The Influence of Analyst Communication in IS Projects

Shadi Shuraida
City University of New York
shadi.shuraida@baruch.cuny.edu

Henri Barki
HEC Montréal
henri.barki@hec.ca

Abstract

Information system (IS) researchers have long noted that IS analysts need to understand users' needs if they are to design better systems and improve project outcomes. While researchers agree that analyst communication activities are an important prerequisite for such an understanding, little is known about the nature of different communication behaviors IS analysts can undertake to learn about users' system needs and the impact of such behaviors on IS projects. To address this gap, this paper draws from the learning literature to articulate the information transmission activities IS analysts can undertake and the content of the information they can transmit when learning about users' organizational tasks and information needs. The influence of analyst communication activities on the generation of valid information regarding user needs, analyst learning, and IS project outcomes are then investigated via a case study of two IS projects. The analysis of the two cases suggests that analysts who encourage the use of concrete examples, testing, and validation, and who solicit feedback about users' business processes are likely to better understand users' tasks, and in turn design systems that better meet users' task needs than analysts who do not.

Keywords: IS Development, IS Implementation, IS Project, User-Analyst Communication, IS Analyst Learning, Case Study.

---

* Bernard Tan was the accepting senior editor. This article was submitted on 15th May 2011 and went through two revisions.

1. The Influence of Analyst Communication in IS Projects

Developing and implementing information systems (IS) in organizations continues to be a challenging task with nearly two thirds of projects facing setbacks or failures (Nelson, 2007; Standish Group, 2009). As inaccurate or incomplete information requirements are believed to negatively impact many information system development (ISD) projects (Bostrom, 1989; Browne & Rogich, 2001; Byrd, Cossick, & Zmud, 1992; Marakas & Elam, 1998; Pitts & Browne, 2007), it is generally agreed that IS analysts[1] need to acquire adequate and valid information about organizational and user needs if they are to develop high-quality systems (Browne & Ramesh, 2002; Byrd et al., 1992; Marakas & Elam, 1998; Mathiassen, Saarinen, Tuunanen, & Rossi, 2007).

Analysts often need to communicate with users to acquire such information (Curtis, Krasner, & Iscoe, 1988; Keil & Carmel, 1995; Neill & Laplante, 2003; Watson & Frolick, 1993) and researchers have studied user-analyst communication and its role in ISD projects (e.g., Hartwick & Barki, 2001; Joshi, Sarker, & Sarker, 2007; Ko, Kirsch, & King, 2005; Lind & Zmud, 1991). A basic premise of much of this work is that analysts who obtain accurate information requirements will develop high-quality systems. However, analysts who are provided with accurate user requirements may still fail to clearly understand them due to ineffective communication that can stem from various factors, such as user-analyst differences in mental models (Boland & Tenkasi, 1995; Bostrom, 1989; Orlikowski & Gash, 1994) or languages (Appan & Browne, 2001; Browne & Ramesh, 2002; Byrd et al., 1992), and, as a result, they may deliver systems that do not meet users' needs (Edstrom, 1977; Gallivan & Keil, 2003; Newman & Noble, 1990).

Hence, some researchers have examined different communication approaches analysts can undertake to learn about users' information needs (e.g., Boland, 1978; Majchrzak, Beath, & Lim, 2005; Salaway, 1987; Walz, Elam, & Curtis, 1993). As ISD is a knowledge-intensive activity in which analysts need to acquire information and translate it into system design characteristics (Churchman & Schainblatt, 1965; Markus & Mao, 2004), investigating analyst communication activities via a learning perspective can be a fruitful approach and can help shed light on the specific behaviors that result in effective communication and the acquisition of accurate and valid knowledge about user requirements. However, past research on analyst communication has essentially focused on relatively broad conceptualizations of communication approaches and strategies such as dialectic communication exchange (Boland, 1978) and cognitive elaboration (Majchrzak et al., 2005), while leaving the information content of analysts' communication exchanges largely unexplored. One exception is Salaway (1987), who found that analysts who practiced learning behaviors as specified in Argyris and Schon's (1974, 1978) organizational learning approach generated more valid information than those who used a traditional interaction approach (the content of these communication behaviors and their impact on project outcomes were not examined).

In sum, IS analysts' communication activities form a key aspect of their knowledge acquisition about users' requirements, which can significantly impact IS project outcomes (Churchman & Schainblatt 1965; Byrd et al., 1992; Markus & Mao, 2004). However, we currently lack both the theory and empirical evidence needed to better understand the relationship between analysts' communication behaviors and IS project outcomes. In an effort to address this gap, we lay the groundwork of a theoretical framework and provide preliminary empirical evidence regarding this relationship. To do so, in Section 2, the paper first synthesizes past empirical research on analyst communication and requirements determination based on Berlo's (1960) communication framework. In Section 3, it draws from organizational learning theories (Argyris & Schon, 1974; 1978; Salaway, 1987) to empirically investigate how analysts' information transmission activities and the content of the information they transmit influence their knowledge of users' information needs and IS project outcomes. A case study approach—described in Section 4—was adopted to examine two IS projects. The analysis (Section 5) and findings (Section 6) of the two cases indicate that analysts who encourage the use of concrete

---

[1] The term IS analyst refers to individuals responsible for defining user and organizational information needs in order to build information systems (Vitalari & Dickson, 1983). IS analysts may also perform other tasks during IS projects such as project management, business process reengineering, and programming.

examples, testing, and validation, and who solicit feedback about users' business processes are likely to better understand users' tasks, and, in turn, design systems that better meet users' task needs than analysts who do not. Moreover, based on evidence from the two cases, propositions and research questions are also suggested for future research in Section 6.

2. Requirements Elicitation and Analyst Communication

According to some researchers, the literature on user requirements elicitation is akin to a confusing "methodology jungle" where little agreement exists on the relative importance of the activities advocated by different methods and how these activities need to be performed (Beath & Orlikowski, 1994; Keil & Carmel, 1995; Marakas & Elam, 1998; Mathiassen et al., 2007; Robertson & Robertson, 2006). Additionally, many ISD approaches have focused on the identification and elicitation of information requirements without being explicit about what specific actions IS analysts need to execute in order to effectively understand the information needs of the organization and its people (Galliers & Swan, 1997; Patnayakuni, Ruppel, & Rai, 2005; Robertson & Robertson, 2006; Stein & Vandenbosch, 1996).

Partial answers that address this issue can be found in user-analyst communication research (Boland, 1978; Bostrom, 1989; Gallivan & Keil, 2003; Hartwick & Barki, 2001; Kirsch & Beath, 1996; Lind & Zmud, 1991; Marakas & Elam, 1998; Markus & Mao, 2004; Newman & Noble, 1990) where it is suggested that "certain forms of communication are more […] valuable than others" (Gallivan & Keil, 2003, p. 41; Boland, 1978; Kraut & Streeter, 1995; Lind & Zmud, 1990; Marakas & Elam, 1998; Majchrzak et al., 2005; Newman & Noble, 1990; Salaway, 1987). For example, past research on this topic found that the use of dialectic dialogue (Boland, 1978; Salaway, 1987) and specific communication procedures and methods (Bostrom, 1989; Majchrzak et al., 2005) positively influenced ISD project outcomes.

To synthesize studies related to IS analysts' communication activities, we reviewed past research on user-analyst communication and requirements determination (RD). The RD stream was included because analysts have been found to elicit system requirements mainly by communicating with system users (Curtis et al., 1988; Keil & Carmel, 1995; Neil & Laplante, 2003; Watson & Frolick, 1993). Appendix A provides a detailed description of the process, criteria, and results of this review.

Our review yielded a sample of 49 empirical studies in 90 articles that have investigated IS analysts' communication activities. The latter were categorized in Table 1 along Berlo's (1960) central communication components: the communication message, which specifies the code or language used, the content of the information communicated, and the form of the treatment or transmission; the communication channel, which refers to the medium of communication used; and the communication source/receiver, which refers to the characteristics of the communicating individuals. The present study focuses on the "communication message" component; that is, the communication "behavior available in physical form" (Berlo, 1960, p. 30), which reflects analysts' transmission of information, the language they use, and the content of the information transmitted.

