3) cognitive processes are distributed over time (temporal distribution).

Social Distribution. Distributed cognition theory contends that cognitive processes are distributed across members of a group. Each individual plays a specific role with respect to the information processing and action of the broader group. This idea has obvious ramifications for the study of distributed RE processes. The vast majority of software design efforts employ a team structure (Guinan et al. 1998). Furthermore, an essential characteristic of development teams is the diversity of knowledge required (Levina and Vaast 2005; Walz et al. 1993).

Structural Distribution. The second implication of distributed cognition is that cognitive processes entail the intertwining of internal and external structure. While traditional cognitive science focuses on the internal states of an individual’s mind, a distributed cognitive perspective highlights the ways in which human actors integrate material elements as part of their cognitive processes. The offloading of cognitive demands onto external structure is readily apparent in prevailing RE practice. The heavy emphasis on formal modeling in RE can be understood as a mechanism for creating external structures that support and enhance the human cognitive processes of design.

Temporal Distribution. Finally, distributed cognition theory contends that cognitive processes may be distributed with respect to time. Earlier decisions and actions influence cognitive processes enacted later. Temporal distribution is present in any context where heuristics are used to guide decision making. Design efforts draw heavily upon requirements and artifacts inherited from earlier projects.

RESEARCH DESIGN

Multi-Site Field Study

In this study, we conduct a multi-site field study of ongoing systems development projects. This multi-case approach enables us to engage in a rich exploration of the sociotechnical, cognitive process of practicing IS development (ISD) professionals (Eisenhardt 1989; Yin 2003). The unit of analysis for the study is the individual ISD project. Specifically, our analysis focused on ISD projects employing three distinct development approaches: 1) structured, platform-based development, 2) agile software development, and 3) OSSD. The site inquiries were conducted in accordance with prevailing case study field procedures, including the development of a case study protocol prior to
data collection, triangulation using multiple sources of evidence, and the maintenance of a chain of evidence (Yin 2003).

Prior to data collection efforts, we collaboratively developed a case study protocol, which included the research questions guiding the inquiry, the essential design of the study (i.e., multiple case analysis within varied development environments), the bases for case selection, the modes of data collection to be executed (i.e., interviewing, direct observation where applicable, documentary review), and agreed upon data collection procedures. This broader protocol document also included a detailed interview protocol to guide our conversations with respondents. The interview protocol incorporated inquiry into individual and organizational backgrounds, the structure of ISD teams/units, ISD environments and practices, specific techniques used for identifying and capturing design requirements, specific technologies or artifacts used by project members, and perceived challenges to effective requirements determination. Within each of these areas, the protocol incorporated multiple contingent probes for additional discussion. Given the space restrictions of the current paper, we have not attached the protocol document, but it is available to interested researchers upon request.

**Summary of cases**

The following provides a brief introduction to each of the projects studied. To adhere to assurances of confidentiality, we have developed pseudonyms for the first two organizations:

*The SIS Project.* The development efforts which we have dubbed the SIS Project focused on the acquisition, customization, and implementation of a vendor-sourced Student Information System (SIS) ERP at a mid-sized Midwestern U.S. university. The SIS platform was intended to integrate all student information and student-facing administrative functions across the university’s nine schools. The project team employed a structured development method with explicit documentation and discussion of design requirements, and a thorough review and sign-off process for any proposed changes to the COTS platform. The university engaged multiple consultants to work on the project. The consultants included a large team from a firm which specialized in enterprise system implementations within higher education. Other consultants were employed for their specialized technical skills. Finally, the project drew significant requirements knowledge from engagement with a web-based user group, called the Higher Education User Group (HEUG), which comprised of other universities who had previously worked on the software platform being implemented.

*The BigD Project.* SocialAgg is a social media aggregator that develops software for both internal and external clients. In particular, their software is focused on the advanced analysis of large amounts of web content information (i.e., Big Data). SocialAgg development teams employ an agile methodology that is a variation on Scrum. While they are very committed to the agile development philosophy, they are not dogmatic in their application of the Scrum methodology. The development teams operate on a three-week sprint cycle, conduct stand-up meetings two or three times weekly (rather than daily), and do not use a formal agile task board. The firm relies upon Amazon’s cloud services for all data storage and maintains their software in a GitHub repository. The specific project that we studied, labeled BigD, focused on the development of a data framework and query tool that enabled the firm to draw business intelligence for the firm and its clients through algorithmic analysis of massive social media data feeds.

*The Rubinius Project.* Rubinius is an OSSD project focusing on the development of a virtual machine (VM) and related compiler for the Ruby programming language. The project is hosted on GitHub, a web-based hosting platform for OSS projects which is based on Git version control system. Rubinius is a partially-sponsored OSSD project. In 2007, two years after the project was initiated, the Engine Yard Company began to sponsor several committers of Rubinius to work fulltime on the project. Engine Yard is one of the largest privately-held firms focused on Ruby on Rails and PHP development. The first fully-functional version of Rubinius was released in 2010 and the project has since been working on a 2.0 release. In keeping with the pattern observed in most OSSD projects (Crowston et al. 2006), Rubinius has a small set of core developers and much larger set of peripheral committers who work on the project on a purely voluntary basis. Not surprisingly, in light of the OSS nature of the project, Rubinius does not employ formalized RE processes. However, the cognitive tasks of RE are accomplished through community discussions and activities mediated through a variety of requirements “informalisms” (Scacchi 2009), including emails communications, developer forums, and internet relay chat (IRC) channels (Gacek and Arief 2004).
Data Collection and Analysis

On initial engagement with each of the projects, we performed an in-depth, semi-structured interview with a key respondent with oversight authority for the relevant project. Relevant areas of focus in the interviews included identification and analysis of processes employed to capture project requirements; the allocation and coordination of RE tasks among participants; artifacts used to capture, share, monitor, and communicate requirements; the flow of requirements knowledge among project roles and artifacts; and the protocols to maintain requirements priorities over the life of the project. Based on the recommendations of the initial respondents at each site, additional interview respondents were identified. To ensure multiple perspectives on the software development processes employed on each project, we sought participation from development leads and individual developers. In addition, where applicable, we sought participation of project sponsors or executive stakeholders. All of the interviews were conducted using the aforementioned interview protocol. In all, we interviewed 24 subjects, with the typical duration of each interview being approximately 60 minutes, although the interviews ranged from 45 to 90 minutes. Our interviews were augmented with a review of artifacts that supported the interaction of project participants (e.g., project documentation, requirements specifications, and communication media). In some cases (e.g., not applicable in the OSSD context), we conducted direct observation of participants to assess how different people work and use requirements knowledge within these projects.

In accordance with a grounded analytical approach, the research team began the coding process concurrently with data collection activities. Specifically, our data coding employed a thematic analysis of the case data (Boyatzis 1998). While the thematic analysis was be conducted in line with key principles of grounded theory methodology (Glaser and Strauss 1967; Strauss and Corbin 1990), such as constant comparison and open, axial, and selective coding, it differed from a pure grounded theory approach in that the analysis was informed by the framework drawn from theories of distributed cognition. All data was coded and parsed into categories to identify important themes in the RE practices of the projects. The coding was conducted with online, collaborative coding software, called Dedoose. The coding processes addressed two distinct axes of analysis: (1) RE tasks reflected, and (2) modes of distribution. The analysis of RE tasks focused on the processes through which the projects achieved the ends of discovery, specification, negotiation/prioritization, and monitoring. This analysis led to the identification of the computation structures, or primary cognitive workflows, of a project. The second analytical axis was the forms of cognitive distribution reflected within the workflows and tasks. Thus, we coded the distinct social, structural, and temporal distribution mechanisms employed by project teams. Combining the output of these two analytical lenses, we determined forms of distribution within and across tasks as they evolved through discovery, specification, negotiation, and monitoring.

FINDINGS

The key findings of this research can be categorized into two areas as below – a) cognitive distribution of the requirements along multiple axes, and b) computational structures observed for the distributed cognition of such requirements.

Cognitive Distribution

The three projects reveal very different patterns with respect to the ways in which cognitive effort is distributed along the social, structural, and temporal axes. We highlight the distinguishing features of each environment along each of these dimensions.

Social Distribution. The analysis of social distribution mechanisms in the three projects revealed a number of critical dimensions for comparison, including the degree of specialization, the modes of communication between the social actors, and the integration of third-party insights. First, the degree of specialization clearly varied across the cases. The SIS project involves significant specialization in both technical and functional areas:

“I think [the Technical Lead’s] got that laid out. He’s got one developer that tends to do more of the admissions stuff, one in the student financials ... So he has segregated duties.” – SIS Technical Consultant

In contrast, the BigD project reflects limited specialization in keeping with an agile ideal of broad-based and flexible skill sets for developers. On the BigD project, and indeed throughout SocialAgg’s development teams, work is executed in very small collaborative teams:
“My style is to break [the work] down into smaller manageable chunks that can be achieved in the quarter, by no more than two people— one to two people. We call this a one-pizza team, or a two-pizza team, and the basic idea is the team size was just enough to feed them all on either one box of pizza or two boxes of pizza.” – BigD Project Director

Finally, because of the voluntary nature of OSSD work, the Rubinius project has specialization on a self-selected basis. However, despite the significant size of the community, our analysis revealed actual specialization to be relatively low in that most members have similar skill sets and professional backgrounds.

In addition to the degree of specialization observed, we see broad variance across the three projects with respect to communication between stakeholders. The one-pizza teams of BigD emphasize face-to-face communication reinforced through collocation.

“...I think the war room helps a lot, especially when talking about [issues]... When you’re sitting next to one another, you get a lot of information” – BigD Project Manager

The SIS project instead opts to discuss all requirements changes through a structured walkthrough process. The walkthroughs are conducted by the project’s Communications Lead and attended by most project team members. The process is very document-centric, focusing on formal specifications generated by the project’s Functional Lead or Consultants (see “Structural Distribution” below).

“I think the walkthroughs were really a good idea. It got people around a table. Some of them were pretty [contentious]... but I think it was what needed to happen.” – SIS Communications Lead

In the Rubinius case, almost all communication is computer mediated. In particular, the core and peripheral developers, as well members of external communities using the Rubinius platform (e.g., Ruby and Ruby on Rails developers, Travis CI users, Puma users), communicate with one another through ‘informalisms’ such as IRC chats, emails, and community posts:

“We were on the IRC channel one day and [the concurrency stuff] kind of came as a dare, like ‘I bet you can’t add concurrency’. It wasn’t a direct dare like that but it was basically like, it got me. We were talking about it and it got me thinking and I was like ‘Well, I’ll just sort of do it as a spike and see how far can I get in the shortest amount of time” – Rubinius Developer

With respect to third-party sources, each of the three projects draws requirements knowledge from sources external to the project, but the forms are quite different across the three contexts. In the SIS environment, the heavy reliance upon consultants and the HEUG community provides access to lessons learned at other institutions. The following statement illustrates this dynamic process for integrating externally-generated insights:

“I was encouraging our functional consultants to use [the HEUG community] and they did... [One challenge] I dealt with I said, ‘Put the question out to the user community. Let’s find out what they did with this.’ So it’s a very useful thing. I think it should be used a lot.” – SIS Technical Consultant

For BigD, the project benefitted from SocialAgg’s explicit focus on the identification of insights or possible avenues for future development from the broader social media and data analytics communities. SocialAgg maintains an engineering unit specifically focused on technology scanning and research and development activities. The role of this group is to call the attention of the firm’s executive leadership to promising and innovative developments in the marketplace and to channel such insights to the individual project teams.

“What I mean by ‘front end’ is not user interface, but it’s a leading edge group that is looking at ideas six months, nine months down the line that I could eventual productize, but I need to give them time to think through that stuff and work on it. So it’s a small group of three people, but it’s a science group and it’s applied research. They test ideas.” – BigD Project Director

Finally, the Rubinius project draws heavily on insights from communities outside the focal Rubinius community. As noted above, these external communities include other OSS developers and users that focus on distinct platforms that leverage the Rubinius compiler (e.g., Ruby, Ruby on Rails, Travis CI). The Rubinius developers take insights
from this broader community of Ruby-based projects, as well as from developer conferences to identify possible enhancements worth pursuing:

“RubyConf attracts developers, more experienced and knowledgeable developers, so it’s a good opportunity to talk to people about a lot of very technical aspects of Ruby and Rubinius, so I do. I talk to a lot of people.” – Rubinius Developer

**Structural Distribution.** Similar to the social distribution, the analysis of structural distribution revealed some relevant bases for comparison, including the role of documentation, the use of graphical representations, and the reliance upon system-embedded requirements. Perhaps the most obvious point of contrast between the three environments centers on their approaches to requirements documentation. The SIS project favors formal documentation. All requirements are written up in formal specifications with a standardized format.

“The primary mechanism of communication between my folks and the line-of-business folks is that spec document. Then there [are] other documents that get handed off. There’s a tracking document that’s sort of like the high level tracking document above the spec.” – SIS Technical Lead

In keeping with the agile philosophy, the BigD project adopts a lightweight approach, with very limited formal documentation of requirements. The team does create user stories, but these are informal in nature and the team explicitly eschews a document-intensive approach in favor of “just doing what makes sense in a given case.”

Similar to BigD, the Rubinius project does not use standardized documentation, the majority of design requirements are embodied in the ‘informalisms’ of email and IRC:

“We did coordinate pretty much exclusively on the IRC channel... [T]he actual discussions mostly took place on IRC. I think we had a couple of conference calls maybe, but mostly IRC, and just told people what we were doing, gonna be doing.” – Rubinius Developer

With respect to the use of graphical representations, we see some similarity between SIS and BigD projects. Both of these projects make extensive use of mock-ups and white board drawings. However, the emphasis on this form of representation as a primary mode of exchange and requirements capture is greater in the BigD context.

“So as the requirements become clearer for some of these larger projects, sprint stories will be entered into FogBugz [the feature tracking tool] against the project, both by the developers and the Project Managers... It’s sort of lightweight – it may include mocks, flows, pictures, etc., and so they just get attached to that.” – BigD Project Director

In Rubinius, we found little, if any, evidence of graphical representation – the ideas outlined in text-based ‘informalisms’ get converted right into code by individual developers. This may be due to the fundamental distribution of the team members across both spatial and temporal dimensions, and the inherent limitations imposed by such distributions.

**Temporal Distribution.** In the area of temporal distribution, we see differences between the three projects with respect to the importance of iteration and the role of “inherited” requirements. The emphasis on iteration is greatest in the BigD project, where the sprint cycles reinforce an evolutionary approach to the development. Each sprint builds upon the affordances created in the previous sprint, while remaining open to changes based on new requirements flowing from the project sponsors:

“The quarter is divided into three-week sprints, so there’s typically going to be about four sprints in every quarter. Then in each sprint, the team that is assigned to that sprint of the product line will decide what they want to achieve. So it’s a stepwise breakdown from the yearly strategic arc to the quarterly to the three-week sprints.” – BigD Project Director

Iteration is similarly critical in the Rubinius project as the platform is incrementally enhanced by the diverse developer community. As is common in OSS environments, the platform is considered to be in a constant state of evolution. This iteration of the design is a fundamental guiding principle.
In the SIS project, iteration is relatively limited. In keeping with a structured development approach, the team is modifying the system for a full-scale roll-out to the university, rather than adopting an evolutionary build process. Interestingly, the project’s Technical Lead stated that the limitation of changes to the underlying vendor platform was a primary focus for the development effort:

“I would say I should be trying not to do much systems development. I sort of judge how well we do by limiting that.” – SIS Technical Lead

Finally, with respect to “inherited” requirements (i.e., requirements embedded within the specific systems or tools adopted), in the cases of SIS and Rubinius, the teams are highly reliant upon requirements embedded within the systems they employ. For example, in SIS, a large number of requirements are established a priori based on the capabilities of the vendor platform and interfacing needs of the university’s existing systems:

“Basically what you find is the old system had certain structures and the new system has certain structure. If they’re going to take full advantage of the new system, they have to redefine things to meet the new structure. For example in the legacy system, you can only have three programs of study ... and then in PeopleSoft world, that’s normalized ... You can imagine other things like that where you wouldn’t be able to take the data in the old format and put it into the new one.” – SIS Technical Consultant

Similarly, in the case of Rubinius, a significant source of ‘inherited’ requirements was driven by compatibility with existing standards and other standards-based products in the Ruby milieu. In some cases, the project team focused on reverse-engineering from the related systems:

“MRI is the de facto standard. You know there is no language standard other than what MRI does...A lot of Rubinius work is actually basically reverse engineering what MRI does, figuring out its behavior.” – Rubinius Developer

In contrast to the SIS and Rubinius projects, the BigD developers have not inherited extensive requirements with the tools used in their development. This difference undoubtedly relates to the fact that the BigD project is focused on a fundamentally new tool while both SIS and Rubinius are modifying an existing platform.

A summary of comparisons of the three projects with respect to their social, structural, and temporal modes of distribution is provided in Table 1.

<table>
<thead>
<tr>
<th>Modes of Distribution</th>
<th>Projects</th>
<th>SIS</th>
<th>BigD</th>
<th>Rubinius</th>
</tr>
</thead>
<tbody>
<tr>
<td>Social</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specialization</td>
<td>High</td>
<td>Low</td>
<td>Limited; self-selection</td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td>Document-mediated; Structured walkthroughs</td>
<td>Face-to-face/co-location</td>
<td>Computer-mediated; Informalisms</td>
<td></td>
</tr>
<tr>
<td>Third-Partyies</td>
<td>Consultants; HEUG</td>
<td>Technology scanning</td>
<td>Ruby communities</td>
<td></td>
</tr>
<tr>
<td>Structural</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td>High/Formal</td>
<td>Low</td>
<td>Informalisms; Code base</td>
<td></td>
</tr>
<tr>
<td>Graphical</td>
<td>Mock-ups; Whiteboards</td>
<td>Mock-ups; Whiteboards</td>
<td>Limited</td>
<td></td>
</tr>
<tr>
<td>Temporal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iteration</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Inherited requirements</td>
<td>Extensive</td>
<td>Limited</td>
<td>Extensive</td>
<td></td>
</tr>
</tbody>
</table>

Computational structures
In distributed cognition theory, a significant emphasis is placed on the computational structure of a given cognitive system. The computational structure is the sequence by which information is processed through the propagation of representational state across representational media (Hollan et al. 2000; Hutchins 1995b) – steps that move the cognitive system from raw data to a desirable solution. Within a given work environment, the computational structure can be either explicit (e.g., a documented standard operating procedure) or implicitly understood by the social actors within the cognitive system (Al-Rawas and Easterbrook 1996).

When considering the distributed cognitive aspects of requirements-oriented processes, we must identify the computational structure of the various systems we analyze. Accordingly, in this section of our findings, we model the computational structures of the three cases to support comparison and contrast. Figure 1 through 3 provide graphical representations of the computational structures for the SIS, BigD, and Rubinius projects, respectively. In these models, we have attempted to articulate (1) the modes of social, structural, and temporal distribution reflected in each environment and (2) how these distribution mechanisms relate to the essential cognitive objectives of anRE process – i.e., discovery, specification, and validation & verification (V&V) of design requirements.