Table 1. A Synthesis of Empirical Research on User-Analyst Communication				
Communication perspective		**Study**	**Focus**	**Gaps**
Communication message	Code (or language)	Agarwal et al. (1999), Gemino and Parker (2009), Moynihan (1999)	Compared the effectiveness of modeling techniques on users' information comprehension.	Did not provide guidelines regarding specific analyst communication activities. No links to IS project outcomes.
	Transmission/ treatment (communication techniques)	Darke and Shanks (1997), Kaiya et al. (2005), Lichter et al. (1994), Maiden and Hare (1998), Potts et al. (1994), Saiedian, Kumarakulasingam, and Anan (2005), Shin, Sutcliffe, and Gregoriades (2005)	Evaluated the effectiveness of different communication tools and techniques (e.g. prototypes, card sorting, use cases).	Did not provide guidelines regarding specific analyst communication activities. Did not identify information content to be acquired.

Table 1. A Synthesis of Empirical Research on User-Analyst Communication (cont.)

Communication perspective		Study	Focus	Gaps
Communication message	Transmission/ treatment (interview techniques)	Agarwal and Tanniru (1990), Browne and Rogich (2001), Duggan (2003), Duggan and Thachenkary (2003, 2004), Laporti, Borges, and Brahanholo (2009), Leifera, Leeb, and Durgeea (1994), Marakas and Elam (1998); Moody et al. (1998); Pitts and Browne (2007)	Compared the effectiveness of different elicitation (e.g. structured interviews, JAD, Focus groups) techniques.	Focused on interviews, neglecting other communication activities analysts undertake. No links to IS project outcomes.
	Transmission/ treatment (interaction methods)	Boland (1978)	Compared the effectiveness of different interaction procedures (traditional rationality vs. alternate mutual learning protocols).	Did not provide guidelines regarding specific analyst communication activities. Did not identify information content to be acquired.
		Salaway (1987)	Proposed the use of learning approach during analysts' interactions with users.	Did not identify information content transmitted. No links to IS project outcomes.
		Majrchzak et al. (2005)	Proposed the use of 'Cognitive Elaboration' strategies during ISD projects.	Provided broad guidelines such as "use multiple ways to describe an idea."
	Information content	Nichols (1981)	Proposed an interview procedure to acquire 3 pieces of information IS analysts need to obtain.	Provided broad guidelines, such as acquire information about the specific task activities. No links to analyst learning or ISD project outcomes.
	Communication channel	Kraut and Streeter (1995)	The use of formal/informal group communication modes.	Did not provide guidelines regarding specific analyst communication activities.
		Gallivan and Keil (2003), Haumer, Pohl, and Weidenhaupt (1998), Lind and Zmud (1991), Shirani (2006)	Richness of user-analyst communication channels used.	Did not provide guidelines regarding specific analyst communication activities.
		Andreou (2003), Keil and Carmel (1995), Watson and Frolick (1993)	Communication channels and methods used by IS analysts to communicate with users.	Did not provide guidelines regarding specific analyst communication activities.
		Butler and Fitzgerald (2001), Hartwick and Barki (2001), Joshi et al. (2007), Kaiser and Srinivasan (1982), Lind and Zmud (1991), McKeen and Guimaraes (1997), , Sawyer and Guinan (1998), Stobart, Thompson, and Smith (1991), Szajna and Scamell (1993)	Communication frequency or occurrence during ISD projects.	Did not provide guidelines regarding specific analyst communication activities.
	Communication source / receiver	Guimaraes, Staples, and McKeen (2003), Ko et al. (2005), McKeen, Guimaraes, and Wetherbe (1994), Saleem , Jones, and Moses (2006)	The effect of analysts' communication competence on shared understanding.	Did not provide guidelines regarding specific analyst communication activities.
	Communication context	Arthur and Groner (2005), Bostrom (1989)	Proposed the use of communication frameworks (e.g., precision model) during an ISD project.	Focused on communication environment/context and meeting format. Did not identify the information content acquired or its importance to IS project outcomes.

Studies along the communication message stream have typically focused on requirements determination and the prescription of interview techniques and tools for eliciting information from users. For example, structured interviews (e.g. Agarwal & Tanniru, 1990; Duggan & Thachenkary, 2003, 2004), certain question types (e.g. Marakas & Elam, 1988; Moody, Blanton, & Cheney, 1998; Browne & Rogich, 2001; Pitts & Browne, 2007), and communication enhancing tools and techniques (e.g., Lichter, Schneider-Hufschmidt, & Zullighoven, 1994; Kaiya, Shinbara, Kawano, & Motoshi, 2005; Maiden & Hare, 1998; Potts, Takahashi, & Antón, 1994; Darke & Shanks, 1997) have been noted as leading to more effective and efficient requirements elicitation. Other studies within this stream found that dialectic communication patterns led to the generation of valid information (Salaway, 1987), and improved user learning and system design outcomes (Boland, 1978; Majchrzak et al., 2005), providing general guidelines to analysts regarding the use of different dialectic communication patterns, such as mutual learning interaction protocols (Boland, 1978), cognitive elaboration strategies (Majchrzak et al., 2005), and organizational learning approaches (Salaway, 1987).

The second stream has focused on assessing the extent (e.g., Butler & Fitzgerald, 2001; Hartwick & Barki, 2001; Joshi et al., 2007; Lind & Zmud, 1991; McKeen et al., 2007) and the impact of different user-analyst communication channels (Kraut & Streeter, 1995; Keil & Carmel, 1995; Watson & Frolick, 1993) on ISD projects. For example, research has found that more frequent communication (Hartwick & Barki, 2001; Joshi et al., 2007; Lind & Zmud, 1991; Sawyer & Guinan, 1998), the use of richer (Gallivan & Keil, 2003; Lind & Zmud, 1991), less formal (Kraut & Streeter, 1995) and direct communication channels (Keil & Carmel, 1995) positively influenced ISD project outcomes.

Finally, studies in the third and fourth streams have investigated contextual factors such as analysts' communication competence and skill (e.g., Ko et al., 2005; Guimaraes et al., 2003; McKeen et al., 1994; Saleem et al., 2006) or general models that facilitate communication (e.g., Arthur & Groner, 2005; Bostrom, 1989).

Thus, while past research on user-analyst communication and RD has provided valuable insights regarding this phenomenon, with the exception of Nichols (1981), it has left unexplored the information content of the messages being communicated. A more in-depth examination of this content can help us better understand what specific activities analysts can undertake to effectively elicit and learn users' information requirements. While some studies have focused on effective information elicitation techniques (e.g., Agarwal & Tanniru, 1990; Marakas & Elam, 1988; Moody et al., 1998; Browne & Rogich, 2001; Pitts & Browne, 2007), only a few have examined information transmission by analysts and how it can help them acquire the knowledge and understanding needed for designing better systems (e.g., Majchrzak et al., 2005; Salaway, 1987). Hence, despite the importance of analysts' information transmission activities and the information content of this communication in ISD projects, theory-based empirical studies that investigate this impact on ISD projects are still lacking. Such research can help provide analysts with specific communication behavior guidelines that can improve their understanding of user and organizational system needs, which in turn are likely to lead to better system design decisions. With this objective, the present study draws on learning theory (Argyris & Schon, 1974, 1978) to examine how analysts' information transmission activities and its content influences their learning and the project.

3. Analyst Learning in IS Projects

Research strongly supports the presence of a significant relationship between user-analyst communication and system success (Hartwick & Barki, 2001; McKeen & Guimaraes, 1997; Lind & Zmud, 1991). A key premise of this research is that, to design systems that support business needs, analysts acquire knowledge about user and organizational needs by communicating with users (Byrd et al., 1992; Chakraborty, Sarker, & Sarker, 2010; Joshi et al., 2007; Markus & Mao, 2004). Hence, consistent with the organizational literature where communication is viewed as a learning activity that leads to knowledge (e.g. Argyris & Schon, 1974; Edmondson, 1999; Huber, 1991; Weick & Ashford, 2001), studies have suggested that analysts can acquire this knowledge via effective communication with users (Boland, 1978; Curtis et al., 1988; Gallivan & Keil, 2003; Walz et al., 1993).

A well-known learning model is that of Argyris and Schon (1974, 1978). This widely adopted model (Arthur & Aiman-Smith, 2001) has been applied to various organizational contexts. It describes learning as the detection and correction of errors, where errors are "any feature of knowledge or of knowing that makes action ineffective" (Argyris, 1976, p. 365). As Table 2 shows, according to this model, learning takes one of two forms (Figure 1) as individuals try to correct these "errors". The first is single-loop learning where interpersonal communication is limited to the methods of action, while the second is double-loop learning where the governing variables (i.e., the underlying assumptions, norms, and values) are examined. According to Argyris and Schon, double-loop learning is more effective than single-loop learning because, when individuals question their objectives and activities, generation of valid information and receptivity to feedback is facilitated, whereas in single-loop learning information search is limited to task activities.

Table 2. Examples of Organizational Research (Based on Argyris and Schon, 1974, 1978)

| Level of analysis	Study	Argyris and Schon concept used	Topic studied
Individual	Salaway (1987)	Model 1 /Model 2 learning	The effect of learning model used on the generation of valid information
	Arthur and Aiman-Smith (2001)	Single/double-loop learning	The proportion of second order employee suggestions over time
	Tucker, Edmondson, and Spear (2002)	Single/double-loop learning	Identify problem-solving behaviors
Group	Edmondson (1999)	Definition of learning	Identify learning behaviors' influence on group performance
	Stein and Vandenbosch (1996)	Single/double-loop learning	Identify type of learning and its antecedents in ISD projects
Organization	Gill (1995)	Definition of learning	Influence of IT on organizational learning
	Sinkula, Baker, and Noordewier (1997)	Higher (double-loop)/lower order (single-loop) learning	The effect of organizational learning orientation on information processing behaviors
	Baker and Sinkula (1999)	Higher/lower order learning	The effect of learning orientation on organizational performance
	Baker and Sinkula (2002)	Higher/lower order learning	The effect of learning orientation on product innovation
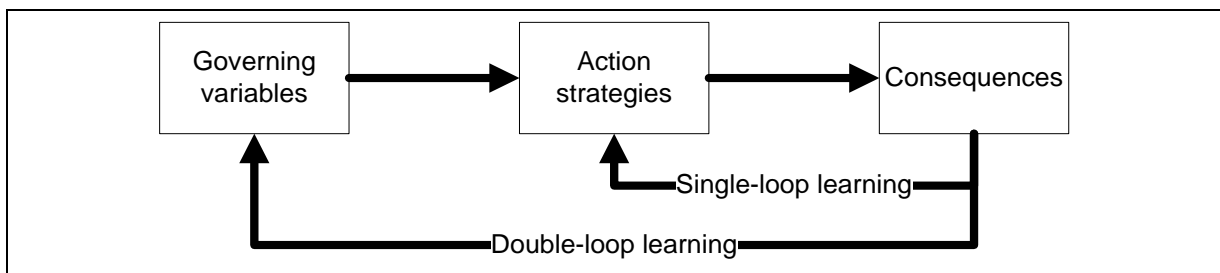
Figure 1. The Elements of the Learning Process

Thus, Argyris and Schon (1974) proposed two models that characterize individuals' governing variables, interpersonal actions, and intended consequences that either inhibit (i.e., Model 1) or enhance (i.e., Model 2) double-loop learning. For example, an underlying strategy of Model 1 is to control others by unilaterally defining task objectives, distorting information, or withholding it. In

contrast, the principal strategy of Model 2 is to generate valid information by encouraging open communication and the public testing of ideas (Argyris, Putnam, & McLain Smith, 1985; Argyris & Schon, 1974, 1978; Salaway, 1987)[2]. Based on Argyris and Schon's (1974, 1978) learning models, Salaway (1987) identified four transmission activities (Table 3): advocacy (the act of supporting or defending a position, idea, or action), inquiry (the act of eliciting knowledge), confrontability (the manner in which individuals challenge others' ideas), and discussability (discussing relevant thoughts regarding a problem). In sum, while Model 2 communication activities promote the testing of concrete examples and solicit feedback when seeking information or expressing one's ideas (Argyris, 2002; Argyris & Schon, 1996; Bolman & Deal, 2008), individuals who engage in Model 1 communication activities do not illustrate their ideas or encourage feedback about them, and do not encourage testing of the claims being made (Argyris, 2002).

Argyris and Schon's (1974) learning theory is based on the premise that individuals design their interactions in order to make decisions that will achieve their intended consequences. As such, the model can be useful in ISD contexts where analysts need to undertake communication activities to acquire the knowledge they need for delivering systems that meet user and organizational needs (Byrd et al., 1992; Curtis et al., 1988; Chakraborty et al., 2010; Gallivan & Keil, 2003; Joshi et al., 2007; Walz et al., 1993). Hence, the present paper adopts Argyris and Schon's (1974, 1978) definition of learning to articulate IS analysts' communication activities as a key influence in the acquisition of valid requirements information, increased knowledge, and better systems and project performance.

Table 3. Characteristics of Model 1 and Model 2 Communication Transmission Activities (Argyris, 2002; Argyris & Schon 1974, 1978; Salaway, 1987)			
Transmission behavior	**Definition of transmission behavior**	**Model 1 approach**	**Model 2 approach**
Advocacy	To support or defend a position, idea, or action	- Without directly observable data (i.e. abstract) - Without feedback solicitation - Without testing	- With directly observable data (grounded in data) - With feedback solicitation - With testing
Inquiry	To seek knowledge, examine or investigate	- Abstract / Inferential - No testing	- Observable (grounded in data) - Testing
Confrontability	The manner of presentation used in challenging actions, ideas, or feelings	- Confront others about their actions - Make demands of others	- Make self confrontable - Express preferences giving others choices
Discussability	Identifying and discussing thoughts that are relevant to solving a problem	- Relevant thoughts held private	- Relevant thoughts discussed publicly

It is interesting to note that, while Model 1 communication activities may be more efficient for addressing routine problems, Model 2 activities are likely to result in the production of the "most complete valid information" and knowledge necessary for taking effective action (Argyris, 1976, p. 369). As individuals seek valid information and feedback (i.e., Model 2 communication activities), they can more effectively address technical or interpersonal issues. For example, Model 2 activities were found to generate more valid information than Model 1 activities (Salaway, 1987), improve the performance of organizational teams (Edmondson, 1999), and prevent the recurrence of problems

---

[2] The concepts of single and double-loop learning can be viewed as having parallels with some elements of Habermas' (1984, 1987) strategic and communicative action. The authors would like to thank an anonymous reviewer for pointing this out

(Tucker et al., 2002). Model 2 communication activities may be especially salient for complex and ill-structured systems where the link between actions and outcomes is delayed or ambiguous (Bolman & Deal, 2008), such as in IS projects where analysts need to effectively communicate with users to gather valid information and acquire knowledge (Argyris & Schon, 1978; Browne & Rogich, 2001). For example, analysts commonly gather abstract system requirements by interviewing or surveying users (Neill & Laplante, 2003; Watson & Frolick, 1993) without necessarily seeking observable data to substantiate the validity of the collected requirements, which reflect Model 1 communication behaviors. While this approach may result in systems that meet all requirements identified by the analysts, it might still not meet users' needs (Arthur & Gröner, 2005; Gallivan & Keil, 2003) due to analysts' incomplete identification or misunderstanding of user requirements (Appan & Browne, 2010; Browne & Ramesh, 2002; Byrd et al., 1992). In contrast, by seeking and presenting observable information and publicly testing it, Model 2 communication activities are likely to yield valid knowledge, which, in turn, can lead to systems that better meet user needs.

Initial support for the above argument in ISD contexts was provided by Salaway (1987) who examined tape records of analysts' interactions with users, and found that analysts who used Model 2 transmission activities detected more errors in the information provided to them than those who used Model 1 activities. However, Salaway (1987) did not examine the content of the transmitted information and did not relate Model 2 transmission activities to project outcomes. Further, as also noted in Salaway (1987), five-minute tape segments of user-analyst interactions were "not adequate for determining larger errors" (p. 261) in ISD project contexts. The present paper extends this research by examining, via case studies of two ISD projects, whether IS analysts who use Model 2 transmission activities acquire more knowledge about users' information system needs than analysts who use Model 1 activities, and whether this knowledge influences project outcomes.