Figure 1. SIS Model: A Structure Development Computational Structure
Distributed Cognitive Analysis of Requirements Evolution in Diverse Environments

Figure 2. BigD Model: An Agile Development Computational Structure

Figure 3. Rubinius Model: An OSSD Computational Structure
Looking beyond the superficial differences between the three computational structures (e.g., idiosyncratic processes and labels), we see some interesting and significant basis of comparison and contrast between the three project environments. First, with respect to comparison, all three of the environments have clear mechanisms through which the objectives of requirements discovery, specification, and V&V are achieved. This is significant since in two of the three environments (i.e., agile→BigD, OSSD→Rubinius) the traditional language of requirements engineering is generally disregarded or devalued. Yet, our analyses illustrate that the objectives traditionally associated with RE (i.e., identifying new requirements, articulating them in a concrete and communicable form, and obtaining feedback on their execution) remain relevant and fundamental to the software development undertaking, even if the more established way of describing those aspects are discarded. Secondly, we see that all three environments involve significant interplay between diverse social actors or stakeholder groups (social distribution) and extensive integration of external representations (structural distribution) to move the system toward effective computation of design requirements. This suggests that the theoretical perspective of distributed cognition provides some useful insight for analyzing diverse software development contexts.

Turning to the points of contrast between the three environments, a number of features are readily apparent. First, the relatively linear nature of the SIS structured development process stands in marked contrast to the other two contexts. While the fact that a structured development effort is more linear than an agile or OSSD process may not be surprising, the computational structure also reveals that this process allows for significantly less feedback and broader community engagement, as the design and development progresses. It similarly reveals that the prominence of gap analysis (i.e., between the acquired platform and specific user requirements) as a primary focus for requirements activity in the structured environment significantly bounds (and furthermore binds) the options for system development. Second, we see that agile development environment (i.e., BigD) stands out for the extensive overlap of discovery, specification, and V&V cognitive processes. Nearly every facet of the process involves some elements of these three fundamental objectives. Perhaps tied in with this observation, the obvious emphasis on iteration is readily apparent in the agile computational structure. Finally, while iteration is also prominent in the OSSD context, the most notable feature of this environment is the role of structural distribution. All interactions between social actors in the Rubinius project are mediated by external structure.

While several of the findings regarding differences between the computational structures of these three development environments are not surprising given a familiarity with the methodologies espoused, the broader insight is that the modeling of the computational structures provides a novel basis for analysis of these environments and methodologies from a cognitive perspective.

DISCUSSION

The research in this paper started out with the dual observations that a) despite more than three decades of requirements engineering (RE) research, RE project phases are a prime source of issues, gaps and failures in information systems development projects, and b) such RE related activities have become increasingly distributed across geographies, organizations, time and project methodologies. To understand such RE related challenges, we leveraged a theoretical model based on distributed cognition, for use as a novel framework with which to analyze and understand RE processes. In this framework, RE is viewed as a socio-technical distributed cognitive process, wherein the cognition is distributed across multiple dimensions of social, structural and temporal elements, which interact to form computational structures, that process information of representational states through various workflows, until a final cognitive stage is reached representing requirements closure.

This theoretical model gave rise to our research questions on the applicability of such a model to distinct development environments types, characterized by different project management methodologies, as well as on the potential differences in the computational structures that may evolve to address the characteristics of the varying development environments types. To confirm the validity of distributed cognition framework across multiple types of development, as well as to analyze the resultant computational structure attributes and workflows, we conducted a multi-site field study of three development environments types—structured development, agile development and open-source development.

From a theoretical perspective, our findings validate that distributed cognition is a valid theoretical framework for understanding RE, even across disparate development environments types. Although the degree and extent to which the distribution varied across the social, structural and temporal modes in each of the development environment types, as well as the characteristics and workflows of the computational structures that evolved in each of the
development environment types varied in terms of linearity and overlap across requirements phases, it is quite clear that all three development environment types have a consistent underlying structure which can be theoretically explained by distributed cognition.

Since RE is key source of project management issues and challenges, and the distributed cognition framework can be used to explain and understand RE, it implies that from a practice perspective, distributed cognition is a powerful mechanism to apply to the understanding of project management challenges, especially those that are increasingly being derived from the distributed nature of projects themselves. Specifically, since our findings indicate that the modes and structures of distributed cognition map with varying degrees to attributes of project management methodologies, it implies that project management practice can analyze and apply this framework, to understand the specific conditions and limitations under which the framework can potentially explain the successes and failures of project management.

Both, the theoretical and practical implications above, indicate multiple areas of future research. First, applying the distributed cognition framework to the existing body of research on project management is a potentially rich area of future novel research streams. Second, research can be done to extend the work done in this paper on the differences in the modes and structures of cognition, to theoretically explain the reasons for the evolution of such modes and structures of distributed cognition. Finally, literature review indicates that prior research in application of distributed cognition framework to RE has been somewhat hampered by the limitations of the ethnographic approach taken in such research. Hence a future area of research, would be to replicate and extend the field study approach of this paper, to further validate such application of distributed cognition framework to RE.

REFERENCES


Hansen et al. Distributed Cognitive Analysis of Requirements Evolution in Diverse Environments


Toward a Typological Theory of Information System Project Team Management Styles

Simon Bourdeau
ESG-UQAM
bourdeau.s@uqam.ca

Henri Barki
HEC Montréal
henri.barki@hec.ca

ABSTRACT

Drawing from research in information systems (IS) and project management, the contingency perspective research stream and from empirical observations, the present paper investigates the effect of the fit between information system project team management (ISPTM) styles and four IS project risk profiles, i.e. challenging, diplomatic, technical and easy (cookie cutter), on IS project performance. Following recommended survey development approaches, distribution practices and general conduct of survey research, questionnaire data was obtained from 182 IS project managers in Canada and the United States. The results show that IS project managers adapt their ISPTM to the context specificities and that, for each IS project risk profile, there are significant differences between ISPTM styles deployed in successful and less successful IS projects.

Keywords

IS project management, team, management styles, typology, and typological theory.

INTRODUCTION

Today’s organizations innovate and improve their business processes and activities by relying on novel information systems (IS) (Tiwana and Keil 2006). Such IS are typically developed and implemented by means of transitory ventures known as IS projects (ISP) (Kirsch 2000). ISP are temporary organizations that exist within organizations for as long as it takes to complete the specific tasks they are assigned or to attain particular objectives (Lundin and Söderholm 1995). Led by project managers, ISP are typically carried out by project teams which play a key role in their performance (Faraj and Sambamurthy 2006). Usually, ISP show structural variations when compared to their parent organizations (Shenhar 2001) and one of the key roles of project managers is to establish “… a team structure where no discernible structure exists” (Frame 1995, p.105).

Over the last decades many project management studies have focused on (1) developing and improving project management tools and techniques (e.g. Besner and Hobbs 2008) and (2) examining how specific project management processes affect project outcomes (e.g. Byosiere and Luethge 2007). However, little research has focused on the approaches/strategies deployed by project managers to manage ISP team (ISPT) and, to the best of our knowledge, no conceptual model currently exists that enables project managers to “understand why different approaches exist, which one to choose, and when” (Pich et al 2002, p.1008). There is also a need to develop success-focused typologies of project management styles and how they can be applied in ISP (Thomas and Fernández 2008).

Drawing from research in IS, project management and the contingency perspective, this paper seek to develop a typological theory of ISPTM based on the premise that ISP performance is influenced by the fit between the ISP’s risk and the ISPTM style deployed. The following research questions are addressed in this exploratory study using a survey approach: Do different ISPTM styles exist? And if so, are some styles better suited for certain types of ISP risks? The results suggest that different ISPTM styles exist and were conceptualized here as configurations of different key ISPT processes and governance attributes. Further, the results also suggest that, depending on the ISP’s risk profile, a project manager is likely to adapt his management style, and that different ISPTM styles were deployed in successful ISPs compared less successful projects.

THEORETICAL BACKGROUND – The Construct of IS Project Team Management and Typological Theories

While projects occupy an important place in both the management and IS fields, and despite extensive research efforts deployed over the years, delivering expected benefits of ISP continues to be a difficult endeavor (Nelson 2007). ISP involve the execution of both technical and managerial activities, and ISP managers face predominantly
managerial challenges, especially related to the ISPT (Faraj and Sproull 2000). These challenges stem from the ISP and ISPT specificities which include their cross-functional nature; their task uncertainty and need for knowledge sharing; the distributed and multidisciplinary nature of team members’ expertise and knowledge; the need for team members to collaborate and coordinate with each other and with other project stakeholders; the complex and rapidly changing nature of the technology; and their focus on designing an IS that is abstract and configurable (Boehm and Turner 2004; Mahring 2002). Given the organizational impacts of ISPs and ISPT, the challenges associated with managing ISPTs, and the key roles they play on ISP performance, a better understanding of ISPT management (ISPTM) is needed (Faraj and Sambamurthy 2006; Hoegl and Gemuenden 2001).

Our conceptualization of ISPTM is depicted in Figure 1 (Bourdeau 2012; Bourdeau and Barki 2010) and is based on two fundamental elements of projects in general, and ISP in particular: project team processes and governance attributes. In essence, we view ISPTM as reflecting the manner in which key ISPT processes are governed, and conceptualize it as a multidimensional profile construct (Law et al. 1998) where three ISPT process categories, i.e. transition/planning, action, and interpersonal, are characterized by four ISPT governance dimensions, i.e. participative, decisional, procedural, and regulative. The three ISPT process categories were drawn from the taxonomy of team processes proposed by Marks et al. (2001) which we adapted to ISP contexts. These processes describe the interdependent acts of team members which convert inputs to outcomes and reflect the fundamental activities that enable ISPT to align, direct and monitor their taskwork\(^3\) (Marks et al. 2001). The four ISPT governance dimensions characterize how different processes can be structured and executed, i.e., the different ways in which ISPT activities can be deployed to align, direct and monitor the taskwork. They were identified based on a literature review (for details, see Bourdeau 2012). Finally, the three ISPT process categories and four ISPT governance dimensions were combined within a multidimensional profile model to complete the proposed conceptualization of ISPTM. Figure 2 depicts the ISPTM construct.

\[^{3}\text{Taskwork represents “… what is that teams are doing” (Marks et al, 2001, p.357), whereas the team processes describe “… how they are doing with each other (p.357)”\).}]

---

**Figure 1. Conceptualization of Information Systems Project Team Management**

<table>
<thead>
<tr>
<th>ISPT Process Categories</th>
<th>ISPT Governance Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transition/Planning Processes</strong></td>
<td>Participative “Who” participate(s) in decision-making regarding each ISPTM process</td>
</tr>
<tr>
<td>Project team activities that focus on the evaluation and/or planning of project team activities and on the effort needed to realize the project team’s objectives.</td>
<td>Decisionsal “Who” make(s) the decisions over the ISPTM processes and “who” is/are accountable over the action(s) that result from those decisions</td>
</tr>
<tr>
<td>Action Processes</td>
<td>Procedural “What” levels of formalization and standardization are used to execute decisions regarding each ISPTM process</td>
</tr>
<tr>
<td>Project team activities that directly lead to the accomplishment of the project team’s objectives and which occur during the execution of the project team’s taskwork.</td>
<td>Regulative “How” the implemented decisions will be regulated in terms of behaviors and/or outcomes</td>
</tr>
<tr>
<td>Interpersonal Processes</td>
<td></td>
</tr>
<tr>
<td>Project team activities that are undertaken to manage interpersonal relationships (e.g. conflict management) among project team members.</td>
<td></td>
</tr>
</tbody>
</table>

Each of the three IS project team process category is characterized in terms of the four IS project team governance dimensions.
Toward a Typological Theory of IS Project Team Management Styles

The notion of “management style” is defined here as “…an underlying mode of thinking and behaving that in turn promotes a specific repertoire of actions that managers draw upon in contexts of varying complexity and uncertainty” (Lewis et al. 2002, p.546). In order to describe ISPTM styles, and its influence on ISP performance, several researchers advocate the development of typologies and typological theories (Short et al. 2008).

A typology identifies multiple archetypes or ideal types which are “…complex constructs that can be used to represent holistic configurations of multiple unidimensional constructs” (Doty and Glick 1994, p.233). Typological theories identify the pathways connecting particular archetypes to specific outcomes, such as project performance, whose variance they are intended to predict (George and Bennett 2005). They differ from traditional theories, since they do not “…highlight the hypothesized relationships between the unidimensional first-order constructs and the dependent variable(s)” (Doty and Glick 1994, p.234). Instead, their focus is on the internal consistency and synergetic effects among the first-order constructs within each archetype (Delery and Doty 1996).

In a typological theory, archetypes must be defined by specifying, empirically and/or theoretically, multivariate profiles (George and Bennett 2005). Usually, greater similarity between an empirical manifestation and an archetype will engender greater effectiveness because factors are thought to be consistent within each archetype (Doty and Glick 1994). In the context of ISP, the correspondence between ISPTM styles (or archetypes) and the ISPTM styles deployed by project managers in actual ISPs can be modeled in terms of profile deviation (Doty et al. 1993; Venkatraman 1989).

The development of an ISPTM typological theory requires the identification of pathways that connect ISPTM styles to specific outcomes. While there is no such thing as a “correct” ISPTM style, some styles are likely to better fit particular project characteristics (Shenhar 2001). In a typological theory, archetypes have to be conditional to certain contingency factors (Doty and Glick 1994) and these factors restrict the number of possible archetypes. Thus, a deviation analysis can be conducted to identify, in the presence of specific contingencies, the most appropriate ISPTM styles (George and Bennett 2005).

RESEARCH MODEL - IS Project Risk and a Contingency Model

The underlying logic of structural contingency theory is that organizational performance is influenced by the fit between task uncertainty and an organization’s ability to structure and execute its activities in ways to cope with this uncertainty (Galbraith 1977; Lawrence and Lorsch 1967). This suggests that contingencies, such as project
uncertainty or risk\(^4\), faced by an ISP are likely to influence the best ways in which to structure and execute different ISPT processes in order to increase project performance (Gresov 1989). In this context, it is important to note that project risk has been recognized as a key contingency factor that needs to be taken into consideration when managing ISP (Schmidt et al. 2001). For example, project risk has been found to influence how ISP are managed (Jiang et al. 2001a) and risk factors represent multiple contingencies which can affect ISPs’ execution and performance (Gemino et al. 2007-8). ISP managers who matched their management approaches/strategies with ISP risks have been found to increase project performance (Barki et al. 2001). Clearly identifying an ISP’s risks prior to its start and during its execution can provide a means to select the most appropriate managerial strategy or style (Jiang et al. 2001b). For example, Ropponen and Lyytinen (2000) found that a general managerial strategy, such as ISP management, was more effective than risk management techniques in mitigating ISP risks. Such results are also consistent with work which views risk management and project management as being analogous (McFarlan 1981; Wallace et al. 2004a).

However, much of past research has examined ISP management and ISPTM strategies in isolation rather than as configurations which some researchers have suggested (e.g. Kirsch 2000). The ISPTM construct (Figure 2) can be viewed as a set of interconnected managerial strategies, i.e. various configurations of ISPTM processes and attributes, which can be structured and executed differently depending on the risks of a particular ISP in order to achieve better performance (Wallace et al. 2004a). Based on these considerations and contingency theory, our premise is that ISP performance will be likely depend on the “fit” between ISPTM styles and ISP risks (Figure 3).

### Four IS Project Risk Profiles

In the IS literature, researchers have identified various ISP risk factors which need to be identified and controlled in order to reduce the likelihood of project failure (Sherer and Alter 2004). Various risk checklists and classification frameworks have been proposed (e.g. Schmidt et al. 2001) but this multiplicity of frameworks and checklists renders their integration and comparison of risks difficult. A useful approach to address these limitations is to categorize risk as either social or technological (e.g. Mathiassen et al. 2007). These two risk dimensions represent complementary aspects of ISP risks and are consistent with the sociotechnical system theory which emphasizes the fit between social and technical subsystems (Trist 1981). As such, the two dimensions can be used to define a matrix of four ISP risk profiles (Figure 4) which can yield more interpretable and theoretically interesting patterns than any risk element taken in isolation (Rousseau and Fried 2001).

---

\(^4\) As noted by Barki et al. (2001), “… strong parallels have been shown to exist in the meanings attributed to these terms (i.e. uncertainty and risk) in the IS literature, with both terms used to describe project characteristics that tend to increase the probability of project failure (p.43).”
Information Systems Project Team Management and Fit

Several researchers have observed that the characteristics of the ISP itself influence the selection and use of risk management strategies (e.g. Martin 2003; McFarlan 1981). Past research has suggested that ISP performance is influenced by the fit between ISP risks and the management strategies deployed by project managers (Barki et al. 2001). Thus, based on contingency theory (Galbraith 1977), it can be hypothesized that different ISPTM styles will be better suited for different ISP risk profiles (Figure 4) and that the degree of fit between the ISPTM style and the ISP risk profile will be likely to influence project performance.

When adopting a contingency approach, the conceptualization of fit should be clearly defined because each conceptualization implies different meanings (Venkatraman 1989). Given our objective to explore the effect of the fit between ISPTM styles and ISP risk profile on performance (i.e. criterion-specific), fit as profile deviation seems to provide an appropriate conceptualization (Venkatraman 1989). The profile deviation approach allows researchers to empirically assess complex and multidimensional relationships that are more consistent with a holistic perspective than traditional approaches (Meyer et al. 1993). In the present study, ISPTM fit is defined as the degree to which an ISPTM style is appropriate for a specific ISP risk profile. The next section describes the research methodology that was used to examine the effect of the fit between ISPTM styles and ISP risk profiles on project performance.

METHODOLOGY - Survey

As described below, following recommended survey development approaches, distribution practices and general conduct of survey research (e.g. Dillman et al. 2008; Furneaux and Wade 2011; Ju et al. 2006) a paper questionnaire was developed and distributed to a cross-section of ISP managers in Canada and the United States.

Construct Definitions. A first step was to clearly define the study’s key constructs, i.e. (1) ISPT processes and (2) ISPT governance attributes. Data were collected for the ISPT process categories, i.e. transition/planning, action and interpersonal (Figure 2), rather than for each of the eleven ISPTM processes since the study was exploratory and that assessing all eleven processes would have required more than 300 items, tripling the questionnaire’s length and completion time. As recommended by Marks et al. (2001), the higher order process categories were used as representative of their underlying lower-order processes. Each was defined based on the definitions of the 11 ISPT processes (see Table 2). The ISPT governance attribute definitions are provided in Table 3 (Bourdeau 2012).