4. Method

The present study adopted a case study approach (Lee, 1989; Yin, 2003) to examine IS analysts' communication activities and their impact on IS project outcomes, an approach that is especially appropriate for studying complex social processes in their real-world contexts (Dubé & Paré, 2003; Eisenhardt, 1989; Eisenhardt & Graebner, 2007; Yin, 2003). Given the complexity of user-analyst communications and their embeddedness in organizational activities, a case study approach is particularly useful for examining the present study's research questions.

4.1. Case Selection

A key criterion in selecting the cases to be studied was to allow a comparison and contrast of the core variables of interest (Eisenhardt, 1989; Yin, 2003) (i.e., the characteristics of analysts' communication activities and their outcomes). The cases were also selected so as to minimize variation in project outcomes due to factors outside the present study's focus (Eisenhardt, 1989), such as analyst expertise (Marakas & Elam, 1998; Pitts & Browne, 2004; Vitalari, 1985), communication channels used (Gallivan & Keil, 2003), and project risk (Barki, Rivard, & Talbot, 1993). Further, projects with prepackaged software were also excluded as their outcomes could depend on factors other than analysts' communication activities, such as decisions made prior to the start of the project (Soh, Kien, & Tay-Yap, 2000; Wu & Shen, 2006). Hence, to minimize the potential impact of factors other than communication activities, the cases were selected from relatively large, recently rolled out in-house developed systems. It was also thought that large, organizational, in-house projects would be likely to involve extensive user-analyst interactions (Argyris, 1976; Salaway, 1987), enabling a clearer focus on the study's key constructs.

In order to identify projects that met the above criteria, several IT managers and executives were contacted and asked about recent IS development and implementation efforts in their organizations, resulting in the identification of two projects (Table 4): one at LifeSci, a subsidiary of a pharmaceutical industry leader, and the other at Regional Insurer, also a regional subsidiary of a leading insurance firm. Each organization had around 2000 employees, and viewed the projects as a priority for replacing existing systems via the use of an Agile methodology, which helped to further control the influence of extraneous factors on project outcomes.

4.2. Data Collection

The main data collection method was face-to-face semi-structured interviews with key informants including the analysts responsible for system development and implementation, key users, the IT and business department managers, and other respondents identified by various stakeholders as being knowledgeable about the project (Table 4). Initially, IT and project managers identified the analysts responsible for system development and implementation, in addition to the key users of the system and their representatives. Subsequently, other stakeholders were identified via a snowball sampling strategy by the managers, the key users, and the analysts either during the interviews, or suggested by them when asked about other potential respondents who participated in the project. The interview protocol is provided in Appendix B, and varied slightly according to the informant's role in the project. Follow-up telephone interviews helped clarify or supplement the information from the interviews, which lasted between 30 minutes and two hours each. All but three interviews were tape recorded and case notes used for the three unrecorded interviews. The recorded interviews were fully transcribed before analysis and project documentation (e.g., emails, meeting minutes, business and data flow charts, system print screens, and system output reports) was used to triangulate the interview data (Dubé & Paré, 2003; Eisenhardt, 1989; Yin, 2003).

Table 4. Summary Description of the Two Cases		
	LifeSci	**Regional Insurer**
The organization	A national subsidiary of a large multinational pharmaceutical company, with approximately 2000 employees.	A regional subsidiary of a global and leading insurance firm, with approximately 2000 employees.
The project	A 6 month project to develop an application for quality control analysis and data upload.	An 18 month project to replace a DOS account billing system with a web application.
People interviewed (# of interviews)	- IS manager/PM (2) - Lead IS analyst (1) - IT programmer (2) - Five users /technicians (1 interview with each user)	- IS manager / PM (1) - Lead IS analyst (1) - Users' manager / Product owner (1) - User (1)
Analyst characteristics	- 10 years development experience - 10 years in quality control laboratory systems - 2 months in current organization	- 10 years development experience - No experience in financial systems - No experience in current organization
Project outcomes	- Schedule adherence: Project late - Task-system fit: Varied - System quality: High	- Schedule adherence: Project on-time - Task-system fit: High - System quality: High

5. Analysis

The cases were analyzed via an analytic induction approach that began by examining the relationships between the study's key constructs, followed by a reexamination of the data to uncover patterns that might lead to emergent findings or revisions to the study constructs or propositions (Patton, 2002, p. 454). Accordingly, we started with an in-depth analysis of each case to determine whether our observations supported the theorized relationships among the study's constructs (Lee, 1989; Patton, 2002) and to facilitate cross-case analysis. With this objective, the transcripts and case documents were reviewed and the data initially coded according to specified constructs (Eisenhardt, 1989; Yin, 2003) and organized according to the four modes of the Chakraborty et al. (2010) framework, which characterizes group interactions in projects: scoping, sense-making, dissension, and termination. Next, the specified constructs were placed in two tables, analyst communication messages and project outcomes, and a finer-grained coding process was undertaken for each construct. For example, quotes associated with communication messages were expanded and

refined to identify the specific transmission activities that had been used and their characteristics, their information content, and the knowledge acquired by the analyst; project outcomes were further categorized into outcomes related to either the ISD process or product. While process outcomes pertain to project quality, including adherence to project schedules and budgets, product outcomes reflect characteristics of the delivered system, as well as stakeholder attitudes and beliefs about the project, the system, and its use (Barki et al., 1993). To validate the collected data and their interpretation, the written-up cases were read by the IT managers of both organizations for verification and confirmation that the events recounted had occurred as described in the write-ups. Subsequently, the case findings were compared and contrasted in a cross-case analysis to better understand the different processes of each project, and to examine the theorized relationships, and if necessary, to revise the study constructs and propositions according to the case data.

5.1. Case 1 Overview: LifeSci[3]

LifeSci is a nationally operating division of a multinational pharmaceutical company. In its quality control (QC) laboratory, LifeSci technicians perform various quality control tests on production batches and new drugs that are in the process of development. To analyze and store their test data, the QC technicians use an elaborate spreadsheet application developed some years ago by one of their senior members. As they perform testing operations, the technicians go through consecutive sheets of the intricate spreadsheet to enter their test data, which flows through the various sheets and yields several reports and graphs.

Realizing the danger of storing sensitive test information in spreadsheets, a senior technician approached the IT Project Manager (ITPM) to ask if the IT department could provide a "corporate" database solution for uploading the spreadsheet information. The ITPM had extensive experience in providing systems to the laboratory, and, after several meetings with the senior technician, he agreed to develop a corporate database solution for the QC laboratory by offering to replace the current spreadsheet application with a system that could also provide further support to the QC technicians' data analysis tasks.

Next, the ITPM assigned an experienced LifeSci analyst to lead the development and implementation of this application. The analyst was recently recruited at LifeSci but had over ten years of systems analysis experience in the pharmaceutical industry. After seven months of development, the new system was piloted with one technician who found it unsuitable for his tasks. To address the technician's concerns, the IT group added more functionality and three months later launched a second release of the system. While the second release was well received and appreciated by many technicians, the system was still not fully used as it did not completely address the needs of all the technicians.

5.1.1. Communications of the LifeSci Analyst
As can be seen in Tables 5 and 6, the LifeSci analyst acquired information from various sources including the users (lab technicians) and his IT colleagues. The content of this information was about users' business processes and tasks, features required from the new IS, existing systems from which data could be extracted for input into the new IS, and relevant systems technicians currently used. Based on Byrd et al. (1992), these were categorized (in the last column of the tables) as information on 1) user requirements and interface design, 2) business process and task, and 3) problem frame (i.e., ISD goals and the existing support environment).

5.1.2. The Scoping Phase
After being assigned to the project, the LifeSci analyst's learning began by communicating with the ITPM who gave him an overview of the project's goals. The ITPM noted that, during the first two months, he spent an average of 8-10 hours/week with the analyst and 4-5 hours/week over the next two months, explaining the functionalities of relevant LifeSci systems and the desired future state of

---

[3] The name of the organization, the industry, the implemented system, and the context of development were disguised in the first case (LifeSci) to ensure confidentiality and anonymity of the organization and its participants. However, the events that transpired in the case, the communication activities of the IS analyst, and all project outcomes were unchanged. For the second case (Regional Insurer), only the name of the organization was disguised.

the new system. The ITPM also viewed the project as "a learning experience for [the newly hired analyst] to become familiar with the systems at [LifeSci]".

During this period, the analyst also met with his IT colleagues in order to learn about current organizational systems and their functionalities. These meetings were in an apprenticeship mode with the analyst sitting next to the IT manager or his colleagues as they demonstrated the organizational systems, often referring to visuals such as data flow charts or system walkthroughs. While the analyst's understanding of existing LifeSci systems mostly occurred in the scoping phase, he also informally solicited more information from his colleagues throughout the project.

> *Well, he did learn it as you need to know it, right, but it takes away a lot from the project where you're on a roll and you're doing things and then oops, you've come to a stop because you don't know how to do something from this point on to the next point* (IT Programmer).

The transmission activities between the IS analyst and his IT colleagues can be characterized as inquiry with the analyst seeking to discover the existing support environment (i.e., problem frame domain entity) (Byrd et al., 1992) in order to perform his work. As can be seen in Table 5, the analyst engaged in approximately 170 hours of transmission activities related to scoping.

These Model 2 inquiry transmission activities helped the analyst gain a high level of understanding of the problem frame domain, and his knowledge of organizational systems impressed the IT manager and several users.

> *I think that the* [...] *tool itself, which was written in JavaServer Faces by* [IS analyst]... *I think that piece broke new ground. We're connecting to the mainframe legacy, the older system, and we're the first - one of the first - to implement this real-time call to the mainframe and get the data displayed on some Java front-end piece. That piece was complex. There are other similar tools that do that, but in an applet, not in a JavaServer Faces environment. Just our own internal technology was a barrier until that piece became available. So that presentation layer to the end user was complex. And then, the number of interfaces that we have to extract data* [...] *each little piece on its own was minor but all together it's complex* (IT Programmer).

> *I can't believe how quickly he was brought up to speed about complexities of the system and wrapped his head around it. It was amazing* (Technician 2).

5.1.3. The Sense-Making Phase

The sense-making phase corresponds to the interactions the analyst had with the users in order to learn their system requirements. Initially, the analyst scheduled 90 minute meetings with a few technicians to ask questions and to have them explain and demonstrate how they accomplished their tasks.

> *So we'd* [selected technicians] *show* [the IS analyst] *the Excel file* [...] *Just a lot of question and answer and just sit back and let us do our thing and he would intervene and basically question as necessary* (Technician 2).

The analyst's observations to understand users' business processes and information requirements reflect an inquiry activity, where, by observing some users perform their work, the analyst could test the validity and his understanding of the information he received by asking the users to articulate and demonstrate their actions (i.e., Model 2 inquiry transmission activities). According to Technician 2, this activity (also shown in Table 6) seemed to be the most useful for the analyst "to wrap his head around" the business processes and information requirements he observed.

Further, the LifeSci analyst developed the system in an iterative process following a general Agile approach. He scheduled weekly one-hour meetings with the technicians to provide them with a status update of the most recent system version, demonstrated the system interfaces using a design prototype, and solicited feedback on the components to be incorporated into the system's interface design. Several users noted that this approach was better than the structured development techniques previously used at LifeSci.

> *And that's different than what IT had done in the past.* […] *they'd spend a month visiting people or talking and getting requirements and then they're closed-door for six months* […]*. I think it was good that* [the IS analyst] *had these regular meetings...* (Technician 1).

The users also noted that, during these meetings, there were "pretty good interactions on the tool" with extensive discussion about users' information requirements. Indeed, emails listing the items discussed in some of the meetings clearly documented that, apart from one exchange regarding the system's response time, all of the other interactions were about how information would be presented in the system's interface. Specifically, the analyst presented to the users a design prototype with simulated data, asking them and discussing about the design of the interface and the information they would like to see on it.

> *An hour* [meeting] – [IS analyst would say] *here's the thing, here's the fields I'm looking at, what I can get, what I can't get, and then when* [the IS analyst] *had something iteratively working he would use a fake data set and bring it up and say, "ok, this is what you're gonna see", and then people could say, "I would like this over there and could you turn this on and allow access for these people but not these people."* […] *Yeah, there was a pretty good interaction on the tool* (Technician 1).

> [These meetings were] *just kind of touch base – a 'what it* [the system] *looks like' sort of thing* (Senior Technician).

> *There were weekly meetings that* [the IS analyst] *had.* […] *most of those meetings were just to do with the sequencing tool itself – what the front end would look like* (IT Programmer).

Thus, during these communications with the users, the analyst presented the design prototype, advocating his design of the system interface while seeking feedback about users' preferences. Further, in presenting the design prototype, he allowed the users to confront his ideas about interface design. As can be seen in Table 6, as a result of Model 2 advocacy transmission activities, these meetings seemed to enhance the analyst's knowledge of users' information requirements, and he ultimately did provide the users with an interface design that contained data elements that met all of their needs.

> [The developed system] *has more features* [than the old one] *and the fact that it's all tied into one screen - as far as pieces of data that are missing from the* [old system…]. *so there's so many benefits* (Technician 2).

> *It's* [system's interface quality is] *good. People* [users] *like it, it's very flexible.* […] *there is a whole series of information-type fields,* […] *comments and all kinds of other information. And you can check – using SharePoint-type functionality on that display screen – so you can determine which columns you do want to see individually. Each* [technician] *can say, "I'm not interested in this, I am interested in this", so they can modify the report* (Senior Technician).

Still, while the analyst presented a design prototype, the technicians did note some functionality issues during these meetings. For example, one issue that was raised concerned embedding sensitivity analysis capabilities into the application. However, the importance of this functionality for the users was never discussed.

We identified that the [sensitivity analysis functionality was] *not in there. I have in a note from one of the sessions […] that said, you know, Issue Number 9 of 12 that we talked about was we're not going to have* [sensitivity analysis] *in there* […]*. [The IS analyst] never did send back notes either, saying, this is what are the issues raised and we are not going to fix this one* (Technician 1).

It [sensitivity analysis functionality] *was mentioned a couple times but again, we kind of expected it to happen, and talking with the guys in IT, it was always feasible* (Technician 2).

The promise was there in writing that they would deliver and I didn't have any doubt that they would deliver (Senior Technician).

Although some users expressed a need for this functionality during the meetings, the analyst viewed it to be outside the project's scope, and therefore decided to exclude it from the new system's design without discussing it with the technicians.

Table 5. IS Analyst Communication and Learning Activities at LifeSci During the Scoping Phase			
Transmission behavior	**Analyst's transmission behavior**	**Characteristics of transmission behaviors (Model 1 / Model 2)**	**Information content**
Inquiry	Informal meetings with the IT manager (122 hrs)	Model 2: [with observable data, and with testing] - Analyst acquired information using actual system walkthroughs and flowcharts.	Problem frame - existing organizational systems
Inquiry	On the job training and informal meetings with IT colleagues (48 hrs)	Model 2: [with observable data, and with testing] - Analyst acquired information using system walkthroughs.	Problem frame - existing organizational systems

Table 6. IS Analyst Communication and Learning Activities at LifeSci During the Sense-Making Phase			
Transmission behavior	**Analyst's transmission behavior**	**Characteristics of transmission behaviors (Model 1 / Model 2)**	**Information content**
Inquiry	Observing users perform task (6 hrs)	Model 2: [with observable data, and with testing] - Analyst tested the validity of received information by asking users to explain and articulate their actions as they were doing them.	Information requirements and business processes
Advocacy	Weekly system presentation meetings (8 hrs)	Model 1: [without observable data, without testing, without feedback solicitation] - The analyst presented to the users a design prototype which identified the information content of the interface. He did not show them the production prototype, which would have demonstrated the system's functionalities.	Business processes
		Model 2: [with observable data, with testing, and with feedback solicitation] - Analyst presented his view of the system information elements using the observable and testable design interface prototype, while soliciting user feedback.	Information requirements

While the analyst had the requisite information about the need for sensitivity analysis functionality, his approach was not Model 2 since he did not test or observe its validity, nor did he further inquire about its user-perceived importance. Hence, since the technicians had not seen a functional prototype before the launch of the system, they were left unaware of the analyst's decision and were unable to confront him about the importance of this functionality.