<table>
<thead>
<tr>
<th>ISPTM Process Categories</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transition/Planning Processes</td>
<td>The transition/planning processes refer to project team management activities which focus on the evaluation and/or planning of a project and on the effort needed to realize the project’s team objectives: (1) mission analysis, (2) goal setting, (3) strategy formulation, (4) work planning, (5) team creation and (6) project evaluation and review.</td>
</tr>
<tr>
<td>Action Processes</td>
<td>The action processes refer to project team management activities that directly lead to the accomplishment of the project’s team objectives and which occur during the execution of the project team’s tasks: (1) monitoring the project’s progress, (2) controlling the project’s resources, (3) managing project risk, (4) coordinating people and activities and (5) sharing information.</td>
</tr>
<tr>
<td>Interpersonal Processes</td>
<td>Interpersonal processes refer to project team management activities that are undertaken in order to manage interpersonal relationships among the stakeholders of an IT project: (1) conflict management, (2) team-building, (3) trust development, (4) motivation building, (5) cohesion establishment, and (6) social integration.</td>
</tr>
</tbody>
</table>

Table 2. Definitions of ISPT Process Categories
Bourdeau and Barki  
Toward a Typological Theory of IS Project Team Management Styles

<table>
<thead>
<tr>
<th>ISPT Governance Attributes</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Centralization of decision rights</td>
<td>The pattern of decision rights allocation between the project manager and the project team, regarding decisions about an ISPT process.</td>
</tr>
<tr>
<td>2. Cooperativeness of decision-making</td>
<td>The pattern of accountability, between the project manager and the project team, about the action(s) related to ISPT processes and resulting from decisions.</td>
</tr>
<tr>
<td>3. Level of Formalization</td>
<td>The extent, to which an ISPT process is written, documented, explicitly formulated and/or scheduled.</td>
</tr>
<tr>
<td>4. Level of Standardization</td>
<td>The extent to which the norms and evaluation criteria of the execution of a given ISPT process are codified in a pre-specified standard.</td>
</tr>
<tr>
<td>5. Level of Behavioral Directives</td>
<td>The extent to which detailed information about the execution of a given ISPT process is provided.</td>
</tr>
<tr>
<td>6. Level of Outcome Directives</td>
<td>The extent to which detailed information about the outcomes of a given ISPT process is provided.</td>
</tr>
<tr>
<td>7. Level of Client Participation</td>
<td>The extent to which the ISP client participate in ISPT process decisions.</td>
</tr>
<tr>
<td>8. Level of External Stakeholder Participation</td>
<td>The extent to which key external stakeholders of the ISP participate in ISPT process decisions.</td>
</tr>
</tbody>
</table>

Table 3. Definitions of ISPT Governance Attributes

Identifying Respondents. ISP managers were deemed to be the most appropriate respondents as they are typically the individuals who are most knowledgeable about the processes, decisions and risks associated with the projects they manage. In order to minimize memory bias (Lechler and Dvir 2010), the respondents were asked to select two recently completed projects (one successful and one less successful) that were relatively fresh in their mind. The first part of the questionnaire contained questions about a project the respondent had recently managed and that they thought had been successful in terms of project budgets and schedules, whereas the second part asked the same questions, but for a different project that the respondent viewed as having been less successful than the first project. The questionnaire items pertained to the ISPTM styles deployed (unit of analysis) by the project manager, the ISPs’ risk profiles and the performances.

Questionnaire Development

ISP risk was measured with 17 items\(^5\) from Wallace et al. (2004a; 2004b) and performance was measured with four items from Gemino et al. (2007-8). ISP satisfaction was assessed with five items developed from Turner and Müller’s (2006) items, whereas ISPTM style items were created in accordance with their construct definitions, as described below.

ISPTM Styles. The definitions of the ISPT process categories and governance attributes served as a starting point for developing items. First, the conceptual definitions were examined to identify their key elements. For the ISPT processes, the original process category (Marks et al. 2001) and the operationalizations suggested by other researchers (e.g. Morgeson et al. 2010) were also reviewed. For the ISPT governance attributes, the definitions of Bourdeau (2012) and the operationalizations proposed by other researchers (e.g. Faraj and Sambamurthy 2006) were reviewed. Next, based on the definitions, items were created to assess the extent to which the structure and execution of each ISPT process category was characterized by the eight ISPT governance attributes. Likert scales (1= “Not at all” to 7= “To a great extent”), were used to measure most governance attributes (Table 4).

---

\(^5\) Due to space limitations, the detailed questionnaire is not included here, but can be provided on demand.
1. **Centralization of decision rights:** In your SUCCESSFUL IT project, how were the organizational decision rights shared between you, the project team members and the other project stakeholders with respect to transition/planning activities? Please provide a percentage representing the degree of responsibility of each group regarding the transition/planning activities. (The total of the three percentages should be 100%).
   a. With regards to decisional rights, YOU were responsible of what percentage of the transition/planning activities:
   b. With regards to decisional rights, the project team members were responsible of what percentage of the transition/planning activities:
   c. With regards to decisional rights, the other project stakeholders were responsible of what percentage of the transition/planning activities:

2. **Cooperativeness of decision making:** In your SUCCESSFUL IT project, to what extent were the project team members involved in the decision-making process regarding the transition/planning activities:

3. **Level of formalization:** In your SUCCESSFUL IT project, to what extent were the transition/planning activities written, explicitly formulated and documented:

4. **Level of standardization:** In your SUCCESSFUL IT project, to what extent were pre-existing guidelines (e.g. standards, reference documents, methodologies, evaluation criteria, etc.) used to manage the transition/planning activities?

5. **Level of behavioral directives:** In your SUCCESSFUL IT project, to what extent did you provide detailed instructions to project team members on how to perform the transition/planning activities:

6. **Level of outcomes directives:** In your SUCCESSFUL IT project, to what extent did you ask that project team members attain specific objectives regarding transition/planning activities:

7. **Level of client participation:** In your SUCCESSFUL IT project, to what extent were representatives of the project’s clients involved in the management of the transition/planning activities:

8. **Level of external stakeholder participation:** In your SUCCESSFUL IT project, to what extent were representatives of the project’s external suppliers involved in the management of the transition/planning activities:

*The example provided here is related to the transition/planning activities realized in the successful IT project. The same logic was applied for the action and interpersonal activities in both the successful and less successful ISP.

**Table 4. Examples of ISPTM Style Items**

**ISP Risk Profiles.** Each ISP risk profile was characterized based on the profile model (Law et al. 1998). This approach is consistent with Iversen et al.’s (2004) risk-strategy models which relate risk configurations to specific patterns of resolution techniques. Thus, ISP risk profiles were assessed along two distinct risk dimensions: technological risk and social risk, via the 17 risk items of Wallace et al. (2004a; 2004b).

**ISP Performance.** According to Nelson (2005), ISP performance should be evaluated on process and outcome criteria, as well as on satisfaction since the main objective of the project manager should be to maximize stakeholder satisfaction. We assessed ISP performance via four outcome and process measures adopted from Gemino et al. (2007-8) and the satisfaction measure of Turner and Müller (2006).

**Pretest, sampling and distribution.** A draft questionnaire was created, pretested with four experienced ISP managers and revised based on their comments and suggestions. The initial sampling frame was based on personal and professional contacts and was completed via a snowball sampling approach (Salganik and Heck 2004). The questionnaires were distributed following the principles of the tailored design method (Dillman et al. 2008) as it seeks to increase respondents’ trust, reduce the perceived cost of responding and increase its completion rate.

**Results and Data Analysis**
In total, 182 usable questionnaires were received (out of the 255 that were sent)\(^6\), for a response rate of 71.3%. Two graduate students separately coded all questionnaires into Excel. Then, the two data files were overlapped into a third spreadsheet to identify any coding errors. For each error identified, the original questionnaire was checked and the correct answer entered. Table 5 provides descriptive characteristics of the sample. As can be seen, the sample exhibited considerable variation in terms of project manager gender, age, and experience, and the duration and size of the projects, providing support for the sample’s representativeness.

<table>
<thead>
<tr>
<th>Respondent Characteristics (N=182)</th>
<th>IS Project Characteristics (N=363)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Descriptions</strong></td>
<td><strong>Descriptions</strong></td>
</tr>
<tr>
<td>Gender</td>
<td>Project duration</td>
</tr>
<tr>
<td>Mean</td>
<td>(months)</td>
</tr>
<tr>
<td>F = 56 ; M = 126</td>
<td>14.9</td>
</tr>
<tr>
<td>S.D.</td>
<td>9.96</td>
</tr>
<tr>
<td>Min.</td>
<td>2</td>
</tr>
<tr>
<td>Max.</td>
<td>60</td>
</tr>
<tr>
<td>Age (in years)</td>
<td>Project size</td>
</tr>
<tr>
<td>Mean</td>
<td>(person-days)</td>
</tr>
<tr>
<td>42.7</td>
<td>6369</td>
</tr>
<tr>
<td>S.D.</td>
<td>17005</td>
</tr>
<tr>
<td>Min.</td>
<td>75</td>
</tr>
<tr>
<td>Max.</td>
<td>132000</td>
</tr>
<tr>
<td>Nb. of years of experience in ISP management</td>
<td>Nb. of individuals who worked in the project</td>
</tr>
<tr>
<td>Mean</td>
<td>31</td>
</tr>
<tr>
<td>S.D.</td>
<td>64.92</td>
</tr>
<tr>
<td>Min.</td>
<td>3</td>
</tr>
<tr>
<td>Max.</td>
<td>1000</td>
</tr>
<tr>
<td>Nb. of managed ISP</td>
<td>Proportion of projects in the private sector</td>
</tr>
<tr>
<td>Mean</td>
<td>68.7%</td>
</tr>
<tr>
<td>S.D.</td>
<td>---</td>
</tr>
<tr>
<td>Min.</td>
<td>---</td>
</tr>
<tr>
<td>Max.</td>
<td>---</td>
</tr>
<tr>
<td>% with PM certification</td>
<td>51%</td>
</tr>
</tbody>
</table>

Table 5. Characteristics of the Respondents and IS Projects

**Construct Reliability and Validity**

The 182 usable questionnaires provided data on 364 ISPs. From these, 18 projects were discarded due to missing data, yielding a final sample of 346 ISPs. The unidimensionality of all constructs (measured with reflective items) was examined via exploratory factor analysis and Varimax rotation using the PASW Statistics 18 (SPSS) package. As all items clearly loaded on their respective dimensions\(^7\), scales were created for each construct by averaging its item scores. The Cronbach alphas (Alpha) of the risk and performance constructs (see Table 6) exceeded the recommended minimum value of 0.7 (Hair et al. 2005; Nunnally 1978).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Process Performance (Pro. Per. -- 2 items)</td>
<td>3.11</td>
<td>1.11</td>
<td>.722*</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2. Outcome Performance (Out. Per. -- 2)</td>
<td>4.25</td>
<td>1.20</td>
<td>.737</td>
<td>.223*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Satisfaction Performance (Sat Per. -- 5)</td>
<td>4.68</td>
<td>1.32</td>
<td>.916</td>
<td>.418*</td>
<td>.636*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Global Performance (Glo. Per. -- 9 items)</td>
<td>12.03</td>
<td>2.87</td>
<td>.893</td>
<td>.672*</td>
<td>.796*</td>
<td>.887*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Organizational Env. Risk (Env. Risk -- 4)</td>
<td>4.16</td>
<td>1.60</td>
<td>.786</td>
<td>-.151*</td>
<td>-.053</td>
<td>-.248*</td>
<td>-.195*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. User Risk (User Risk -- 3)</td>
<td>3.37</td>
<td>1.62</td>
<td>.883</td>
<td>-.276</td>
<td>-.138</td>
<td>-.354*</td>
<td>-.327</td>
<td>.435*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Requirement Risk (Req. risk -- 3)</td>
<td>4.27</td>
<td>1.80</td>
<td>.894</td>
<td>-.417*</td>
<td>-.244*</td>
<td>-.445*</td>
<td>-.468</td>
<td>.342*</td>
<td>.443*</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Project Complexity Risk (Com. Risk -- 4)</td>
<td>4.02</td>
<td>1.66</td>
<td>.838</td>
<td>-.183*</td>
<td>-.123</td>
<td>-.137</td>
<td>-.185*</td>
<td>.152*</td>
<td>.210*</td>
<td>.160*</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. Social Risk (Soc. Risk -- 7)(^9)</td>
<td>3.82</td>
<td>1.38</td>
<td>.819</td>
<td>-.237*</td>
<td>-.111</td>
<td>-.347*</td>
<td>-.298</td>
<td>.887*</td>
<td>.799*</td>
<td>.454*</td>
<td>.219*</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>10. Technological Risk (Tec. Risk -- 7)(^10)</td>
<td>4.13</td>
<td>1.33</td>
<td>.783</td>
<td>-.369*</td>
<td>-.233*</td>
<td>-.355*</td>
<td>-.403*</td>
<td>.307*</td>
<td>.402*</td>
<td>.696*</td>
<td>.820*</td>
<td>.416*</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6. Cronbach Alphas and Construct Correlations

---

\(^6\)Nonresponse bias was tested via two post-hoc techniques (Furneaux and Wade 2011): (1) early and late respondents were compared on substantive variables and (2) follow-ups were conducted with non-respondents to assess the reasons for nonresponse. The results suggest that nonresponse bias was unlikely and that respondents could be pooled with no loss in generalizability.

\(^7\)Due to space limitation, the item cross loadings are not shown here, but can be provided on demand.

\(^8\)All scales are on 7 points, except “Global Performance”, which is on 21 points.

\(^9\)Following Hulin et al. (2001), the standardized Cronbach’s alpha for a two item scale was estimated as: \(\alpha = Kr / (1+(K-1)r)\), where “K” is the number of components, i.e. 2, and “r” is correlation coefficient between the two items.

\(^10\)Global performance is the average score of process performance, outcome performance and satisfaction.

\(^11\)Based on Wallace et al. (2004a), the social risk measure is a composite of the user and organization environment items.

\(^12\)Based on Wallace et al. (2004a), the technological risk measure is a composite of the requirement and project complexity items.

---
Step #1: Key ISPT Governance Attributes Identification

Since the ISPT governance attributes were empirically tested here for the first time, a first step of our exploratory study was to identify the key ISPT governance attributes that captured the essence of ISPTM styles. To do so, a profile deviation approach was applied. Following the procedure used by several researchers (e.g. Barki et al. 2001; Venkatraman and Prescott 1990), ideal profiles were identified from the sample of 346 ISPs. Four ideal profiles were established, one for each of the four performance variables: process performance, outcome performance, satisfaction and global performance.

In profile deviation studies, ideal profiles are calibrated using the highest performing 10% to 15% of the respondents in a data set (Vorhies and Morgan 2003). Thus, for example, the average ISPT governance attribute scores of the projects that best performed in terms of “process performance” formed the “process performance ISPTM style”, i.e. the ideal profile that represented the “best” way to manage ISPT when the objective is to maximize process performance. For the 85% to 90% of the remaining respondents of the sample, a Euclidean distance was calculated (Barki et al. 2001) between the score of the ideal ISPTM style and the score of each project. This calculation yields a profile deviation score that represents the degree to which the ISPTM style deployed in a particular project is similar to the “ideal” ISPTM style.

Next, the impact on performance of the “extent of closeness to the ideal ISPTM style” was assessed. A distance score was individually calculated for each of the three ISPT process categories on each of the ISPT governance attributes. The profile deviation score of each ISP was then correlated with the four performance variables (Barki et al. 2001). Deviations from the ISPTM styles that significantly and negatively correlate with performance variables would provide supporting evidence for the validity and utility of the proposed ISPTM styles. The calculation of the individual distances for each ISPT governance attribute on each of the three ISPT process categories was repeated for each of the four performance variables and yielded a 3x4 correlation matrix.

The results suggested that a combination of the three following ISPT governance attributes yielded the highest number of negative and significant correlations: (1) relative decisional rights distribution between stakeholders (SK) and the project manager (PM)\(^1\) (i.e. centralization); (2) the level of control\(^2\), (3) the team member’s (TM) level of cooperativeness in decision making (i.e. cooperativeness). Due to the exploratory nature of the study and since these three ISPT governance attributes seem to capture the essence of ISPTM, only these three attributes were retained for further analysis.

Step #2: Fit between ISP Risk Profiles and ISPTM Styles

To examine the effects of the fit between ISPTM styles and ISP risk profiles on performance, we first established the risk profile of each ISP by calculating their social and technological risk levels. To do so, the complete sample was used to calculate the medians of social risk (SR) and technological risk (TR), which were 3.857 and 4.0, respectively. Then, each ISP was classified, based on its SR and TR scores, into one of the four risk profile category as follows:

1. Challenging \(\rightarrow\) SR >= 3.857 and TR >= 4.0.
2. Easy (cookie cutter) \(\rightarrow\) SR < 3.857 and TR < 4.0
3. Diplomatic \(\rightarrow\) SR >= 3.857 and TR < 4.0
4. Technical \(\rightarrow\) SR < 3.857 and TR >= 4.0

\(^1\) The relative decisional rights distribution between SK and PM is calculated by dividing the percentage of SK’s decision rights by the percentage of the PM’s decision rights.

\(^2\) The eight governance attributes, empirically tested for the first time, were examined via an exploratory factor analysis. The results show a clear and interpretable pattern formed by four attributes: formalization, standardization, behavioral directives and outcome directives. This result is plausible as these attributes represent mechanisms that project managers use to ensure that team members act in a manner that is consistent with the project’s objectives. A new scale labeled “Level of control” was created by averaging their scores and its Cronbach alpha exceeded the recommended minimum value of 0.7.
Next, the IPSs in each category were further classified as either successful or less successful based on the respondents’ categorization (see Table 7).

<table>
<thead>
<tr>
<th>Technical Projects (N=63)</th>
<th>Challenging Projects (N=118)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Successful</strong></td>
<td><strong>Less Successful</strong></td>
</tr>
<tr>
<td>N = 33</td>
<td>N = 30</td>
</tr>
<tr>
<td>Easy (cookie cutter) Projects (N=104)</td>
<td>Diplomatic Projects (N=61)</td>
</tr>
<tr>
<td><strong>Successful</strong></td>
<td><strong>Less Successful</strong></td>
</tr>
<tr>
<td>N = 77</td>
<td>N = 27</td>
</tr>
</tbody>
</table>

Table 7. Number of Projects in each Risk Profile Category

Then, in order to identify the ideal ISPTM styles of each risk profile category, the ISPTM styles of successful and less successful projects were obtained by following the same analysis steps as described above. To verify that the ISPTM styles in successful and less successful projects were different, two-tail t-tests were done for the governance attributes in each process category (Hair et al. 2005), and the results are shown in Tables 8 to 11 with the key governance attributes of each ISPTM styles printed bolded and italicized. Each project manager’s ISPTM style can vary according to the governance attributes that characterize each ISPT style. While the levels of some governance attributes can vary from one ISPTM style to another, they can be very similar for others. However, it is the combination of all the ISPT processes and governance attributes associated with a particular ISPTM style that differs between different styles (George and Bennett 2005).

Tables 8 to 11 suggest that, in each ISP risk profile category, project managers in successful ISPs deployed ISPTM styles which were different than those deployed by project managers in less successful ISPs since, for each ISPTM style, the levels and the combinations of the ISPT processes and governance attributes differed. For example, in “challenging” ISPs, the project managers of successful ISPs, had: 1) a higher level of control over planning and transition processes (P&T); 2) a lower level of relative decisional rights distribution over the P&T and the interpersonal processes; 3) a higher level of cooperativeness of decision-making regarding the P&T and the action processes, than the managers of less successful ISPs (Table 9).

In addition, the results of Tables 8-11 facilitate the comparison of not only the different ISPTM styles, but also of the ISPT processes. For example, while in the case of “challenging” ISPs five governance attributes seemed to make a difference between successful and less successful ISPs, only two attributes seemed to make a difference for “diplomatic” ISPs. Further, eight of the 15 significant results found in Tables 8-11 can be seen to pertain to planning and transition processes, which highlights their importance in managing ISPs.

To examine the influence of the fit between ISPTM styles (Tables 8-11) and ISP risk profiles on project performance, correlations between performance measures and profile distance scores were calculated. Here, the same analysis steps described above, were followed. First, for each type of risk profile category, an ISPTM style was established based on the top 10 to 15 % projects in terms of global performance. Next, individual profile distance scores were calculated for the remaining 85% to 90% ISPs. Finally, correlations between the profile deviation scores and the different performance variables were calculated (Table 12).

<table>
<thead>
<tr>
<th>Performance Measures</th>
<th>Easy (cookie cutter)</th>
<th>Technical</th>
<th>Diplomatic</th>
<th>Challenging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>-.047**</td>
<td>-.332**</td>
<td>.007</td>
<td>.142</td>
</tr>
<tr>
<td>Outcome</td>
<td>-.241**</td>
<td>-.228</td>
<td>-.441**</td>
<td>-.230*</td>
</tr>
<tr>
<td>Satisfaction</td>
<td>-.298**</td>
<td>-.175</td>
<td>-.168</td>
<td>-.211**</td>
</tr>
<tr>
<td>Global</td>
<td>-.261*</td>
<td>-.302</td>
<td>-.308*</td>
<td>-.137</td>
</tr>
</tbody>
</table>

Table 12. Effects of fit between ISP risk profiles and ISPTM Styles on Performance

\( ** = p < .01; * = p < .05 \)

The results of Table 12 indicate that, generally, deviations from ideal ISPTM styles were significantly and negatively correlated with performance measures, providing support for the proposed contingency model of Figure 3.
### Toward a Typological Theory of IS Project Team Management Styles

<table>
<thead>
<tr>
<th>Governance attributes</th>
<th>SUCCESFUL (N =33)</th>
<th>t-test</th>
<th>LESS SUCCESFUL (N=30)</th>
<th>PLANNING AND TRANITION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>S.D</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1. Level of control</td>
<td>5.49</td>
<td>1.00</td>
<td>**</td>
<td>4.13</td>
</tr>
<tr>
<td>2. Rel. D.Rights</td>
<td>0.67</td>
<td>0.75</td>
<td>-2.16</td>
<td>*</td>
</tr>
<tr>
<td>3. Cooperativeness</td>
<td>5.16</td>
<td>1.51</td>
<td>1.46</td>
<td>ns</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Governance attributes</th>
<th>SUCCESFUL (N =34)</th>
<th>t-test</th>
<th>LESS SUCCESFUL (N=84)</th>
<th>PLANNING AND TRANITION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>S.D</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1. Level of control</td>
<td>4.79</td>
<td>1.17</td>
<td>*</td>
<td>2.09</td>
</tr>
<tr>
<td>2. Rel. D.Rights</td>
<td>0.74</td>
<td>0.99</td>
<td>-2.38</td>
<td>*</td>
</tr>
<tr>
<td>3. Cooperativeness</td>
<td>5.15</td>
<td>1.62</td>
<td>2.63</td>
<td>**</td>
</tr>
</tbody>
</table>

**Note:** Two-tail T-test: ** = p < .01; * = p < .05; ns = none significant.

**Rel. D.Rights** = Relative Decisional Rights distribution between Project Manager and Other Project Stakeholders. **Cooperativeness** = Cooperativeness of decision-making with Team members.
DISCUSSION AND CONCLUSION

The present study had two main objectives: 1) to identify ISPTM styles, and 2) to determine if some styles were better suited to certain types of ISP risk profiles. ISPTM styles were conceptualized here as configurations of different key ISPT processes and governance attributes. The results of Tables 8 to 11 suggest that, as hypothesized, different ISPTM styles do seem to exist among practicing ISP managers. These results are interesting because they show that, depending on the type of ISP risk profile, a project manager is likely to adapt his/her management style. They also show that there are important differences between the ISPTM styles deployed in successful ISPs compared to less successful ISPs. This provides support to the idea that the “one size fits all” (Shenhar 2001) approach is probably not the most appropriate way to manage ISPT. It also suggests that a more holistic and configuration perspective of ISPTM can be useful for providing new insights and enrich our understanding of ISP management.

Further, the mostly significant results of Table 12 regarding the impact on ISP performance of the fit between different ISPTM styles and ISP risk profiles also provide an important step in the development of a typological theory of ISPTM styles (George and Bennett 2005). As suggested in Table 12, the closer an ISPTM profile deployed by a project manager was to an “ideal” ISPTM style, the higher was the ISP’s performance on specific measures. For instance, in the context of easy (cookie cutter) projects, the closer the ISPTM profile deployed by an ISP manager was to the “ideal” ISPTM style of cookie cutter ISPs, the higher was the project’s performance in terms of outcomes, stakeholder satisfaction and global performance. Thus, depending on the characteristics of an ISP and its context (which can be described via risk factors), certain management strategies or styles seem to be better suited than others. Thus, it would be interesting in future research to examine other contingencies which might affect ISPTM styles, such as task interdependence (Sharma and Yetton 2003) and goal conflict (Andres and Zmud 2001), as well as examining how ISPTM styles change throughout the various phases of an ISP (Kirsch 2004).

The present study also identified three key ISPT governance attributes that seemed to capture the essence of ISPTM: 1) relative decisional rights distribution between stakeholders and the project manager; 2) level of control and 3) team members’ level of cooperativeness in decision making. Identifying the relative distribution of decision rights as a key attribute constitutes an interesting finding as it is directly aligned with Rowley’s view (1997) that, when trying to understand a group’s functioning, it is important to look at the balance of power or decision rights between stakeholders, and not only at who has decision rights. The level of control also seems to make sense as a key governance attribute since in any ISP, even in agile projects (Schwaber and Jeff Sutherland 2011), minimal levels of formalization, and a certain standardization of behavioral and outcome directives are required (Kirsch 2004; PMI 2013). The cooperativeness of team members also makes sense since, to make a decision, a project manager needs to have valid and precise information on the project’s progress, with the team members being the best positioned to provide such information (Henry et al. 2003).

Given the relatively exploratory nature of the present study, its results need to be interpreted in light of its limitations. A first limitation stems from the cross-sectional nature of the study which limits our ability to explore how ISPTM styles unfold over time or throughout life cycle stages. Second, the ISP performance measures were based on project managers’ own assessments and were therefore subjective. Although self-evaluation of performance has been widely adopted in ISP research, self-perceptions of success may provide biased evaluations. Finally, the potential effects of industry characteristics (e.g. technology versus manufacture) and the complexity of projects could not be considered in the interpretation of results.

REFERENCES


Lawrence, P. R., and Lorsch, J. W. 1967. *Organization and Environment: Managing Differentiation and Integration* Division of Research, Graduate School of Business Administration, Harvard University, Boston, MA, p. 279.


Understanding the Information Technology Growth Options: Effects of Gender and Experience on Option Exercise Decisions

Sarah S. Khan  
National University of Sciences & Technology, Pakistan  
sarah.khan@seecs.edu.pk

Ram L. Kumar  
University of North Carolina  
rlkumar@uncc.edu

Kexin Zhao  
University of North Carolina  
kzhao2@uncc.edu

Antonis Stylianou  
University of North Carolina  
astylian@uncc.edu

ABSTRACT

To account for managerial decision flexibility in risky IT investments, Real Option Valuation (ROV) has been advocated. ROV formalizes managers’ intuition, thus creating a disciplined decision making process. However, evidence suggests that ROV is usually utilized intuitively by professionals, in the form of “Real Option Thinking”, and is subject to various judgmental biases. We focus on growth options for this study. Prior research has shown that, while valuing projects with real options, managers ascribe the greatest importance to projects with growth options. Similar results hold for IT projects, where IT managers perceive a growth option as adding more value to the project. This perception of growth options might suggest their vulnerability to the IT managers’ risk preferences, through Prospect Theory. By conducting a survey-based experiment among 150 IT professionals, our results indicate that gender and experience impact biases in growth option exercise decisions significantly, depending on project size. However, we also observe some exceptions.

Keywords

Project management, real options, decision making/makers, gender differences, experience

INTRODUCTION

An IT project possesses a real option when it offers managerial flexibility to change the course of the project in response to endogenous or exogenous events (Benaroch and Kauffman, 1999). Such managerial flexibility in decision making allows more efficient capital investments while curtailing the unsystematic risk (Benaroch and Kauffman, 1999; Tiwana et al. 2006). In order to better account for decision-making flexibility and managerial risks, several studies (Clemons, 1991; Dos Santos, 1991; Kambil et al., 1993; Kumar, 1996; 2002) have used illustrative examples to propose using real options theory in IT investments. The IS literature primarily focuses on real options analysis for large and risky IT investments with known risks, under the assumption that some embedded real options already provide management with strategic and operational flexibility needed to manage risks (Benaroch et al., 2007).

Several types of real options may exist in an IT investment. These include the strategic growth option, the option to defer investment, the option to scale (up or down), and the option to terminate the investment. Benaroch et al., (2007) describe the types of real options identified in the IT context along with related risks. IT real options are further classified in two groups (Trigeorgis, 1993, Benaroch, 2002): growth and operational options15. From an options perspective, an IT investment project is seen as creating a base asset with some expected value, such as the baseline implementation of an enterprise resource planning (ERP) package (Tiwana et al., 2006). Growth options capture the possibility of building additional assets on top of the base asset if the initial project is to be completed.

15 Some studies like Tiwana et al., (2006) view the option to defer as the growth option and not the operational option.
For example, a firm may have an opportunity to build a data warehouse to facilitate the analysis of data captured in an ERP system implemented earlier. Operational options relate to flexible actions that managers can make to reduce the potential for losses (usually) or increase the potential for gains (occasionally) on that base project. Operational options give managers the flexibility to change the features of a base project by modifying its timing, scale, or scope, while strategic growth options provide an opportunity to create one or more additional but related assets beyond the asset produced by the base project (Benaroch, 2002). It is important to note that in the case of operational options, there is only one asset under evaluation (i.e., the base system), while in the case of strategic growth options, there are multiple assets to consider (the base system, plus one or more future assets that build on the base system).

This classification of real options has been used in identification and recognition of real options (Benaroch, 2002), and valuation of IT investments with embedded real options in different scenarios (e.g. Dos Santos, 1991; Kumar, 2002; Benaroch and Kauffman, 1999; Su et al., 2009 etc.). Growth options are considered as having call option characteristics (right to buy an asset in the future), while operational options are considered as having put option characteristics (right to sell an asset in the future) (Benaroch, 2002).

Although the real options approach has been shown to be beneficial in managing the uncertainty in IT projects, this approach has been found to be utilized informally by managers in practice. Such informal use has been referred to as “Real Options Thinking” (Amram and Kulatilaka, 1999; Fichman et al., 2005). Unlike real option valuation, which is based on formal mathematical models, real options thinking refers to intuitive decision making after identifying real options embedded in an investment. Recent studies have shown that decision makers exhibit several systematic biases while utilizing this methodology (Goswami et al., 2008; Lankton and Luft, 2008; Tiwana et al., 2006; 2007). Previous research also illustrates that subjective frames impact the expected value of real options in general, as well as in the case of IT projects (Millar and Shapira, 2004; Tiwana et al., 2006).

Real options thinking and valuation are beneficial when their value is realized during the project management. However, real option value depends on the optimal timing of exercise (Dos Santos, 1991; Kumar, 1999). Real options are most beneficial when they are exercised optimally. Given the evidence that managers may be biased and hence not exercise options optimally, we examine the implications of such bias. We study option exercise decisions made by IT managers, who we view as bounded rational agents with preferences governed by Prospect Theory (Kahneman and Tversk, 1979). Prospect theory suggests that biased preferences exist independently of individual differences (Kahneman and Tversk, 1979). However, other studies suggest that individual characteristics, such as gender and experience, may guide risk taking behavior in individuals. We chose gender and experience for the following reasons. First, real option exercise decisions are considered risky decisions because of the prevailing uncertainty about the outcome at the point of the decision (Sullivan et al., 1999), and literature suggests that female decision makers tend to be more risk averse than male decision makers (Powel and Ansic, 1997). Two, literature also suggests that past experience contributes to increased familiarity in a given problem domain and can affect risk perceptions (Sitkin and Pablo, 1992). Our research intends to solve the conflicting points of view by exploring whether existence of managerial biases varies among different sub populations.

We focus on growth options for this research because they play a significant role in the economic justification of projects embedding them, by allowing decision makers to consider further investment opportunities as well as cash flows (Panayi and Trigeorgis, 1998). Also, decisions involving growth options are often strategic in nature and biased decisions in the context of such options are an important research topic (Tiwana et al., 2006). In addition, prior research has shown that, while valuing projects with real options, managers ascribe greatest importance to projects with growth options (Busby and Pitts, 1997). Similar results hold for IT projects (Tiwana et al., 2006), where IT managers perceive a growth option as adding more value to the project than any other type of option. This perception of growth options as gains might suggest growth options’ vulnerability to the risk preferences of IT managers once looked at through Prospect Theory. Hence we evaluate this research question by focusing on the growth option “exercise decision”. Our results indicate that gender and experience do impact biases in growth option exercise decisions significantly. However, we also observe some exceptions.

The rest of the paper is structured as follows. Next, we briefly describe the relevant literature supporting our research, followed by the hypotheses. This is followed by research design, methodology, preliminary analysis and results, and conclusions.
LITERATURE REVIEW AND HYPOTHESES

Growth Options and Gender Differences

A substantial body of literature has studied gender differences in the context of risky decision making. Findings indicate that females are often more risk-averse than males (Powel and Ansic, 1997; Hudgens and Fatkin, 1985; Johnson and Powell, 1994; Sexton and Bowman-Upton, 1990; Levin et al., 1988). While there is significant evidence in favor of this, some studies also show circumstantial contradictions. One antecedent for this gender difference in risky decision making is related to the strong evidence that men, on average, are more overconfident than women (Barber and Odean, 2001; Beckmann and Menkhoff, 2008).

Lower preference for risk taking among females is one gender difference that is persistently found in the business literature as well as other studies (Powel and Ansic, 1997; Hudgens and Fatkin, 1985; Johnson and Powell, 1994; Sexton and Bowman-Upton, 1990; Levin et al., 1988). Eckel and Grossmann (2001) find significant gender differences in choices among several risky prospects, with females indicating a preference for the less risky prospect. Also women show lower risk propensity than men in their attitudes towards financial risks (Barsky et al., 1997). Fellner and Maciejovsky (2007) find that females are more risk-averse than males, especially in binary lottery choices. Beckmann and Menkhoff (2008) show that female financial experts are more risk-averse than their male counterparts.

While studying risk propensity in financial decision making, Powell and Ansic (1997) show that females are less risk-seeking than males, irrespective of framing of decisions, familiarity with the scenario, and cost or ambiguity associated with the decision. However, Schubert et al., (1999) show that females are more risk-averse than males in gambles framed as gains. Although Schubert et al. (1999) find gender differences in abstract gambling decisions, the differences disappeared with the introduction of an investment decision context. Kruse and Thompson (2002) also find no significant differences between men and women in low probability loss situations. Similarly, Schubert et al. (2000) find weak differences under two different formats of ambiguity but again no differences under risk.

Since growth options are like call options (Benaroch, 2002) and call options have been shown to be framed as gains (Miller and Shapira, 2004), based on the above studies we expect that females will be more risk-averse while exercising growth options. Hence, we hypothesize that:

Hypothesis 1: For growth option exercise decisions, female decision makers will take more suboptimal decisions than males.

Growth Options and Experience Differences

Experience is considered a primary way for decision makers to learn, and affects their response to similar as well as new situations (Levitt and March, 1988, Mintzberg et al., 1976). Learning occurs since decision makers tend to conceptualize a problem in terms of a most recognizable situation and try to solve it using familiar preexisting solutions (Cohen, March, & Olsen, 1972; Dearborn & Simon, 1958; Langer, 1978; March and Simon, 1958). Past experience is also shown to result in overconfidence among decision makers due to intentional ignorance of underlying risks (Lewin, 1989; Baird & Thomas, 1985; Langer, 1975; Lefcourt, 1973; Slovic et al., 1980) and illusion of control (Langer, 1975; Lefcourt, 1973). Slovic, Fischoff, and Lichtenstein (1980) argue that the prior experience of decision makers influences risk behavior by encouraging higher levels of confidence in extremely experienced individuals. However, the lack of relevant experience is shown to lead the decision maker towards incorrect assumptions and diagnoses, even in the face of well-established operating procedures (Jemison & Sitkin, 1986). Decision makers with limited experience may be more likely to hold risk perceptions that are strongly influenced by available information, because either it is the only data point in that domain, it is recent, or it is obvious (Tversky & Kahneman, 1973), hence making them prone to biased decision making.

Experience has been studied to some extent in IS research (Keil et al., 2000) as well as real options studies (Liu et al., 2010). When decision makers are more experienced, they may be more willing to undertake risks than less experienced individuals (March & Shapira, 1987; Staw & Ross, 1987). Since exercising a growth option is a risky decision, we expect the experience of the IT managers to play a significant role in their decisions. Individuals tend to be risk-averse when a choice is presented as a gain (e.g., a growth option), and experience can help them overcome such tendency. Hence, we hypothesize that:
Hypothesis 2: For growth option exercise decisions, experienced managers will take less suboptimal decisions than inexperienced ones.

RESEARCH DESIGN AND METHODOLOGY

In this study, our motivation is to explore whether managers of a certain gender and work experience are able to fully extract the value of real options (Fichman et al., 2005). In Real Options theory, it is assumed that options are exercised optimally. Of course, a real option is most valuable when exercised optimally (Kumar, 2002). The optimal exercise of a real option is generally conceptualized as taking the exercise decision at a time period (before expiration) when the benefit from exercising the option is highest. However, we lack empirical evidence that options are indeed exercised optimally.

In order to capture gender and experience differences in growth options’ exercise decisions, we conducted a survey among IT professionals by presenting the real option exercise decision for a project with a growth option as a gamble. We designed our survey based on Tversky and Kahneman’s (1981) testing of the “framing of act” because decision making under risk has been conceptualized as choices between prospects or gambles (Kahneman and Tversky, 1979). The closest approximation of capturing such decisions in organizations has been to treat them like a gamble (Kahneman and Lovallo, 1993). A gamble is characterized by uncertainty and the dependency of payoffs on a decision, like a real option exercise decision. The gambling design is a popular experiment design to capture risk preferences in various disciplines, including business, economics (Levin et al., 1998), and dominates framing research (Kuhberger, 1998). Based on prior research, we believe that the gambling design is the simplest, yet most realistic way to represent a real option exercise decision.

The Project Profiles

The respondents were presented with two IT project profiles. For each project, information about the embedded growth option, the size of the project, and risk and return associated with real option exercise decisions were presented. The profiles only differed in terms of project size. For the growth option, the risky outcome was to exercise the option by investing further, with probable higher returns. We incorporated uncertainty by presenting the outcome with a probability. The corresponding riskless decision was set as letting the option expire, leading to a certain but lower outcome. We assumed gains for either outcome, since growth options are valued higher due to the potential for larger future benefits (Miller and Shapira, 2004; Tiwana et al., 2006). However, the future realized benefits are uncertain when exercising the option (Coff and Laverty, 2007). It can be argued that the outcome uncertainty can be reduced by deferring the investment. Our intention is to capture the investment behavior in the absence of such flexibility to delay the investment and to examine the risk behavior of IT managers under such situations. In reality, the flexibility to delay the investment is usually not utilized by firms operating in a competitive market, where growth is vital for their survival (Lankton and Luft, 2008).

Further, testing of risky decisions under Prospect Theory requires the scenarios to be built around a reference point (Kahneman and Tversky, 1982; Kahneman, 2003). For IT managers, the common criteria used to evaluate investment decisions is the project’s NPV (Fitchman et al., 2005; Keil et al., 2007). Hence, we have used project NPV as a reference point. Based on the reference point, the growth options in both project profiles are presented as prospects with possibility of minimum zero NPV.