Therefore, while the meetings focused on information requirements, the analyst simply noted the business processes and related functional requirements, but did not discuss them or seem to realize their importance for the users and how they could influence their business processes, which Table 6 also notes.

> *But I did bring it up* [the need for sensitivity analysis] – *they just didn't understand. I believe that they really didn't understand what I was saying* […] *that the tool wouldn't provide a* [report] *that would be usable.* [...The analyst] *didn't understand the impact on the project* (Senior Technician).

5.1.4. Project Dissension and Termination

The technician who piloted the system for one week considered it unusable, noting that while the system provided adequate data analyses and reports for the linear and formalized quality control process of production batches, it did not have the sensitivity analyses needed for new product development tests. The senior technician outlined this issue in an email message: "The way it is designed right now, […it] is not useful".

To address this shortcoming, the IT department provided a workaround that allowed technicians to manually change the initial data that had been entered into the system. But, even with this modification, the system could not recalculate test results or outcome reports, so the technicians were allowed to manually alter these parameters and outputs. While this was not an issue for the technicians responsible for some product lines, it was not a viable solution for others who continued to use their old spreadsheets to manage QC testing because they found it easier than the workaround for their sensitivity analyses. They only used the part of the new application that uploaded data to the corporate database. In sum, the workaround was adequate for some technicians, but did not meet the needs of others.

As for the system's technical design, the users and their representatives found the system easy to use and user friendly, and were generally satisfied with the user interface and the features incorporated into the system. The system's modular design and customizability made it easy to use:

> *I find it fairly easy to use* [the system]. *I'm kind of impressed.* […the IS analyst] *brought a lot to the table that I felt was new to the offering that we haven't seen in other systems or platforms yet* (Technician 2).

The project had no formal timeline, but informal discussions between the ITPM and the technicians had suggested that a functional product would be available by September. However, this date was extended to allow time for the second release that included the workaround solution, and delayed the final roll-out to February of the following year.

> *No,* [the] *implementation was not on schedule* (Senior Technician).

> *The timeline was definitely probably a lot shorter than what we are currently looking at* (Technician 2).

5.2. Case 2 Overview: Regional Insurer

In an opportunity analysis review, the IT and user groups of Regional Insurer jointly identified their billing system as a high-priority project. The billing system was a mid-1980s DOS application that helped manage billing transactions and loans between Regional Insurer and its broker network. Two factors made it a high priority project: its platform was becoming increasingly ancient with each new version of MS Windows, and organizational knowledge about the system was in danger of being lost because the accounts receivable director who had developed the system was planning to retire in a couple of years.

Once top management approved the project, the IT group decided to develop and implement a secure web application to facilitate its interface with existing and future web systems. To do so, the IT group and the project manager decided to use an Agile methodology rather than the classical methods the firm usually relied on. However, because they had little experience with Agile methods, they hired an IT consulting firm experienced with the methodology. The consulting firm's analyst (hereafter labeled Regional Insurer analyst) was asked to work with the Regional Insurance user manager who, in turn, was both the product and project owner and was responsible for interacting with the Regional Insurer's IT group.

The Regional Insurer analyst had three years of experience in Agile methods and over ten years as a programmer. Despite being unfamiliar with Regional Insurer and having little background in finance, he delivered a system that the user group widely accepted and appreciated and did so two months earlier than initially scheduled.

5.2.1. Communications of the Regional Insurer Analyst

As can be seen in Tables 7 and 8, the Regional Insurer analyst gathered information primarily from the user manager and his group via formal and informal activities that focused on obtaining specific information about business processes and the features to be incorporated into the system. Following the Scrum Agile methodology, he divided the project into three-week cycles labeled sprints during which he daily and weekly interacted with the users to understand their needs, evaluate progress, and present system demos.

5.2.2. The Scoping Phase

As he knew little about finance, the Regional Insurer analyst spent half of every day with the user manager during the first two weeks to learn about the area, its business processes, and vocabulary. However, realizing that he needed more information than the user manager could provide, he also consulted other users.

> *During those two weeks* [beginning of project] *I think we were meeting almost each day, maybe half a day each day - in the morning* […]*. What we do in modeling is try to establish a common language; close that gap between IT and business* […]*. Kind of try to work up or establish a glossary in really common language we can work with* (IS Analyst).

> *We realized at this point that the manager didn't have all the information* […]*, so we felt that we needed to have some power users to really . . .* [discover] *"what's going on in the field"* […] (IS Analyst).

In addition, he spent a day each with two users to observe and ask questions as they performed their daily tasks. He noted that, in these initial interactions, he focused on understanding each user's business process and tasks by asking them to explain "what [they] are doing" without discussing the new system.

Thus, by asking users to describe their business processes and how they executed them, the Regional Insurer analyst was seeking concrete examples and testing the validity of the acquired information. He also undertook observations that were Model 2 inquiries since the validity of the

provided information was demonstrated. Although the analyst had no finance background, he quickly acquired business domain knowledge and a high level of understanding of the user vocabulary as a result of his Model 2 inquiry transmission activities, swhich Table 7 shows.

> [Because they were external consultants] *I was surprised that they understood very well the business needs so quickly* (User 1).

5.2.3. The Sense-Making Phase

In the next phase, the Regional Insurer analyst undertook transmission activities that focused both on users' work processes and the system features they required. He met with the user group once a week for half a day throughout the project to identify their specific business needs and the system functionalities and features they desired. These meetings focused on "user stories", in which users explained and discussed their particular work tasks and their value. Specifically, the stories were uncovered by executing the users' tasks in the existing system, while simultaneously modeling the process on a white board. The group discussed the process and its business value, and debated how it could be improved. Once the process was agreed, the development team entered the user stories on index cards to ensure they were incorporated into the new IS. The use of such dialogue and idea elicitation shows that the Regional Insurer analyst relied on inquiry to acquire information about user business processes and information requirements. Further, during meetings he tested the validity of the acquired information by performing the tasks in the current system and asking users the value of each process to the business.

> [While] *we try to avoid implementation here* [during the initial two-week scoping period], *we obviously want to discuss* [a] *specific solution, but we just want to clarify the business needs* [first] *then we are going to see how we can do it* (IS Analyst).

The analyst viewed these meetings as a "base for discussion between IT and business people" because they enhanced his knowledge of the users' business processes and information requirements, which then informed the system's design and features. The users noted that, as a result of the analyst's Model 2 inquiry transmission activities, the functional prototype he delivered clearly showed that he understood their business and information requirements (Table 8).

Moreover, at the end of each cycle, the development team presented the functional prototypes during "iteration review" meetings to project stakeholders, the user group, the project manager, and the finance manager. The Regional Insurer analyst demonstrated specific system functionalities (i.e., "user stories") by walking through the process of executing specific tasks in the new system, while soliciting feedback from those present. As can be seen in Table 8, these meetings evidenced the analyst's knowledge of the users' business processes and information requirements. Further, they also fostered a user-IT dialogue on business processes and system functionalities, helped uncover features to be incorporated into the system, and revealed users' own conflicting views on business processes and the new system's goals. Hence, the analyst undertook Model 2 advocacy transmission activities in the meetings by advocating his view of the system design with observable data to demonstrate how the task would be performed via a functional prototype, and by soliciting user feedback that was testable and grounded in the prototype data.

> *We have at the end* […] *each three weeks, there is a demo. We are showing them a piece of the software and that is really powerful to generate feedback and for closing that gap because this is where they start to understand what we can provide to them* (IS Analyst).