Uncertainty and Payoffs:

Consistent with reality, we modeled the real option exercise decision as a simple “exercise” vs. “do not exercise” decision. Growth option exercise decisions are usually high risk-high return decisions, where risk is contributed by the uncertainty around future outcomes. For example, exercising a growth option means investing in an ongoing project further with an aim for higher returns. In our survey, each real option decision scenario had a certain

---

16 Framing of act experiment captured risky choice framing (Tversky and Kahneman, 1981, 1986; Kahneman and Tversky, 1979) with implications to Prospect Theory. The design itself is referred to as a gambling design, where respondents are asked to participate in risky decision making. The gambling design is a popular experiment design to capture risk preferences in various disciplines including business, economics (Levin et al., 1998), and real option valuation (Millar and Shapira, 2004), and dominates framing research (Kuhberger, 1998).
outcome and an uncertain outcome. For the growth option, the risky decision was the “exercise” decision, in order to capture the uncertainty associated with further investment in the project. The return from the growth option exercise decision was much higher but with a relatively low probability. Prospect Theory shows that the threshold for risk behavior change is approximately at 50%. Risk seeking behavior in gains is observed for better outcomes with probability less than 50% (Tversky and Kahneman, 1991). Therefore, we chose 25% as the probability that the better probable outcome would happen, in a manner similar to “framing of acts” experiment (Tversky and Kahneman, 1981). We kept the same uncertainty in all scenarios for simplicity and consistency.

We used payoffs data based on real ERP systems cost figures to make them realistic. ERP systems are a good example of IT projects due to their wide implementation. Also these investments are considered important due to the variety of applications that are enabled by the ERP systems. The average costs we found for ERP systems ranged from approximately $0.4 Million to $2.3 Million (Aberdeen Group Inc., 2007). We kept payoffs in all the profiles close to these figures. The details on the project sizes are given below.

Project Size:

To control for the projects’ size, we chose $0.5 million for small projects and $2.0 Million for large projects. All projects were positioned as completed mid-way, where the project progress and resource investment was set at 50%. So the earned value\(^\text{17}\) for small projects became $0.25 Million and $1.0 Million for the large projects. Earned value enabled us to create a suitable decision point in terms of planned value of the projects\(^\text{18}\). Respondents had to decide the future of the projects purely based on the embedded real option. Table 1 gives a breakdown of the payoffs and respective probabilities. Consistent with Prospect Theory, the net payoff difference between risky and riskless options was kept the same in small and large projects (equal to $200,000), to capture the size effects.

<table>
<thead>
<tr>
<th></th>
<th>Exercise</th>
<th>Do Not Exercise</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pr.</td>
<td>Payoff</td>
</tr>
<tr>
<td>Small Projects</td>
<td>25%</td>
<td>$1,800,000</td>
</tr>
<tr>
<td>Expected Value</td>
<td></td>
<td>$450,000</td>
</tr>
<tr>
<td>Difference</td>
<td></td>
<td>$200,000</td>
</tr>
<tr>
<td>Large Projects</td>
<td>25%</td>
<td>$4,800,000</td>
</tr>
<tr>
<td>Expected Value</td>
<td></td>
<td>$1,200,000</td>
</tr>
<tr>
<td>Difference</td>
<td></td>
<td>$200,000</td>
</tr>
</tbody>
</table>

Table 5: Uncertainty and Payoffs used in the Scenarios

Gender:

The respondents were asked to disclose their gender in the survey.

Experience:

We captured the general work experience of the respondents in the survey on a scale of 1 to 5. To further refine the data, we also captured the experience related to IT investment as well as experience with real option decision making.

Control Variables:

We further controlled for the size of respondents’ organization.

---

\(^{17}\) The earned value of a project is the budgeted cost of project multiplied by its completion percentage (Anbari, 2003)

\(^{18}\) The planned value of the project is the value to be earned as a function of project work accomplishment up to a given point in time (Anbari, 2003)
Sample and Data

We pilot tested the decision scenarios using Dun and Bradstreet Executive’s list 2010 (Tiwana et al., 2006; 2007) consisting of top and middle level management in US organizations involved in IT management. A total of 43 usable responses were generated to pre-test the decision scenarios among professionals. Based on the results, no major modifications to the survey were made. The finalized survey was sent out to IT management professionals in the US using Project Management Institute (PMI) US chapters and communities of practice. We received 150 usable responses. The sample size met the requirement based on our a-priori sample size estimation for the study given the effect size, error probability, and power for the parametric data analysis. Prior to running the analyses, all the variables were examined for accuracy of data entry, outliers, and missing values using SPSS v17. All the values for all variables were within acceptable ranges, suggesting that there were no data entry errors. Descriptive data analyses are given in Table 2. Work Experience scale included: 5=Above 20 years, 4=16-20, 3=11-15, 2= 6-10, 1=5 years or less. Firm size scale included: 4=More than $1 Billion, 3= $500 Million - $1Billion, 2=$ 1 Million - $ 500 Million, 1= Less than $1 Million.

<table>
<thead>
<tr>
<th>Scale of 0 and 1</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Mean</th>
<th>Std. Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Project</td>
<td>0</td>
<td>1</td>
<td>.45</td>
<td>.499</td>
</tr>
<tr>
<td>Large Project</td>
<td>0</td>
<td>1</td>
<td>.33</td>
<td>.473</td>
</tr>
<tr>
<td>Female</td>
<td>0</td>
<td>1</td>
<td>.66</td>
<td>.475</td>
</tr>
<tr>
<td>Scale of 1 to 4</td>
<td>Firm Size (in $s)</td>
<td>1</td>
<td>4</td>
<td>2.69</td>
</tr>
<tr>
<td>Scale of 1 to 5</td>
<td>Work Experience (in years)</td>
<td>2</td>
<td>5</td>
<td>4.33</td>
</tr>
<tr>
<td></td>
<td>IT Investment Experience</td>
<td>1</td>
<td>5</td>
<td>3.42</td>
</tr>
<tr>
<td></td>
<td>Real Options’ Experience</td>
<td>1</td>
<td>5</td>
<td>3.49</td>
</tr>
</tbody>
</table>

Table 6: Descriptive Statistics (n=150)

RESULTS ANALYSIS

To examine the unique contribution of gender and experience on growth option decisions, logistic regression was performed using the following logit model:

\[
\text{Option Exercise Decision} = \beta_0 + \beta_1 (\text{Gender}) + \beta_2 (\text{Work Experience}) + \beta_3 (\text{IT Investment Experience}) + \beta_4 (\text{Real Options’ Experience}) + \beta_5 (\text{Firm Size}) + \epsilon
\]

Before running the data analyses, the data were coded in binary, i.e., 1 for the optimal project decision, and 0 for the suboptimal project decision. We ran a frequency analyses on project decisions to see the presence of rational and biased decisions. The test results were significant for both of the projects’ choices under consideration. For the small project, we had 82 (54.7%) biased project decisions and 68 (45.3%) optimal decisions (t-value (149,150) = 11.12, p< .001). For large project, we had 100 (66.7%) biased project decisions and 50 (33.3%) optimal decisions (t-value (149,150) = 8.63, p< .001). This showed the presence of biased decisions at the real option exercise time for both projects.

We first ran the direct logistic regression analysis with small project decisions as a DV (dependent variable) with five predictors, i.e., gender, work experience, experience in IT investments, experience with real options decisions, and firm size. The test of the full model with all predictors against a constant-only model was not statistically reliable ($\chi^2(5, 150) = 1.303, p=.254$), indicating that the predictors did not reliably distinguished between optimal and biased real option choices. The variance in real option choice accounted for was not good, with Cox and Snell $R^2$ equal to .06 and Nagelkerke $R^2$ equal to .08. Predicted success was 76.8 % for the biased real option decisions and 41.2% for the optimal real option decisions, with an overall success rate of 60.0%. Table 3 shows the regression coefficients, Wald statistics, statistical significances, and odds ratios for each of the predictors and control variables for the small project. According to the Wald criteria, gender and IT Investment Experience reliably predicted the difference in growth option exercise decisions. The odds ratio indicated that for every growth option exercise time in a small project, females were 0.442 times more likely to fall prey to biased decision making (they showed risk-
averse behavior). Similarly, the odds ratio for IT investment experience indicated that for every growth option exercise time in a small project, experienced people were 1.428 times less likely to fall prey to biased decision making. In other words, for every unit increase in IT investment decisions experience, people were 14.28% less likely to make biased decisions for growth option.

<table>
<thead>
<tr>
<th>Variables</th>
<th>B</th>
<th>S.E.</th>
<th>Wald</th>
<th>df</th>
<th>Sig.</th>
<th>Odds Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gender</td>
<td>-0.816</td>
<td>0.386</td>
<td>4.475</td>
<td>1</td>
<td>0.034**</td>
<td>0.442</td>
</tr>
<tr>
<td>Work Experience</td>
<td>-0.279</td>
<td>0.180</td>
<td>2.412</td>
<td>1</td>
<td>0.120</td>
<td>0.756</td>
</tr>
<tr>
<td>IT Investment Experience</td>
<td>0.356</td>
<td>0.179</td>
<td>3.955</td>
<td>1</td>
<td>0.047**</td>
<td>1.428</td>
</tr>
<tr>
<td>Real Options Experience</td>
<td>-0.030</td>
<td>0.195</td>
<td>0.024</td>
<td>1</td>
<td>0.877</td>
<td>0.970</td>
</tr>
<tr>
<td>Firm Size</td>
<td>0.109</td>
<td>0.168</td>
<td>0.419</td>
<td>1</td>
<td>0.518</td>
<td>1.115</td>
</tr>
<tr>
<td>Constant</td>
<td>0.142</td>
<td>0.907</td>
<td>0.025</td>
<td>1</td>
<td>0.875</td>
<td>1.153</td>
</tr>
</tbody>
</table>

Table 7: Logistic Regression results – Small Project

We further ran the direct logistic regression analysis with large project decisions as a DV with the same predictors. The test of the full model with all predictors against a constant-only model was statistically reliable, \( \chi^2(5, 150) = 16.01, p<.001 \), indicating that the predictors reliably distinguished between optimal and biased real option choices. The variance in real option choice accounted for was also good, with Cox and Snell \( R^2 \) equal to .172 and Nagelkerke \( R^2 \) equal to .238. Predicted success was 87.0 \% for the biased real option decisions and 48.0\% for the optimal real option decisions, with an overall success rate of 74.0\%. Table 4 shows the regression coefficients, Wald statistics, statistical significances, and odds ratios for each of the predictors and control variables. According to the Wald criteria, gender again reliably predicted the difference in growth option exercise decisions. The odds ratio indicated that, for every growth option exercise time in a large project, females were 0.194 times more likely to fall prey to biased decision making. Based on these results, H1 is supported irrespective of project size. However H2 is supported for only small projects.

<table>
<thead>
<tr>
<th>Variables</th>
<th>B</th>
<th>S.E.</th>
<th>Wald</th>
<th>df</th>
<th>Sig.</th>
<th>Odds Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gender</td>
<td>-1.638</td>
<td>0.409</td>
<td>16.018</td>
<td>1</td>
<td>0.000***</td>
<td>0.194</td>
</tr>
<tr>
<td>Work Experience</td>
<td>-0.339</td>
<td>0.193</td>
<td>3.087</td>
<td>1</td>
<td>0.079</td>
<td>0.712</td>
</tr>
<tr>
<td>IT Investment Experience</td>
<td>-0.237</td>
<td>0.187</td>
<td>1.599</td>
<td>1</td>
<td>0.206</td>
<td>0.789</td>
</tr>
<tr>
<td>Real Options Experience</td>
<td>0.127</td>
<td>0.212</td>
<td>0.360</td>
<td>1</td>
<td>0.548</td>
<td>1.136</td>
</tr>
<tr>
<td>Firm Size</td>
<td>0.000</td>
<td>0.191</td>
<td>0.000</td>
<td>1</td>
<td>0.998</td>
<td>1.000</td>
</tr>
<tr>
<td>Constant</td>
<td>2.104</td>
<td>1.021</td>
<td>4.246</td>
<td>1</td>
<td>0.039**</td>
<td>8.200</td>
</tr>
</tbody>
</table>

Table 8: Logistic Regression results – Large Project

Conclusion and Future Work

Through this study, we explore the relationship between gender and experience differences and the real growth options exercise decisions. By conducting a survey-based online experiment with IT managers as subjects, we simulated growth option scenarios that occur in IT investments. Our results indicate that gender and experience differences do impact growth option exercise decisions, and may help us predict decision makers’ behavior. It turns out that, as expected, female decision makers exhibit risk-averse behavior, which may lead to suboptimal exercise decisions for growth options. This result held in our case for both small and large projects. We do intend to increase the sample size, to ensure better model fit.

Experience also played a significant role in predicting growth option exercise decisions. We found that IT investment experience was significant for small projects. We may conclude from this result that for small IT investments, specific experience relating to IT investments is required to minimize biased decision making. We need
to further examine this effect by exploring the potential interaction effects. Firm size and real options experience did not play a significant role though.

Our study has several implications. First, it moves away from the real option valuation problem into the real options exercise decision problem, which is under-researched. IT real options exercise decisions are challenging due to prevalent uncertainty about commitment to the option. Also, managers are expected to make economically optimal decisions, while taking into account all possible outcomes and future opportunities. Our results provide another dimension to this expectation by illustrating that managers’ personal characteristics, such as gender and experience, play a significant role in option exercise decisions. Moreover, project characteristics play a significant role as well. Greater experience might look like a logical facilitator for the risky economic decisions, however, we find contrary results. Second, we are contributing to the literature on IT project management. Organizations are considering the idea for managing their IT investments using real options. Our results indicate a need to consider gender and experienced-based decision maker differences while interpreting IT option exercise decisions. Third, we try to extend literature on IT investment behavior by studying the effects of gender and experience in a real options setting. Gender and experience differences have been studied in various areas including, but not limited to, general investment behavior, psychology, and financial markets. However, existence of these differences in the context of IT real options has not been studied.

APPENDIX – SAMPLE QUESTIONNAIRE

General Instructions
You will be evaluating four individual IT projects and three IT portfolios in order to determine their future direction.

- Each IT portfolio consists of two IT projects that may or may not be related.
- Each project (individual and in portfolios) that you will be evaluating in this survey, will have either an option to invest further in it for its future expansion (Option to Grow) or an option to kill it before it is completed (Option to Abandon).
- You will be given information regarding expected future payoffs from the project and the uncertainty around these payoffs.

Evaluation of Individual Projects

- All the projects are approximately mid-way in their life cycle i.e. they have spent 50% of their allocated budget and are only 50% complete.
- The estimated net present value (NPV) for each project depends on your decision. NPV of a project is the net future cash inflows of the project, adjusted for the time value of money.
- Based on the information given, please make a decision in terms of exercising the option (described below).

Sample question 1 (from the questionnaire)

<table>
<thead>
<tr>
<th>IT Project Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>This Project has the option to Grow (Further investment in this IT project may enhance future revenues). Your choices are as follows.</td>
</tr>
<tr>
<td>Invest (Exercise the option)</td>
</tr>
<tr>
<td>25% chance of NPV being $1,800,000</td>
</tr>
<tr>
<td>75% chance of NPV being $0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>What would you choose to do for Project 1?</td>
</tr>
<tr>
<td>o Invest (Exercise the option)</td>
</tr>
</tbody>
</table>

REFERENCES


Agile Information Systems Development Teams: Is Empowerment Taken for Granted?

Behnaz Gholami
Institute for Enterprise Systems
University of Mannheim - Germany
bgholami@uni-mannheim.de

ABSTRACT

Agile Information Systems Development (ISD) principles emphasize self-organizing teams and empowered individuals in order to build more effective architecture and design. Agile ISD puts a higher emphasis of social interactions and human aspects of software development, and self-organizing teams act as enablers of these human aspects. In agile ISD teams, members have high autonomy over choosing their tasks and the way they perform the tasks. Team members benefit from the collective decision making and shared ownership of the project. However, in the agile ISD literature the terms “self-organization”, “autonomy” and “empowerment” are using interchangeably and without clear and specific definition. Adapting an interpretive case study design in a leading enterprise software company, this research-in-progress differentiates “self-organization”, and “empowerment” and uses the team-level four-dimensional conceptualization of team psychological empowerment by Kirkman and Rosen (1999) (potency, meaningfulness, autonomy and impact), to study whether self-organization indicates empowerment in agile ISD teams.

Keywords

Agile Information Systems Development (ISD), self-organization, team empowerment.

INTRODUCTION

Agile ISD methods and practices have been widely adopted in the enterprise software industry. The principles of “The Agile Manifesto” emphasize self-organizing teams and empowered individuals in order to build enterprise systems more effectively. Self-organizing teams have been associated with high productivity (Cohen and Ledford, 1994; Kirkman and Rosen, 1999) satisfied employees, lower turnover, and lower absenteeism (Cohen and Bailey, 1997; Wall, Kemp and Jackson, 1986), and creativity (Houghton and Yoho, 2005). Self-organizing teams are fundamental features of agility in software development (Cohn, 2010; Hoda, Noble and Marshall, 2013; McAvoy and Butler 2009). Agile ISD teams are considered to be democratic, without a strict hierarchy. The process of decision-making resides now with the team members rather than with the team manager. Team members learn how to work together and how to mutually oversee each other (Barker, 1993). Agile ISD places a greater emphasis on people as well as on social interactions, which requires a higher degree of human empowerment and communications.

An agile ISD team should be able to self-organize its challenges and constraints that have been posed by management (Cockburn and Highsmith, 2001; Cohn 2010; Moe Dingsøyr, and Dybå, 2010; Takeuchi and Nonaka, 1986). In general, a self-organizing team has autonomy over decisions about policies and programs, and is responsible for determining methods, procedures, and schedules and for making work-related decisions (Tata and Prasad, 2004). The label “self-organizing teams” is most widely used as a synonym for “autonomous teams” (e.g., Moe, Dingsøyr, and Dybå, 2008; Pearson, 1992). Some researchers use the terms “empowered teams” and “self-organizing teams” synonymously (Ford, Fottler and Russ, and 1995; Manz and Sims, 1995; Moe et al., 2008), while others differentiate between the concepts (e.g., Kirkman and Rosen, 1999). Kirkman and Rosen (1999) define team empowerment as having four dimensions: potency, meaningfulness, autonomy, and impact. In this study we use the term “self-organizing teams” the same as the term “autonomous teams”. However, the conceptual scope of
“empowered teams” is more specific, since members of empowered teams also share a sense of doing meaningful teamwork towards organizational objectives (Kirkman and Rosen, 1997, 1999; Kirkman, Rosen and Tesluk, 2004).

Researchers try to study and understand self-organization and autonomy in agile ISD teams (Moe et al., 2008, 2010; Vidgen and Wang, 2009). However, a thorough analysis of team empowerment in agile ISD teams within four above mentioned dimensions is deemed to be useful. For instance, Maruping and Magni (2012) argue that through an expanded set of responsibilities and expectations fostered by team empowerment climate, team members may experience work overload, thus reducing their likelihood of exploring and learning. Therefore, the main objective of this paper is to develop an understanding of team empowerment in agile ISD teams and addresses the question: to what extent agile ISD teams are empowered?