> [During the demos] *they* [IS group] *gave us a course: "how it will function, will it do your work, is there something in it that does not work, are there things we can improve?" it went like that. Then if there was a bug,* […] *they came…"if we do this like that, what is the response that you want, what does your business need?" (User – tr[4]).*

---

[4] Quotes with "tr" indicate translation from transcripts of interviews held in French. The interview with the IS analyst was partly in English and partly in French.

Table 7. IS Analyst Communication and Learning Activities at Regional Insurer During the Scoping Phase			
Transmission behavior	**Analyst's transmission behavior**	**Characteristics of transmission behaviors (Model 1 / Model 2)**	**Information content**
Inquiry	Preliminary meetings with users and their manager (40 hrs)	Model 2: [with observable data, with testing] - The analyst acquired information by asking the users to provide specific examples of their work processes.	Business processes
Inquiry	Observing users perform their tasks (16 hrs)	Model 2: [with observable data, with testing] - Analyst tested the validity of received information by asking users to explain and articulate their actions as they were doing them.	Information requirements and business processes

Table 8. IS Analyst Communication and Learning Activities at Regional Insurer During the Sense-Making Phase			
Transmission behavior	**Analyst's transmission behavior**	**Characteristics of transmission behaviors (Model 1 / Model 2)**	**Information content**
Inquiry	Weekly group meetings (180 hrs)	Model 2: [with observable data, with testing] - The analyst solicited detailed information about users' work processes, while asking them to demonstrate their work processes by using the existing system.	Information requirements and business processes
Advocacy	End of 3-week system presentation meetings (45 hrs)	Model 2: [with observable data, with testing, with feedback solicitation] - Analyst presented his view of the system design using a functional prototype, while seeking users' feedback about it.	Information requirements and business processes

5.2.4. Project Dissension and Termination

The prototype presentations were particularly useful for uncovering conflicting requirements between users and the analyst, and among the users themselves. For example, during a requirements planning session the analyst had identified the process of entering checks into the system, but once he presented the prototype, the users noted that the proposed design would make their work more tedious. As the issue was identified prior to the development of additional system functionality, it became easy to redesign the module for the next prototype demonstration.

> *I remember that during that planning* [session] *when we selected the* [user case] *story of typing in cheques, I'm sure I explained that "this is what will happen, you will select a broker and then in that broker you will select the cheque form and then you are going to type it and save". And they all said fine, it was fine to them* [then] (Regional Insurer IS Analyst).

According to the project team, stakeholders, and the users, the project was successful in terms of both its process and product.

> *We were supposed to deliver the system by the end of April; well they have already had the system in their hands for nearly a month and half* (IS Analyst – tr).

The users and their manager were also happy with the delivered system and felt that it met their needs, was easy to use, and better than the previous system.

> *The users are satisfied and I have never seen the users satisfied* (IS project manager).

> *Absolutely! As I am telling you, it* [the IS] *responds to all the team's needs.* […] *it is very user friendly. It is not complex to work with* (User-tr).

6. Analysts' Communication Activities and Learning about Information Requirements

As can be seen in Table 9, the contextual similarities between the two projects facilitate a within-case analysis of the variables of interest (Yin, 2003); namely, analysts' communication activities and project outcomes. Both projects were carried out in large subsidiaries of multinational organizations in order to replace existing systems, were a high priority in both organizations, received top management support, and benefited from the participation and involvement of expert users. Moreover, both analysts followed an evolutionary development methodology by using similar transmission activities and communication channels. Yet, despite his lack of experience with the business domain, the Regional Insurer analyst was able to deliver a more successful system than his counterpart at LifeSci. Our analysis suggests that the different outcomes observed in the two projects largely resulted from what each analyst learned and their transmission activities.

According to Argyris and Schon (1974, 1978), Model 1 behaviors inhibit the detection and correction of knowledge deficiencies that make action ineffective and the events that transpired in LifeSci are consistent with this proposition. Although the LifeSci analyst noted the sensitivity analysis issue raised by the technicians, the manner through which this requirement was identified and decided on (i.e., Model 1 transmission behavior) did not enable the analyst to see how important this functionality was for the users in their tasks. In particular, the analyst did not anticipate how his decision to exclude this functionality would impact the users' tasks. As the Senior Technician stated, "the [IS analyst] didn't understand the impact [of his decision not to include the sensitivity analysis function] on the project".

Thus, by not encouraging users to provide concrete examples about the sensitivity functionality they requested and to justify its importance, and by not soliciting their feedback about his decision to exclude it from the system, the LifeSci analyst acquired little knowledge about this process requirement, which in turn resulted in task-technology fit issues and project delays due to subsequent patch up modifications, and ultimately to an underused system. While a large part of the users' requirements were met, neither the users nor the LifeSci analyst were able to identify his knowledge deficiency concerning how important this functionality was for the users because it was identified via Model 1 transmission activities.

> *There is probably 10 critical things we were worried about, and they knocked down 9 of them* (Technician 2).

> *As is right now the satisfaction* [with system] *is zero.* […]. *I guess to that extent - what the original requirement was – we are 75% of the way there* (Senior Technician).

In contrast, the Regional Insurer analyst undertook extensive Model 2 communication activities with the users in order to understand their business processes and information needs. More notably, his inquiries asked users to justify the validity of the information they provided by explaining the value of their processes and showing their accuracy via the existing system. Further, he also demonstrated the new system and how it would support users' business processes during end-of-cycle system meetings, which either enhanced his knowledge of users' business processes and information requirements or helped him modify it In addition, the Regional Insurer analyst frequently solicited users' feedback during meetings in order to validate his knowledge of their work, and to help them reach a consensus concerning their business processes and desired system functionalities. Hence, by engaging in Model 2 inquiry and advocacy transmission activities, the analyst promoted the generation of complete and valid information as users validated and identified gaps in his and their own knowledge.

Table 9. Assessing the Influence of Rival Explanations on Project Outcomes			
Rival explanation	**Factor description**	**Influence in both cases**	**Evaluation of factor's influence on project outcomes**
Project's priority for the organization	The project's priority and criticality for the organization, and level of top management support for it.	Both projects were deemed as high priorities in both cases. The LifeSci system was identified as the top priority project for the IS department, and the Regional Insurer project was deemed critical. *[The IT PM] brought it out to [Manufacturing VP…and] this was [LifeSci's] number one issue.* [LifeSci Technician 1] *There was risk at each version of Windows where we are not sure if the application will continue to exist because it's under DOS. The knowledge to maintain the application or to fix bugs only is only in the head of one user and this user had planned to retire.* [Regional Insurer PM]	*Unlikely influence.*
Allocated resources	People and financial resources allocated to the project	The LifeSci analyst was assigned full time on the project, in addition to six other developers who worked on various parts of the system. *Because now we have [IS Analyst], [programmer 2], myself, [programmer 3], and [programmer 4]. [Programmer 5] was on and off for a while, [Programmer 6] was on and off for a while so you have all these people plus [IT PM].* [LifeSci Programmer] For Regional Insurer, there was the IS analyst and a developer who worked on the project full-time. *We were two from the IT side, in the beginning there were two [… then] there were three developers […] an average of three developers […].* [Regional Insurer Analyst] Both companies declined to divulge the financial resources invested in the project.	*Unlikely direct influence on project outcome.* Resource limitations are unlikely to have directly influenced project outcomes, but may have impacted them by limiting the intensity of the interaction analysts undertook, which in turn might have influenced project outcomes. For example, being under time pressure, the LifeSci analyst may have limited his communication activities with the users, resulting in a system that did not fully meet users' functional needs. However, there were no indications in the LifeSci analyst's interviews that his actions were a result of time constraints.