The remainder of this paper is organized as follows. In the next section, based on the literature, we briefly go through the concept of self-organization and team empowerment and differentiate the concepts. We then present a brief background of self-organization and autonomy in agile ISD teams. Following the introduction of the research design, we then present the preliminarily results of a case study of a multinational software company. Finally, the expected contributions, limitations and further steps are discussed.

THEORETICAL FOUNDATIONS

Self-organizing and Empowered Teams

Self-organizing teams are teams which are able to cultivate their own structure (Bunderson and Boumgarden, 2010). They are usually small groups of people (Hoda et al., 2013; Pearson, 1992; Wall et al., 1986) who share highly related and interdependent tasks (Guzzo and Dickson, 1996; Pearson, 1992) and have both authority and responsibility over their work (Pearson, 1992) and decision making (Cohen et al., 1997; Guzzo and Dickson, 1996; Hoda et al., 2013; Hoegl and Parboteeah, 2006; Kirkman and Rosen, 1999; Wall et al., 1986). In self-organizing teams, co-workers have freedom to schedule their tasks and allocate their jobs (Hoda et al., 2013; Morgan, 2006). Self-organizing teams are responsible for variety of tasks (Pearson 1992) and consist of various specializations and different skills (Hoda et al., 2013; Pearson, 1992; Takeuchi and Nonaka, 1986; Wall et al., 1986). Self-organizing teams are identified as social networks (Barker, 1993; Guzzo and Dickson, 1996) and learning systems (Hoda et al., 2013; Morgan, 2006) in an organization. To expand their view for upcoming decisions, such teams need to get feedback from peers and management to assess the goal attainment and learn from everyday work (Hoda et al., 2013; Morgan, 2006; Pearson, 1992). Some researchers associate team empowerment with self-organizing teams (e.g., Manz and Sims, 1995) but others distinguish the concepts (e.g., Kirkman and Rosen, 1997, 1999).

The concept of team empowerment has widely been investigated in management and organization research as well as psychology (e.g. Hempel, Zhang, and Han, 2012; Honold 1997; Kanagaretam, Mestelman, Nainar, and Shehata, 2012; Kirkman and Rosen, 1999; Kukenberger, Mathieu, and Ruddy, 2012; Mathieu, Gilson, and Ruddy, 2006; Maynard, Gilson, and Mathieu, 2012; Spreitzer, 1995). Basically, there are two different approaches to study empowerment in the literature: structural and psychological (Mathieu et al., 2006; Maynard et al., 2012). On one hand, structural approach focuses on delegation and transition of authority and responsibility from management to employees (Mathieu et al., 2006; Maynard et al., 2012). On the other hand, the psychological approach is concerned with psychological states, cognitions or perceptions of employees that they have control and competence over their job (Honold, 1997; Mathieu et al., 2006; Maynard et al., 2012). Both concepts have been examined at both individual and team levels (Mathieu et al., 2006).

Another version of psychological empowerment is defined by Spreitzer (1995, 1996) in terms of a four-dimensional framework of employees’ perceptions: (a) experiencing a sense of value and importance toward their work (b) competence and capability to perform their job well, (c) self-determination or having choice over how to carry out their tasks, and (d) belief that their work has an impact on the organization at strategic and administrative level. Building upon this work, Kirkman and Rosen (1999) advanced the definition of team empowerment as having four dimensions: potency, meaningfulness, autonomy, and impact (Maynard et al., 2012). Potency is the “collective belief of a team that it can be effective” (Kirkman and Rosen, 1999 p. 59). Meaningfulness refers to a team experiencing its tasks as important, valuable, and worthwhile (Kirkman and Rosen, 1999). Autonomy is the degree to which a team manages its own operations, processes, and decisions without a lot of control or guidance from the team advisor (Kirkman and Rosen, 1999; Lewis, 2003). Impact refers to significance and importance of work a team produces for an organization (Kirkman and Rosen, 1999). Team psychological empowerment is not only the result
of team having autonomy and control over their task but also the result of members’ perceived authority and responsibility (Maynard et al., 2012). In fact, in their meta-analysis Seibert et al. (2011) showed that structural empowerment is an antecedence for psychological empowerment and there is a positive and significant relationship between them. The team-level four-dimensional measure has largely been used and supported by many analysis (Maynard et al., 2012).

**Agile Software Development and Self-Organizing Teams**

Successful agile teams benefit from team empowerment (Dybå and Dingsøyr, 2008; West and Grant, 2010). Besides facilitating software development practices, self-organizing teams are also enablers of human and social aspects of software engineering (Hoda et al., 2013). These social aspects are, for instance, emotional attachment to the organization, greater commitment and motivation to perform and desire for responsibility (Hoda et al., 2013; Moe et al., 2008). Moreover, describing an agile ISD team as a complex adaptive system, Vidgen and Wang (2009 p. 358) defined self-organization as “the ability of interconnected autonomous agents of a complex adaptive system to evolve into an organized form without external force”. The ability to intervene and take the appropriate action is what Vidgen and Wang (2009) called “autonomy” of agents in an agile ISD team. Furthermore, Lee and Xia (2010) reviewed the literature on agility and autonomy in software development research. The literature review shows that one of the most important characteristic of agile teams is self-organization and empowerment (Dybå and Dingsøyr, 2008; Lee and Xia, 2010).

Not only in organizational research (e.g., Pearson, 1992; Wall et al., 1986), but also in agile ISD research (e.g., Lee and Xia, 2010) the terms “self-organizing”, “autonomous” and “empowered” teams were used interchangeably. Self-organizing and empowered teams are both autonomous, but the members in empowered teams experience a sense of effectiveness and meaningfulness in their work towards organizational objectives (Kirkman and Rosen, 1999). In agile ISD, self-organization was defined as the spirit of shared ownership (Cao et al., 2009) and the degree of independence and freedom granted to the team for scheduling the work, selecting and assigning the tasks and decision making (Lee and Xia, 2010). Nevertheless, on one hand, self-organization is most analogous to only one dimension of psychological empowerment (autonomy). On the other hand, measures of self-organization and autonomy are very similar in the literature (Kirkman and Rosen, 1999). Therefore, in this paper we use the term self-organizing and autonomous teams as the same, and raise a question about whether self-organization indicates empowerment in agile ISD teams since the concern about being self-organizing is still worth to study in agile ISD teams. Merely delegating responsibility to the team and giving team members freedom and autonomy for performing their tasks and making team-related decisions (i.e., structural empowerment) is not enough. For example, as Vidgen and Wang (2009) reported, over-communication between team members, over-reliance on informal communication and collaboration, over-reliance of common sense or wrong management practices are, in fact, the inhibitors of agility in self-organizing ISD teams.

**RESEARCH DESIGN**

**Research Context**

This study is being conducted in a leading software corporation in which agile development and practices has been adopted for four years in order to increase the efficiency in developing enterprise software products. Scrum is chosen as the main agile development method in the company but it is tailored from team to team and various subsets of agile practices are adopted by teams. This research is embedded in a larger research program aiming at understanding and improving the process of ISD within ISD teams. Headquarter of the company is located in Europe with several locations all around the world. The company values people empowerment, respects team autonomy, and believes in open environment where people can easily speak up, trust each other and share their knowledge. Teams are self-organizing and empowered to selectively implement or change particular techniques according to their own need and situation. Team members have autonomy over choosing their task and how to perform the task. However, the company still has difficulties empowering employees and facing reluctant employees to take over their responsibilities. Despite having multi-skilled and highly expert employees, the managers are still challenging with how to lead the self-organizing ISD teams in the company to give them the autonomy while guiding them within company’s strategy and objectives. Moreover, results from employee satisfaction and empowerment in the company show that some aspects of employee empowerment are missing and the team-level
four-dimensional framework seems an appropriate framework to consider the HR department and managers concerns about empowerment in the company.

Data Collection and Data Analysis

This study is based on a case design (Yin, 2008). We follow interpretive case study guidelines (Sarker and Sarker, 2009 pp. 445–446) to examine and make sense of the data. Teams were chosen according to selective sampling (Coyne, 1997; Glaser, 1978). Selective sampling refers to “the calculated decision to sample a specific locale according to a preconceived but reasonable initial set of dimensions (such as time, space, identity or power) which are worked out in advance for a study” (Glaser, 1978 p. 37). At the time, the agile team was suggested by the senior staff and one of the managers who was responsible for agile process development and learning in the company. The team was responsible for developing and maintaining a banking information system. The team was working in scrum mode since it was assigned two years ago.

The data collection took place in the period between November 2012 and January 2013. During this period the team was at the final phase of the project and maintenance. The data were collected as part of a larger research program on leadership and learning in agile ISD teams. The data collection and analysis consist of two phases. The first phase data collection and analysis intended to address a question about areas in which a leader can influence an agile ISD team learning (cf. Gholami and Heinzl, 2013). The current research question on team empowerment in agile ISD teams emerged from data analysis of the first phase. Since the team meetings were conducted in a non-English native language, a native research assistant visited team meetings during the observation period. She participated in some daily meetings, planning meetings, review meetings and retrospectives during two sprints. The semi-structured face-to-face interviews generally lasted between 40 to 70 minutes (see Table 1). The interviews were recorded or noted and then transcribed. First author was also provided access to relevant internal documents (e.g., internal wiki pages, company survey results on employee empowerment, etc.). During the first phase team members were asked about their job and the team environment, how they perform their tasks, share knowledge, collaborate, manage the conflicts and decide within their team. The second phase is planned to directly address team empowerment in more agile ISD teams in the company.

<table>
<thead>
<tr>
<th>Team members</th>
<th>Product Owner</th>
<th>Scrum Master</th>
<th>Developer</th>
<th>Translator</th>
<th>Total # of interviews</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>12</td>
<td>1</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 1. Number of Interviewees

PRELIMINARY RESULTS

During the open coding process and referring to the observations, we found that there are some obstacles regarding the knowledge sharing and decision making within the team. Team members did not seem to be highly motivated to learn and collaborate. During the interview, some team members explicitly complained about low autonomy despite being labeled as a self-organizing team. This caused the author to shed more light on the concept of empowerment and on the difference between self-organizing and empowered ISD team.

The preliminary analysis of the first phase interview data and observations indicate that despite their two years experience of agile ISD, team members feel less empowered and more under pressure. According to some of team members, not only was self-organization unable to engender a sense of freedom and meaningfulness; it also implied a sense of being monitored and over-loaded with responsibilities and tasks. This is consistent with Kirkman et al. (1996 p. 49) findings about employees’ concern about the “desirability of team-related assignments” and about the possibility of “increased confrontations with coworkers” while delegating freedom and responsibility to the team. The preliminary findings show that the four-dimensional team empowerment can be affected by two main factors: team leadership and team task.

Team Leadership

The formal leaders of the team were the Scrum Master (SM), the Product Owner (PO). The SM was also a developer and the Project Leader (PL) in the team. The role of a PL is not formally defined within the company and team.
members did not know why both roles as SM and PL are defined for their team. The SM defined her role of PL as being responsible for system delivery. Besides having two leadership roles for the SM, there seemed to be confusion between roles and responsibilities of the SM (and the PL) and the PO. The relationship between the PO and the rest of the team was not well established and team members believed that the PO does not transfer and clarify customers’ need. However, some team members believed that the SM as a PL have a closer relationship to the customers. This ambiguity among roles and responsibilities led to some hesitations about the extent that team could be effective to satisfy customer needs (potency). Furthermore, during daily and sprint planning meetings team members were able to take responsibility over each task. It was the responsibility of the SM to make sure that all tasks were assigned to a responsible expert and to prioritize the tasks according to wishes of the PO. Keeping track of the task improvements and problems helped the team to assess the team’s ability to perform the task at the end of each sprint (potency).

The team shows quite high level of autonomy and freedom over choosing the tasks during the sprint planning meetings and over decision makings during daily and review meetings and retrospectives. Rather than single and central decision making, the team benefits from shared decision making and diffused responsibility. The PO’s role in the team was to exert autonomy from external management and the SM’s role was to facilitate internal autonomy within team members. Although the PO believed that the team is highly self-organizing and does not need much of management guidance, the rest of the team believed that the PO was lacking leadership capabilities and is not able to guide them (autonomy). Furthermore, some team members believed that despite of his lack of guidance, the PO expects them to respond quickly to the urgent customers’ needs and to be flexible regarding changes in work strategies or processes (autonomy). Also, since there is more than one formal leader within a team, some members felt that they need to have one of those leaders (e.g., PO, SM, or PL) by their side in order their voice to get heard.

Additionally, the PO had one important role to exert the feeling of impact to the team. Basically, the extent customers are satisfied with the product is a way which helps the team to evaluate whether their task has a positive impact on company’s customers. This belief mainly originated not from the PO but from the efforts of the SM (in both roles) who was a very experienced person. During two years of close collaboration with the team in scrum mode, the SM was successful to establish the feeling of value within the team (meaningfulness), and to convince them that the project matters to the company (impact). Moreover, sprint review meeting and retrospectives facilitate close collaboration between the teams and the customers and enable them to receive necessary feedback (impact).

**Team Task**

Multi-functionality of team members (Vidgen and Wang, 2009) is one of the characteristics of self-organizing teams (Hoda et al. 2013; Pearson 1992; Takeuchi and Nonaka 1986; Wall et al. 1986). Team members are supported not to be specialized only for specific tasks but gain new skills to do different tasks (Gholami and Heinzl, 2013). The team, believed that there are enough sources of information within the team and the company to get the information needed to do the task (potency). Agile practices such as pair programming helped the team to have more than one expertise on each topic so others can manage to handle the task and maintain knowledge (potency) in case somebody leaves the team temporarily or permanently. Moreover, according to relatively routine task which the team was responsible for the last two years, team members were confident in the team that they are able to perform the task (potency).

The team had autonomy over task design, task schedule and task allocation. Team members ensure quality of their task. The team can select different ways to do the team work (autonomy). However, as a scrum team, team members believed that they do not have much autonomy over choosing their tasks since skills are clear and the task is automatically assigned to the person who has is specialized for particular task. Hence, multi-skilling is not very much considered. One of the most important inhibitors of both autonomy and meaningfulness in the company since introduction of agile ISD is time pressure. Team members believed that they are very much under pressure to do the backlog items during one sprint and autonomy does not make much sense while the goal is only to finish the task. There is also the impression that scrum is a management tool which divided the task into smaller pieces and does not give room for meaningful activities like learning and innovation (meaningfulness).
However, the team believed that they are a unique team and their task is unique for the company (impact). Team members believed that what teams does is not under scope of any other scrum team in the company and they share a sense of difference and pride (impact). Table 1 shows some examples of open coding.

<table>
<thead>
<tr>
<th>Potency</th>
<th>Example Interviews</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>SM: In some cases, the information we need is not properly transferred to the team by the PO and this may cause problems for us to do the task well.</td>
<td>Developer: In the team one can get the knowledge he or she want to do the task since there are several sources of information open to the team and offered also by the company.</td>
<td>Developer: With our team often the topics go from sprint to sprint and they are not getting finished. But I know that there are other teams who try and finish everything in each sprint. Do everything by time. Their [customers] requirements and feedback are not communicated to us properly.</td>
</tr>
<tr>
<td>Developer: There is a person in the team who never comes to the meetings and barely talks; this may affect our feelings as a team. I told this indirectly to the SM.</td>
<td>Developer: [For learning and innovation] we don’t have enough motivation. I think it is because of the task of the team which is more about maintenance and gives the impression that our job is not any more interesting and important.</td>
<td>Developer: We are not a very high performance team but I think what the team does is important for the company.</td>
</tr>
<tr>
<td>Developer: The decisions are just made by the PO, the SM and maybe the architect and we just should do it. For my previous role [in traditional development model] I had more autonomy over my tasks and decisions. In this mode of development, we should have a powerful person by our side to have the chance that our decision is taken into account.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Developer: With our team often the topics go from sprint to sprint and they are not getting finished. But I know that there are other teams who try and finish everything in each sprint. Do everything by time. Their [customers] requirements and feedback are not communicated to us properly.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 2. Coding Examples**

**EXPECTED CONTRIBUTIONS**

These initial findings can extend our understanding of team empowerment in agile ISD teams. First, depending on various factors such as the characteristics of team task, stage of the project, team leader behavior, an agile ISD self-organizing team might experience different levels of empowerment. Second, rooted in team effectiveness research, these findings extend recent works on team empowerment in agile ISD by showing that there are several factors that should be taken into account in order to better understand and predict team empowerment. Previous studies on team empowerment confirmed that the two categories (team leader and team task) are antecedences of team empowerment (Maynard et al., 2012).

Also, it is important for managers to know that implementing agility into software development does not necessarily enable team empowerment in agile self-organizing teams. This in depth study of team empowerment gives more insight to managers about the aspects, new opportunities, and behaviors of self-organization and empowerment in agile ISD, which have hitherto been completely or partially ignored or taken for granted.

**LIMITATIONS AND FURTHER STEPS**

The results of this study have to be studied in other scrum teams. There was another important management role as a Line Manager. During the interviews, the important role of Line Manager emerged. However, it was not possible to
interview the Line Manager during the data collection period. As mentioned the main aim of the study was to address team learning and leadership in the team and the issue of empowerment emerged from the interviews and some informal talks with managers and the Human Resource department staffs. More data is needed to address team empowerment directly within other teams to enable us both within and cross case analysis.

Analysis of archival data on team empowerment and observing and interviewing more teams in the company are the next steps of this research. For interviews during the next, I intend to use the interview guide by Kirkman et al. (2004) phase.

REFERENCES


The Impact of Group Cohesiveness on Decision-Making Outcomes under Conditions of Challenge and Hindrance Time Pressure

Garry Lohan
Lero, NUI Galway
Garry.lohan@nuigalway.ie

Tom Acton
Lero, NUI Galway
Thomas.acton@nuigalway.ie

Kieran Conboy
Lero, NUI Galway
Kieran.conboy@nuigalway.ie

ABSTRACT

Group decision making is increasingly important for the successful completion of software development projects. Group oriented development approaches such as agile methods, which emphasize a sense-and-respond approach are becoming an integral part of software development. These methods are being used by an increasing number of organizations as a means of improving the agility and quality of the development process, and within these processes groups are increasingly involved in critical decision making. Groups are required to make regular group decisions and group members work closely with each other to develop software in time-boxed iterations. However, the literature lacks a clear understanding about how varying degrees of time pressure affects the decision outcomes of the development groups. As group cohesion is viewed as the most fundamental issue facing group decision-making processes, in this research-in-progress paper we develop a research instrument to measure the impact of time pressure and group cohesion on decision-making outcomes.

Keywords

Decision Making, Group Cohesion, Time Pressure, Agile, Software Development, Group Decision Making

INTRODUCTION

As information becomes more readily available and group decision-making scenarios become increasingly complex, two major streams of research have emerged from the literature. Both streams aim to support the decision-making process, albeit by entirely different means. The first of these is the decision support system (DSS) literature with a focus on the technology used to support a decision-making environment (Jensen, Lowry, Burgoon and Nunamaker 2010). Technological advances in systems design, artificial intelligence and machine learning have led to the creation of DSS that not only use data to recommend optimum decisions but can also learn and grow in a dynamic manner (Nissen and Sengupta 2006). The second major stream of decision making focuses on the behavioral features and cognitive abilities of the decision makers. Making informed decisions involves the gathering of all pertinent information and the processing of that information to establish an output choice from a number of possibilities. While there have been major technological advances in DSS, the behavioral side of the decision making process lags behind the technology, with many reporting that decision makers often do not use the technology at their disposal (Appelt, Milch, Handgraaf and Weber 2011; Kayande, De Bruyn, Lilien, Rangaswamy and van Bruggen 2009). It is imperative that researchers continue to gain greater understanding of the human behavioral aspect of the decision-making process. Humans and software agents complement each other during the decision-making process; however, the technological advances made may be underutilized if the human characteristics of the decision-making process are not fully embraced (Grudin 2002; Nissen et al. 2006).

This research-in-progress paper focuses on the behavior aspect of decision making, in particular, we examine group decision making. Group dynamics are ingrained in human nature through millions of years of evolution and these dynamics are unsusceptible to change and inaccessible to conscious awareness (Grudin 2002; Kock 2009). Given these arguments, DSS must accommodate the way we act naturally and group decision support systems (GDSS) will be of little value if group dynamics or group cohesion is not considered as an antecedent to their employment. Much of the extant research within group decision making highlights the positive effects of GDSS on decision outcomes,
such as improved group cohesion (Dennis and Garfield 2003; Dennis and Wixom 2001). However, the premise for this study is that GDSS, while valuable in their own right, are of secondary importance to the cohesion of the group itself. That is, group cohesion should be antecedent of GDSS use rather than an outcome and group cohesion is an important factor in group decision making irrespective of GDSS design or use. In this study we develop an instrument to measure the effects of group cohesiveness on the relationship between time pressure and group decision outcomes for software development groups.

**Group Decision Making**

Many critical strategic and operational decisions for an organization are group based. As groups and group work begin to become a more fundamental part of organizations (Mathieu, Maynard, Rapp and Gilson 2008), modern environments requiring operational agility utilize group-work as a means of achieving flexibility while remaining economically efficient (Tannenbaum, Mathieu, Salas and Cohen 2012). Within the management and organizational literature there is a general trend towards more empowering and autonomous groups and a move from command and control to member-led leadership. More shared leadership within the group is driven by the belief both that self-management is motivational, empowering and engaging and also by the economic need for leaness (Tannenbaum et al. 2012). As groups become more self-managing they begin to take on greater decision-making responsibility. The decisions made by the group will often have a critical impact on both the project success and other less tangible aspects of the development environment, such as group learning (Brodbeck, Kerschreiter, Mojzisch and Schulz-Hardt 2007), satisfaction (McNamara, Dennis and Carte 2008), mental workload (Speier and Morris 2003), participation (Yoo and Alavi 2001) innovation (De Dreu and West 2001), and creativity (Watson, Kumar and Michaelsen 1993).

Although under-researched in information systems development (ISD), fields such as psychology have recognized the importance of group cohesion and work groups have long since been designed with cohesiveness in mind (Campion, Medsker and Higgs 1993; Campion, Papper and Medsker 1996). Several theories have been proposed that address the group cohesiveness aspect of group decision-making, arguing that the cohesiveness of the group will impact the decision-making outcomes (Beal, Cohen, Burke and McLendon 2003). Leadership-confidence theory posits that groups with an assertive-directive member form decisions based on the choices of the assertive-directive member. Other research of a similar vein into group decision-making shows that decision making outcomes tend to polarize around individuals demonstrating greater assertiveness (Pruitt 1971). On the other hand, groups composed of flexible-cohering members and without an assertive-directive member have a tendency to converge towards the group average. This is similar to groupthink, defined by Janis (1972) to be “a mode of thinking that people engage in when they are deeply involved in a cohesive ingroup, when members' striving for unanimity override their motivation to realistically appraise alternative courses of action”. Work on group cohesion by Harrison et al. (1998) shows that group cohesion is also impacted by surface-level (demographic) and deep-level (attitudinal) diversity within the group. As the group work together over time, the effect of the surface-level diversity is weakened while the effect of the deep-level diversity is strengthened. Given that group cohesiveness is the most fundamental issue facing modern software development processes (Gartner 2009), researchers have called for more studies to examine how group cohesiveness will affect decision outcomes of software developer groups, particularly under conditions of time pressure (Drury, Conboy and Power 2012; Moe, Aurum and Dyba 2012).

**THEORETICAL DEVELOPMENT**

**Time Pressure**

The ability to make quick decisions and take fast actions is usually seen to be beneficial to organizations operating in changing environments(Forbes 2005). Previous studies show that quick decision making helps individuals and groups improve information processing and coordination (Kerstholt 1994). Others highlight the economic benefit to rapid decision making (Baum and Wally 2003). However, rapid decision making has also been shown to have a negative affect on decision outcomes (Waller, Zellmer-Bruhn and Giambatista 2002). The tendency to rely on past decision making strategies is greater when there is time pressure to make quick decisions (Perlow, Okhuysen and Repenning 2002). This can result in the same mistakes being made and learning being inhibited. Decision speed or time pressure is also known to impact decision outcomes. As shown by Perlow (2002), decision speed helps some groups but hinders the performance of others. The impact of decision speed is not fully understood and there appears to be a trade-off between decision speed and high quality decision-making. Despite the importance of group decision making within organizations and the current trend towards speedy decisions, there is a paucity of literature
examining the affects that both group cohesion and decisions speed have on the decision outcomes (El-Shinnawy and Vinze 1998). Research shows that group characteristics and cohesion will have an important impact on decision outcomes, yet, as highlighted earlier, research tends to focus on technology aspects and task complexity rather than group characteristics (Appelt et al. 2011) and group decision-making is one of the under researched areas within both the general decision-making literature and the information system development (ISD) literature in particular. Previous research also shows that group cohesiveness may be detrimental to the group decision-making process and therefore negatively impacting group decision-making outcomes. Theories such as groupthink suggest that group cohesiveness is counter-productive and is not conducive to optimum decision-making. However, there are a number of reasons to suggest that this may longer be the case. Research on group decision-making highlights the interplay between task complexity, collaboration system usage, the decision making environment and group composition, noting the affects these constructs have on decision outcomes (Nunamaker, Dennis, Valacich, Vogel and George 1991). Given the developments in technologies, decision support systems and collaboration tools, the decision making process has changed considerably over the past decade. Newer tools help improve the transparency of the decision-making process, therefore issues such as groupthink may no longer have the same negative impact they had when they were first introduced into the decision-making literature.

In complex tasks, groups can become more concerned with reaching a consensus or reaching a decision quickly and less concerned with other goals such as decision outcomes or systematically evaluating alternatives or more creative decisions (Kelly and Loving 2004). Measuring time pressure and decision speed has received a lot of attention in both the management and management information systems literatures (Ancona, Goodman, Lawrence and Tushman 2001; Arrow, Poole, Henry, Wheelan and Moreland 2004; Mitchell and James 2001; Saunders and Ahuja 2006; Street and Ward 2012; Zaheer, Albert and Zaheer 1999). One interesting area is the study by Zaheer et al. (1999) who discuss the circadian rhythm of time intervals. Circadian rhythm, as distinguished from its biological science definition, is defined by Zaheer et al. (1999) as the pattern of variation in business activity levels that occurs over a period of 24 hours and shows the differences in analysis that may be achieved using different time scales. Therefore it is important to consider the time intervals to be used in this study and their relevant impact on the outcome of the research. Zaheer et al. (1999) conceptualize five time scales associated with the phenomena under study.

- The existence interval: The basic type of time scale for one instance of the process, pattern phenomenon, or event to occur
- The validity interval: The interval that defines the time scale over which the theory holds
- The observation interval: A researcher-defined period of observation required to develop the theory
- The recording interval: The granularity of the data recording (collect data every month, day, hour, minute etc.)
- The aggregation interval: The time scale the recorded information is to be aggregated for testing theory about the phenomena

Given that the existence interval constrains the requirements of the other intervals, this puts the study of the time of decision outcomes in an ISD environment at an advantage (i.e. because we can study groups with this defined real world interval, scrum meetings, planning, review and retrospectives). Because we can observe more than the existence interval we have a greater probability of covering variability within the model. Time pressure is often operationalized using two levels (low v high) (Bowman and Wittenbaum 2012; Hwang 1994) however for this study we use the terms challenge and hindrance rather than low and high as indicators of time pressure. While studies show that an increase in time pressure is likely to lead to poorer decision performance, increased time pressure is also often associated with increased or better performance (Hwang 1994; Maule, Hockey and Bdzola 2000). Therefore challenge and hindrance time pressure are viewed more suitable to indicate decision-making performance. The two indicators have been adapted from a previously validated instrument (Chong, Van Eerde, Chai and Rutte 2011). Challenge time pressure is the degree to which a group perceives time pressure as a stressor which promotes goal achievement, while hindrance time pressure is the degree to which a group perceives time pressure as a stressor which constrains goal achievement (Chong et al. 2011).

**Group Cohesion**

Many meta-analyses have been published on the cohesion–performance relationship (Beal et al. 2003; Carron, Brawley, Bray, Eys, Dorsch, Estabrooks, Hall, Hardy, Hausenblas and Madison 2004). The general conclusion stemming from these quantitative studies is that the correlation is moderate, positive, and highly dependent on intragroup processes (Chiocchio and Essiembre 2009). However, a study by El-Shinnawy et al. (1998) found that
group cohesion had no impact on the decision outcome. Their study does not rule out the importance of group cohesion as they control for factors such as group size and history and call for future research to further examine the group cohesion construct. So while early work on group cohesion revealed no relationship between group cohesion and group performance (Deep, Bass and Vaughan 1967), recent work has found that there is, indeed, a relationship between group cohesion and task performance with members of established groups formulating varying levels of cohesion over time (Schwarz and Schwarz 2007). Others have used the group attitude scale (Evans and Dion 1991) to measure group cohesion and the results indicate a positive impact it has on group consensus (Yoo et al. 2001) and user satisfaction of group support system technology (Chidambaram 1996). Schwarz and Schwarz (2007) show that group cohesion predicts enjoyment and effectiveness but did not have an impact on the efficiency of the group. Efficiency was measured by the time it took to come to a decision. In an ISD environment, groups have a deadline in which to arrive at a decision. However, there is a scarcity of literature examining the impact of time pressure on decision outcome and the moderating effect group cohesion has on this relationship. In this study we view cohesion as a multi-component construct consisting of four sub components: social cohesion, task cohesion, perceived cohesion and emotional cohesion (Forsyth 2006).

**Decision Outcomes**

Measuring decision outcomes is difficult because it is difficult to determine the optimum decision outcome in differing scenarios. Some studies use simplistic measures such as the group is either right or wrong based on previous experience or on expert judgement techniques. Most of these studies are, however, experimental and are carried out under controlled laboratory conditions. Other real world studies have used outcomes such as, consensus (Cooper and Haines 2008; Yoo et al. 2001), quality (McNamara et al. 2008), learning (Brodbeck et al. 2007), satisfaction (Dennis 1996), mental workload (Speier et al. 2003), participation (De Dreu et al. 2001), innovation (De Dreu et al. 2001), accuracy (Speier et al. 2003) and creativity (Watson et al. 1993). Others used outcome measures such as process improvement, commitment, creativity, pro-activity, efficiency, effectiveness, confidence, initiative, time taken to internalize role, supervisor ratings, feedback improvement and identity. However, two major indicators of decision outcomes that emerge from the literature are decision confidence (Schwarz et al. 2007) and decision consensus (Salisbury, 2002; Yoo et al., 2001; Cooper & Haines, 2008). Decision confidence relates to how the group view the choice they have made. It is often not be possible to measure the actual final outcome of that decision but measuring the decision confidence should provide a good indication about how positively the group feels about a decision. Decision consensus relates to how the group as a whole understood the reasons for the decision. High consensus will result in complete group buy in and represents high group participation in the decision making process. Based on this theoretical overview we developed our conceptual model for this study (Figure 1).
HYPOTHESES DEVELOPMENT

Insights from goal-setting theory (Locke and Latham 2002) show that challenging goals are a strong motivator and help a group focus its activities on achieving the goal they perceive are attainable. Other studies have shown that challenge time pressure is seen as a positive stressor and enhances group performance (Chong et al. 2011; Chong, van Eerde, Rutte and Chai 2012; Podsakoff, LePine and LePine 2007). Under conditions of challenge time pressure (i.e. a pressure that promotes goal achievement) we hypothesize that groups will work with each other to produce the required decision. Therefore we hypothesize the following:

**Challenge time pressure positively impacts decision outcomes**

Once a group perceives that the goal is no longer achievable within the allocated time, motivation drops and performance suffers. Under conditions of hindrance time pressure (i.e. a pressure that constrains goal achievement) groups will tend to accept the choice of a single group member and polarize quickly around that choice to produce the required decision (Cheng and Chiou 2008). Therefore the majority of the group members will not actively participate in the decision-making process and will be less confident of the decision-making outcome. Bearing this in mind we hypothesize:

**Hindrance time pressure negatively impacts decision outcomes**

*Group cohesiveness* is the extent to which a group is attracted to the group and to each other (Chidambaram 1996). When groups members have a strong attraction to their group they will place emphasis on shared group commitment to tasks and group membership (Beal et al. 2003). This may indicate that, regardless of the time pressures placed on the group tasks, group members will share the commitment to group decisions and outcomes, regardless of the levels of participation in the decision-making process. We therefore hypothesize that:

**High/low levels of group cohesion moderate the impact of time pressures on decision outcomes**

This study is a quantitative study involving a large scale survey to determine the effects of group cohesion on decision outcomes under differing degrees of time pressure. The survey is available online at [www.groupcohesion.com](http://www.groupcohesion.com). The draft questionnaire was pretested for validity with 5 software developers who have experience working in groups within software development organisations. This resulted in some modifications and/or deletions to the questionnaire.

CONTRIBUTION

We intend to make the following contributions through this paper and its subsequent research. Firstly, IT project managers will be interested to understand the effects of time pressure on software development group decision processes. Project managers will be contacted directly or via a third party and asked to complete the survey themselves along with at least three other members of their team. Potential respondents are identified through a database of organisations we know or expect are working with agile methods. The results of this study will indicate how time pressures will affect decision outcomes and the moderating impact group cohesion will have on this relationship. We will measure four components of group cohesion, social, task, perceived and emotional cohesion. Each of these may impact the decision outcomes differently and project managers can use this information to aid them in the design and management of software development teams.

Secondly, we wish to further our understanding of the effects of time pressure on decision-making outcome. Previous results show that time pressure and decision speed is often positively associated with decision outcome. However, practitioners in particular, should note that in certain situations, time pressure will often have a detrimental effect on group decision-making outcome. We will discuss both the positive and negative effects of time pressure on group decision-making outcomes.

Thirdly, we adapt previously used instruments and use them in a time pressured, decision-making environment, exploring the effects of group cohesion on decision outcome under conditions of rapid decision-making. To date, no other study has attempted this important step in developing our understanding of group decision making in rapidly changing environments. We will offer advice on how group cohesion can aid in balancing the negative effects associated with rapid decision-making.
Fourthly, while prior ISD literature has not explicitly defined the optimum group cohesiveness required for ISD projects, we propose that group cohesiveness has an important effect on decision outcome, particularly under conditions of rapid decision response time. We will discuss our findings and offer advice for the ISD field on group cohesiveness for ISD projects.

ACKNOWLEDGEMENTS

This work was supported, in part, by Science Foundation Ireland grant 10/CE/I1855 to Lero - the Irish Software Engineering Research Centre (www.lero.ie)

REFERENCES


Dennis, A.R. (1996) Information exchange and use in group decision making: You can lead a group to information, but you can't make it think, MIS Quarterly, 20, 4, 433-457.


Project Management with a Purpose: Delivering Project Management Education through Service Learning

Christina N. Outlay
University of Wisconsin - Whitewater
Outlayc@uww.edu

R. A. Mendoza
Saint Joseph’s University
RMendoza@sju.edu

ABSTRACT

This research describes the development of an introduction to project management (IPM) course taught in the business college of a Midwestern University. Due in part to an educational initiative launched by the University, the IPM was revised to follow a service-learning model. Following an overview of service-learning pedagogy, related theoretical perspectives, benefits, and practical considerations, we describe the prior learnings on implementing service-learning in project-based courses and in particular, the development of the IPM course and our intentions to develop a model of student outcomes of service-learning for project management education. We collected surveys from 85 undergraduate and graduate students enrolled in the course to examine student attitudes toward service-learning, impacts of service-learning, and adoption of project management career skills due to participation in the service-learning project. We intend to analyze the student data in order to identify the key constructs and relationships that affect learning outcomes.

Keywords

Service-learning, project management education, projects, pedagogy, experiential learning.

INTRODUCTION

Service-learning (SL) courses combine classroom instruction with community service to help students better understand course materials through direct engagement with outside organizations (Tan & Phillips 2005). SL experiences are content-rich, help students develop higher-level cognitive skills, and provide lasting value to students (Brown 2000). SL benefits both the students and the service recipients as part of the learning objectives of the course and differs from other pedagogies by focusing equally on the academic objectives and student learning in the course, and on the service being provided (Furco 1996). Students in SL courses provide services that are relevant and meaningful to the community and enhance academic learning objectives (Howard 2001), and are engaged in purposeful civic learning (Andrews 2007). Students are also provided with structured opportunities for discernment on the impact of their academic efforts and of their service on the recipient communities (Cauley 2001, Rhoads 1998). The community service component of SL courses can help students develop personal and professional skills to better manage the uncertainty and complexity of business operations and of crafting strategic business objectives (Govekar & Rishi 2007, Weis 2000).

Incorporating a SL approach in the classroom potentially provides academic, professional, and personal benefits of across disciplines. Academic benefits include increased student satisfaction, confidence, and motivation to study subject area and higher final course grades Accounting Information Systems (Rose et al. 2005) and in undergraduate-level courses in various disciplines (Berson & Younkin 1998). In Political Science, (Hunter et al. 2000) students reported greater satisfaction with the development of professional skills such as leadership, communications, critical thinking, and conflict resolution. Personal benefits reported by students include greater civic engagement (Astin & Sax 1998, Simons & Cleary 2006).

SL pedagogy has been used across additional disciplines in areas such as sociology (Kendrick 1996), economics (Govekar & Rishi 2007), and health care (Cauley et al 2001). In business curricula, SL has been used in the areas of management (Weis 2000), ethics and leadership (Kolenko et al 1996), accounting (Rose et al 2005, Weis 2000), marketing (Andrews 2007), and information technology/information systems (IT/IS; Hoxmeier & Lenk 2003, Petkova 2012, Preiser-Houy & Navarrete 2006, Rose et al. 2005, Wei et al. 2007), though researchers have noted the relative scarcity of research on the use and success of SL pedagogy in this area (Johnson & Johnson 2005, Wei et al. 2007).
In the area of IT/IS, the SL paradigm has been deployed in support of various topics such as database design and development (Hoxmeier & Lenk 2003, Rose et al. 2005), e-business process modeling and website development (Hoxmeier & Lenk 2003, Preiser-Houy & Navarrete 2006). SL has also been used in systems analysis & project management courses (Hoxmeier & Lenk 2003, Petkova 2012, Rose et al. 2005, Wei et al. 2007), in which students completed analysis and requirements gathering stages, but whose design and implementation stages result in the deployment and use of relatively simple databases by community organizations.

The startup costs of SL courses are significant (Govekar & Rishi 2007), and adapting existing courses to run as SL courses requires a significant investment of time on the part of faculty, who discover there are further changes required after the first offering, implying additional work. Identifying potential clients and negotiating opportunities that are a good match for student skills and client needs can be a time-consuming effort, and working with chronically-underfunded non-profit organizations requires flexibility and adaptability to changing needs and resource levels (Mendoza 2013). The risk to faculty is significant: in addition to partially surrendering control of the learning experience (Stacey & Foreman 1999), student evaluations can be affected by client issues over which faculty have little to no control (Govekar & Rishi 2007). Additionally, IT/IS-focused SL courses are best suited for students with significant experience on the technologies and development processes of information systems (Lazar & Lidtke 2002). This level of expertise is far from the norm in the typical business curriculum. Lastly, faculty involved in or thinking about developing SL courses have to contend with critics of the pedagogy who argue the courses lack clear academic standards or are simply ‘easy As’ (Butler 1994).

SERVICE LEARNING IN IT PROJECT MANAGEMENT EDUCATION

Project management is a core item in the essential body of knowledge of all business students (Ives et al 2002) and is a highly-valued talent consistently listed at the top of employers’ skills lists (Pratt 2012). The Project Management Institute, the world’s leading professional organization for project management professionals, defines a project as a temporary effort to create a unique product or service by the application of knowledge, skills, tools, and techniques to meet requirements (PMI 2013). Business education has a deep tradition of employing experiential learning in the form of project-based curricula (Larson & Drexler 2010) to help bridge the disconnect between the abstract focus of typical business curricula and the applied world of business (Angelidis et al. 2004).

Not all project-based courses can be categorized as SL, and not all SL courses use project-based approaches. Further, the differences between project-based and SL methodologies can be subtle and can be difficult to separate in the literature, but research on these very similar pedagogies points to mixed results in student performance and skills development. Student performance may be more evident in subjective grading categories, such as essays, versus more areas such as multiple-choice exam questions, and in soft skill development rather than academic or career skills (Rose et al. 2005). Students without extensive prior technical grounding may reach “cognitive overload,” and recommendations for a series of project-based courses to evaluate the best combination of learning environments for students ((Barab et al. 2000), are deeply impractical for most curricula.

However, when project-based and SL methods are deftly combined, careful planning, clear milestones, and continual communication with clients contributed to students’ project success (Brown 2000). Project-based and SL courses result in increased contextualization and learner control of the learning process, as well as in more effective combination of interdisciplinary knowledge (Helle et al. 2006). In IT/IS, these benefits are most achievable for Juniors and Seniors with a “thorough understanding” of information systems and their development processes (Lazar et al. 2002). Assigning students specific team roles is a good way for faculty to manage multiple project teams (Tan et al. 2005), but allowing students to manage their own team roles can also work well (Mendoza 2013).

Claims about the benefits of project-based and SL courses abound in the literature, but remain largely unsupported by rigorous theory or by quantitative analysis. A review of over 600 articles on project-based and SL pedagogy in higher education found that the primary focus of the literature is on descriptive approaches to course design and implementation covering various approaches (in-class projects, external projects, projects with external organizations, hands-on software experience, etc.), with very few attempts to ground results on existing pedagogical or other theories, to quantify benefits, or to systematically identify success factors (Helle et al. 2006). Additionally, despite the powerful rationale for the use of SL (Heffernan 2001), few resources exist for faculty interested in developing project-based courses that meet the requisites for SL and have the specific goal of teaching project management career skills.
With these significant concerns in mind, taking a structured approach to identifying target learning outcomes and goals of SL prior to implementing an SL course is key. The theory of change provides valuable guidance that can be applied in this context (Anderson 2006). The theory of change (TOC) describes a process-oriented approach to achieving an identified goal through following several steps: (1) identifying long-term goals/outcomes, (2) mapping requirements or pre-conditions necessary to achieve goal-related outcomes, (3) identifying the environmental context and interventions necessary to meet the requirements, (4) identifying appropriate variables to measure the effectiveness of the interventions, and finally, (6) writing a narrative that describes the logic behind your chosen initiative or program (why it works).

This research contributes to the literature on project-based and SL pedagogies by describing key factors to predict the success of a SL course to teach project management skills at the undergraduate level and to identify and quantify benefits of SL. Our goal is to apply TOC for using SL pedagogy to achieve student learning outcomes in project management education. In doing so, we ultimately seek to develop a model that describes potential learning outcomes, required pre-conditions and interventions, and variables to measure the effectiveness of such interventions at influencing student attitudes toward SL projects, particularly for learning project management. With this in mind, we explore three key research questions:

1. What are students’ attitudes toward completing SL projects for learning project management?
2. To what extent will students apply and adopt project management career skills (i.e., tools and techniques) learned through the SL project experience?
3. What are some impacts of SL projects on students?

INTRODUCTION TO PROJECT MANAGEMENT - DEVELOPMENT OF THE COURSE

Motivation – Identifying Desired Learning Outcomes

The motivation for using a SL approach for the Introduction to Project Management (IPM) course came from two sources. First, the University had recently adopted a major curriculum initiative that included, among other things, widespread adoption of High-Impact Educational Practices (HIPs) within the undergraduate curriculum and wanted to achieve a set of learning outcomes including deeper learning, greater emphasis on community service and richer, more hands-on training in job-related skills. Second, the IPM faculty wanted to provide training that would achieve the University goals as well as ensure that the IPM course would provide relevant skills to non-IT students as well as the IT majors in the Department.

HIPs are guidelines to reinvigorate teaching, learning, and educational experiences to help all students achieve the essential learning outcomes necessary in college learning in the 21st century (Kuh 2008). SL is one of the HIPs included in the initiative. SL, in this context, emphasizes the use of field-based experiential learning with community partners as an instructional strategy. The idea is to give students direct real-world experience with issues they are studying in the curriculum and with ongoing efforts to analyze and solve problems in the community. A key element in SL programs is the opportunity students have to apply what they are learning in real-world settings and to reflect in a classroom setting on their service experiences. These programs model the idea that giving something back to the community is an important product of a college education, and that working with community partners is good preparation for citizenship, work, and life. The SL HIP appeared to be an appropriate intervention to meeting the new learning outcomes desired by the University and IPM faculty.

In addition to changing learning outcomes, the types of students enrolling in the IPM course had changed over time. The University began offering the course in 2007 to business students majoring in information technology (IT). Initially, the majority of students enrolled in the course were IT majors. The course content consisted of project management fundamentals taken largely from the Project Management Body of Knowledge (PMBOK). The PMBOK, developed by the U.S.-headquartered Project Management Institute, is a collection of best practices for managing projects that could be applied across multiple industries (PMI). Although the course content was based on the PMBOK, the overall application of those concepts in the course was solely focused on managing IT projects.

As initially offered, the course had a group project component that was case study-based around an IT project scenario. Group project deliverables consisted of documentation common in the initiation and planning stages of projects. Since the group project was based on case study scenarios, all student learning was theoretical in nature,
and students did not get real-world hands-on experience with each of the project management process stages: initiation, planning, execution, monitoring and controlling, and closing.

Over time, the number of non-IT majors enrolling in IPM increased substantially, in some semesters even outnumbering the IT majors (Table 1). Since the course was growing in popularity for both IT and non-IT majors, the enrollment increases and the rollout of the campus-wide curriculum initiative presented an opportunity to revise the course content to add more hands-on training and development activities, and to provide content that would be more beneficial for both IT and non-IT majors enrolled in the course. The IPM faculty felt confident that adopting SL pedagogy would help with both resolving the theoretical emphasis and providing more hands-on training throughout the PM lifecycle, and providing content for a broader population of students.

<table>
<thead>
<tr>
<th>Term</th>
<th>SP08</th>
<th>FA08</th>
<th>SP09</th>
<th>FA09</th>
<th>SP10</th>
<th>FA10</th>
<th>SP11</th>
<th>FA11</th>
<th>SP12</th>
<th>FA12</th>
<th>SP13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enrollment</td>
<td>26</td>
<td>32</td>
<td>35</td>
<td>32</td>
<td>27</td>
<td>41</td>
<td>64</td>
<td>60</td>
<td>63</td>
<td>57</td>
<td>63</td>
</tr>
<tr>
<td>Non-IT Majors</td>
<td>27%</td>
<td>19%</td>
<td>23%</td>
<td>44%</td>
<td>52%</td>
<td>37%</td>
<td>67%</td>
<td>42%</td>
<td>62%</td>
<td>44%</td>
<td>54%</td>
</tr>
</tbody>
</table>

Table 1. IPM Undergraduate Course Enrollment

Structure of the Course – Requirements, Pre-conditions and Interventions

When designing a SL course, there are three essential stages to consider: the planning of the course, the experience itself, and reflection (Weis 2000). In the planning stage, instructors identify appropriate service partners whose needs match course objectives and student abilities. The experience stage consists of the delivery of services to the client organizations and of classroom material to students. Reflection is the set of structured or unstructured activities that help students process and interpret their experiences. Flexibility and trust in the ability of students to rise to challenges outside the instructor’s control are also key to the success of the project or service experience itself (Weis 2000), as is the instructor’s willingness and ability to adjust to unanticipated developments and to let go of some control (Mendoza 2013).

Planning

During the planning for IPM, the project management faculty developed a new group project that fit well with the SL HIP, aiming to achieve simplicity in the design of activities, superior matching of student capability with project requirements, and direct relevance of project activities to classroom materials that would be critical planning-stage factors to influence student success (Lattery et al. 2001). Careful planning (Brown 2000), clear direction for project objectives, and active support from external partners were also critical at this stage (Kloppenborg & Baucus 2004). Thus, the faculty began by choosing designated community partners to sponsor the student projects prior to the start of the course. The community partners selected strategic objectives students’ projects should support for their organization. Faculty worked closely with the community partners to make sure projects were narrow enough in scope and contained reasonable timeframes to enable students to complete them during the semester, in addition to working on the other course components. Brief descriptions for each available project were provided to students in the course materials.

To allow for flexibility in the students’ choice of projects, the faculty also developed a process to allow students to select their own community partner. Should students choose this option, the students would need to establish a regular contact person at the community organization who would agree to serve as sponsor for the project. The faculty would, in turn, contact this project sponsor to confirm their participation and coach them on the expected duties of their role.

Service-Learning Project Experience

Students were assigned to work in groups for the project. In order to match student capabilities with project requirements, the student groups were required to submit proposals for projects they wished to conduct, including justification for how the project supported the strategic objectives selected by the community partner and the skills and/or prior experience students possessed that qualified them to conduct the project. The project management faculty and community partners evaluated the proposals. Once each student group received notice of the approved
project proposal, they begin actively working to initiate their projects, applying knowledge learned in the course. Students learned the tools and techniques for managing projects in the classroom and applied this learning to conduct their projects throughout the semester. Outside of working on the class project, students completed assigned readings from the textbook, received skills training on project management tools and techniques, and completed in-class activities, quizzes and individual assignments to learn the project management concepts they would apply in their projects. The final outputs of the project were a formal presentation of project outcomes and a project website containing (1) the documentation produced during each project phase and (2) a team report reflecting on their project experience. The community partner also completed a project acceptance/completion form and provided feedback to the project team. Table 2 lists the documents and major tasks completed by students at each stage of the project.

<table>
<thead>
<tr>
<th>Process group</th>
<th>Project Deliverables</th>
<th>Key Project Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposal</td>
<td>Project Proposal</td>
<td>Select desired projects to implement and formally submit the project request and justification.</td>
</tr>
<tr>
<td>Initiation</td>
<td>Project Charter</td>
<td>Clarify project scope and complete tasks to formally begin work on the project.</td>
</tr>
<tr>
<td></td>
<td>Project Kickoff Meeting Agenda and Minutes</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Stakeholder Management Strategy</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Team Contract for managing team communications</td>
<td></td>
</tr>
<tr>
<td>Planning</td>
<td>Project Management Plan. Includes: Project scope statement, project requirements, work breakdown structure, project schedule, project cost estimates, communications plan, risk management plan and risk register.</td>
<td>Gather requirements and develop estimates for completing the project work.</td>
</tr>
<tr>
<td>Executing</td>
<td>Bi-weekly status reports and/or notes from team meetings and meeting with project stakeholders.</td>
<td>Execute the project, communicate with stakeholders, report project status</td>
</tr>
<tr>
<td></td>
<td>Project-specific outputs produced while executing the project</td>
<td></td>
</tr>
<tr>
<td>Monitoring</td>
<td>Completed project schedule with schedule variances</td>
<td>Monitor the progress of the project against the estimates and project requirements developed during planning.</td>
</tr>
<tr>
<td>and Control</td>
<td>Cost Tracking Worksheet</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Updated Risk Register</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scope Change Request (if applicable)</td>
<td></td>
</tr>
<tr>
<td>Closing</td>
<td>Customer Acceptance / Project Completion Form</td>
<td>Obtain project sponsor approval and feedback, report the final results of the project and reflect on the project experience.</td>
</tr>
<tr>
<td></td>
<td>Final Project Report</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Project Lessons Learned</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Project Presentation</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Project Deliverables and Tasks
Reflection

As part of SL, reflection activities, such as clarifying expectations to help students activate prior relevant knowledge, provide additional keys to success (Weis, 2000; Helle et al. 2006). In IPM, students completed several types of reflection activities. At regularly scheduled dates throughout the semester, the normal class lecture and activities were replaced by a formal status meeting. Prior to the meeting, each project manager submitted a status report that described the work completed to date, project issues, what worked well and suggestions for other projects, and project management concepts that applied to the phenomena experienced during that reporting period. During the meeting, project managers reported this status to the instructor and other students. Following the completion of status reporting, general class discussion focused on lessons learned and other knowledge exchange between the students.

At project closure, student groups also reflected on the entire project experience. As part of the final project presentation and the final report, groups reported their lessons learned on the project, what they would do differently, project management tools and techniques that were critical to their success, and whether team members wished to pursue project management as a future career. The third and final reflection activity was the service project feedback survey, which served as the data gathering method for this research.

METHOD

Identifying Variables

Research participants were 96 students enrolled in the IPM course (66 undergraduate, 30 graduate). All undergraduate students were Junior or Senior academic level.

We provided an online survey to students to determine (1) student attitudes toward completing service projects for learning project management, (2) the impact of the IPM SL project on students, and (3) the likelihood these students would continue to use project management skills learned from the service project in the future. Survey items were based on SL literature and adapted from existing instruments used in prior research. Table 3 offers a summary of survey items and the sources for each scale item:

- Rose et al. (2005) - items asking students what they learned about information systems design, the applicability of classroom material to their service project, of course skills to their future careers, and opportunities to interact with clients and learn about the client organizations.
- Toncar et al. (2006) - Service Learning Benefit scale (SELEB). This instrument investigated student perceptions of their SL experiences, the development of personal skills, citizenship values, and personal growth resulting from their SL course.
- Parker et al. (2009) - questions exploring the link between SL and students’ insights into community service agencies, classroom skills deployed in support of service, personal development and confidence, and expected level of volunteerism in the future.
- Fu & Cecil (2011) - questions regarding the relationship between the SL project and students’ increased understanding of course principles, analytical and problem-solving skills, and their ability to deploy classroom materials to project work.
- Mendoza (2013) – items about students’ experience with the course, ways to improve the course, how the course benefitted them personally, tools and techniques of particular relevance, and business and community agency insights obtained by students.

We made the surveys available to students on the last day of the course, following completion of all required course activities. Students completed the surveys on a voluntary basis. We collected a total of 85 post-project surveys, of which 58 were undergraduate students and 27 were graduate students. This number represents 91% of the total population of 96 students who participated in the course with the same instructor for the academic year 2012-2013 (Fall 2012 and Spring 2013 semesters).
<table>
<thead>
<tr>
<th>Latent Construct</th>
<th>Survey Item <em>(Question Type)</em></th>
<th>Item Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service-Learning Attitudes</td>
<td>Recall what you did for the project. Do you think you can apply the knowledge and tools that you learned from doing the project when dealing with <strong>personal or professional projects</strong> in the future? If yes, what would be the <strong>project management</strong> concepts or tools that you think would be used in the future? <em>(Open-ended)</em></td>
<td>Fu &amp; Cecil (2011) Parker et al. (2009) Rose et al. (2005) Toncar et al. (2006)</td>
</tr>
<tr>
<td></td>
<td>How has completing a ‘service-learning’ project benefitted you? Please explain. <em>(Open-ended)</em></td>
<td>Mendoza (2013) Parker et al. (2009)</td>
</tr>
<tr>
<td></td>
<td>Should future students work on similar ‘service-learning’ projects? Please explain. <em>(Open-ended)</em></td>
<td>Mendoza (2013)</td>
</tr>
<tr>
<td></td>
<td>If you had the ability to change the ‘service-learning’ project for future students, what would be the first thing that you would change? Please explain. <em>(Open-ended)</em></td>
<td>Mendoza (2013)</td>
</tr>
<tr>
<td></td>
<td>I would like to continue my volunteering as part of my future development as a result of my community service experience. <em>(5-point Likert scale, strongly disagree/strongly agree)</em></td>
<td>Parker et al. (2009) Toncar et al. (2006)</td>
</tr>
<tr>
<td></td>
<td>I learned more about myself working in a group. <em>(5-point Likert scale, strongly disagree/strongly agree)</em></td>
<td>Fu &amp; Cecil (2011) Parker et al. (2009) Toncar et al. (2006)</td>
</tr>
<tr>
<td>Application and Adoption of Project Management Skills</td>
<td>I believe this project gives me ______ working on future projects. <em>(5-point Likert scale, no value/great value)</em></td>
<td>Parker et al. (2009) Rose et al. (2005) Toncar et al. (2006)</td>
</tr>
<tr>
<td></td>
<td>I will _____ follow the project management processes (initiating, planning, executing, monitoring &amp; controlling, closing) for working on future projects. <em>(5-point Likert scale, not/definitely)</em></td>
<td>Fu &amp; Cecil (2011) Rose et al. (2005)</td>
</tr>
<tr>
<td></td>
<td>I will _____ use project management techniques (stakeholder analysis, team contract, Gantt chart, personality test, etc.) working on future projects. <em>(5-point Likert scale, not/definitely)</em></td>
<td>Fu &amp; Cecil (2011) Mendoza (2013) Rose et al. (2005)</td>
</tr>
<tr>
<td></td>
<td>I will _____ use project management related software (Microsoft Project, websites, Microsoft Excel etc.) working on future projects. <em>(5-point Likert scale, not/definitely)</em></td>
<td>Fu &amp; Cecil (2011) Mendoza (2013) Rose et al. (2005)</td>
</tr>
</tbody>
</table>

Table 3. Survey items
Analysis and Next Steps – Measuring Effectiveness

We derived three major constructs from our research questions: student attitudes toward the SL project, impact of the SL project, and student intentions to adopt and apply project management skills to future projects (Table 3). Content analysis of the student responses with emergent coding will enable us to identify common themes among the responses and, as the data permits, establish categories of student attitudes toward SL. Following factor analysis to confirm the survey items with our intended construct, we intend to test correlations with the other constructs in preparation for development of a model based on the confirmation of these factors.

CONCLUSION

This paper described our current efforts to use a theory of change approach for developing a model of SL for project management education. We applied the theory of change process in describe our desired learning outcomes for teaching IPM, and preconditions for using SL pedagogy as an intervention for achieving the learning outcomes. We anticipate the results of our analysis will help us to confirm a set of variables and a preliminary model to be used for measuring the effectiveness of SL pedagogy for achieving our desired learning outcomes.

REFERENCES


Outlay and Mendoza

Project Management Education through Service Learning