Table 9. Assessing the Influence of Rival Explanations on Project Outcomes (cont.)			
Rival explanation	**Factor description**	**Influence in both cases**	**Evaluation of factor's influence on project outcomes**
Task complexity	Complexity of the task to be supported by the system	The respondents in both cases noted that the user tasks were complex. *In my mind, [the task is] very complex [...] it just has a lot of relations with other systems, and integrations seem to be very key and at the same time detailed and complex.* [LifeSci Technician 2] *I would say that it was quite complex.* [Regional Insurer Analyst]	*Unlikely influence.*
ISD methods used	The procedures and techniques used to develop the system	While the ISD methodology of both cases was similar, i.e. Agile, it was not applied equally. For example, while both analysts developed the system in an iterative manner involving the users, some differences existed. More specifically, the Regional Insurer analyst used fully functional prototypes and focused on users' business processes through case stories. In contrast, the LifeSci analyst's prototype was not functional and served only as a tool for obtaining the users' interface requirements.	*Likely influence.* The differences in applying the methods are likely to have influenced the two projects. However, these differences can largely be explained by the proposed communication framework. For example, by using a fully functional prototype, the Regional Insurer analyst was able to communicate observable business process information with the users.
Communication channels used	The medium of communication used for exchanging information	Both analysts used identical communication channels to communicate with the users, mostly relying on face-to-face meetings and presentations, with very limited use of documentation or electronic communication. *There were just verbal meetings. Not a great deal [of emails].* [LifeSci Senior Technician] *So mainly 99% of the communication is face to face.* [Regional Insurer PM]	*Unlikely influence.*

Table 9 Assessing the Influence of Rival Explanations on Project Outcomes (cont.)			
Rival explanation	**Factor description**	**Influence in both cases**	**Evaluation of factor's influence on project outcomes**
Analyst characteristics	The analyst's systems analysis and development experience, and domain knowledge.	While the LifeSci analyst had nearly 10 years experience as a systems analyst and developer, the Regional Insurer analyst had been a systems analyst and developer for four years, and had been a programmer for six years prior to that. Further, while the LifeSci analysts' 10 years experience had been in quality control laboratory systems, the Regional Insurer analyst had no knowledge or experience in finance or financial systems. Additionally, both analysts were new to their respective organizations.	*Unlikely influence.* Given his more extensive experience in systems analysis and domain knowledge, the LifeSci analyst should have delivered a more successful project than the Regional Insurer analyst. The fact that the opposite actually transpired reinforces the present study's findings regarding the influence of the proposed communication framework,
Political / social issues	Conflict or trust issues between both groups	While users in both projects expressed dissatisfaction with previous IS implementation experiences, they all had positive attitudes towards the process undertaken by both analysts. No evidence of conflict or distrust between the two groups could be found. *And that's different than what IT had done in the past. […] they'd spend a month visiting people or talking and getting requirements and then they're closed-door for six months […]. I think it was good that [the IS analyst] had these regular meetings....* [LifeSci Technician 1] *In 40 years, this was the most beautiful project I had, because we had the power to decide, we had the power to participate.* [Regional Insurer Users' Manager]	*Unlikely influence.*
User participation	The participation of users in the IS project	All users at Regional Insurer participated in the sessions, while some LifeSci users participated.	*Unlikely influence.* Although some LifeSci users who participated in the sessions did not use the system as it did not match their needs, other users who did not attend the sessions used it.

Thus, it is likely that the Regional Insurer analyst's knowledge of users' business processes resulted in the delivery of a system that met users' organizational and task needs while adhering to project schedules and budgets. In fact, the users' manager also confirmed that the Regional Insurer analyst's knowledge of the users' business processes was extensive, that he could execute users' tasks easily, and even act as their supervisor if necessary.

> *What* [the accounts receivables supervisor] *does, just the operations…the analyst would be able to do* (Users' manager - tr).

> *I could supervise the accounts receivable operators, I know what they do* (IS Analyst – tr).

The analysis of the two cases also support earlier studies that note that the usefulness of an IS depends in part on how well its designed functionality supports task demands (Dishaw & Strong, 1999; Goodhue & Thompson, 1995). The findings indicate that analysts are more likely to make system functionality design choices that meet users' task needs when they identify their own knowledge deficiencies about users' business processes, which can be more readily identified via Model 2 transmission activities. For example, the Regional Insurer users' manager stated that the analyst's attempts to identify his knowledge deficiency of the users' business processes were extensive:

> *Sometimes I express some business issues the* [IS analyst] *does not understand, but if he does not understand and does not tell me that he did not understand, then it is certain that there will be a problem somewhere. But the* [IS analyst] *asks me 322 questions to be sure that he has well understood* (Regional Insurer Users' Manager - tr).

Thus, the present study's findings support, in ISD contexts, Argyris and Schon's (1974, 1978) argument that Model 2 activities result in more effective decision making and suggest that it is particularly Model 2 communication activities about users' business processes that are likely to yield effective design decisions that meet users' task needs. Hence:

Proposition 1: *IS analysts' Model 2 transmission activities about users' business processes is more likely to promote analysts' knowledge of users' system task needs, and will in turn more positively influence the developed system's task-technology fit and implementation process success than Model 1 transmission activities.*

6.1. IS Analysts' Transmission Activities

The communication activities summarized in Table 10 indicate that, while both analysts undertook essentially Model 2 transmission activities to learn about users' system needs, they also focused on different project issues, which likely resulted in the different project outcomes. More specifically, while there was little difference between the two analysts' transmission activities from a Model 1/2 perspective, the case data suggest that they had different assumptions about their respective projects, and in turn focused their attention on different project issues. This observation is also consistent with the view that individuals' activities and communication messages are governed by their assumptions (Argyris & Schon, 1978) and suggests that integrating the Model 1/2 learning framework with the literature on analyst assumptions and orientations can provide a richer understanding of analysts' communication activities in IS projects.

According to Hirschheim and Klein (1989) and Markus and Benjamin (1996), IS analysts' activities and communication messages in ISD are a function of their assumptions toward the project, its objectives, and the roles of its stakeholders. These orientations are summarized in Table 11 and can be described along different change agentry roles[5]: traditional or systems expert, facilitator, and advocate[6]. Systems expert analysts are thought to assume that different stakeholders share well-articulated system goals that are set by management and which aim to improve organizational efficiency and effectiveness (Hirschheim & Klein, 1989). Viewing their role as a provider of technology, systems experts direct their efforts to "productive technical solutions" (Hirschheim & Klein, 1989, p.

---

[5] It is important to note that these roles constitute theoretically ideal profiles. The actual profiles of most IS analysts may not exactly correspond to them. In fact, many analysts may only tend towards a given profile, but exhibit different profiles at different times (e.g., adopt different roles across different projects or across the different stages of a project).
[6] While Hirschheim and Klein (1989) propose a fourth role, that of the emancipator, they acknowledge it as being difficult to observe and assume in practice, and therefore to be rather hypothetical.

1204) (i.e., problem frame issues). On the other hand, facilitators are thought to believe that reality is complex and subjective, viewing their role as helping users make sense of reality (Hirschheim & Klein, 1989; Markus & Benjamin, 1996). Because they help clients reconcile individual and group interests in order to reach agreement about system needs (Hirschheim & Klein, 1989; Markus & Benjamin, 1996), facilitators engage in extensive transmission activities to learn users' tasks and system needs.

In both cases, stakeholders identified divergent and ambiguous system needs, and our analysis suggests that directing learning efforts to users' business processes is more appropriate for determining system functionality than problem frame issues. For example, the Regional Insurer analyst avoided setting objectives at the beginning of the project, and felt that system objectives and functionalities would emerge during his discussions with the users about their business processes.

> [While] *we try to avoid implementation here* [during the initial two-week scoping period]*, we obviously want to discuss* [a] *specific solution, but we just want to clarify the business needs* [first] *then we are going to see how we can do it* (IS Analyst).

> *We don't want change management; we want change to just be part of the process* […] (IS Analyst).

Further, even though the users of Regional Insurer had divergent interests, the analyst mainly relied on extensive Model 2 transmission activities to learn users' business processes. He thought that extensive dialogue and discussion with the users about their processes was necessary to allow them articulate their business needs and reach consensus on the desired future state of the business processes and the system to support them.

> *Obviously, they* [users] *don't know exactly and they have their own contradictions within how they work.* […] *I'll try to make sure to* […] *revisit or clear those contradictions* (IS Analyst, Regional Insurer).

> *During planning* [weekly sessions] *we had a solution, we get to review* [demo meeting]*, we do the demonstration,* [the accounting manager] *says "I am not sure, I think there is something missing, blablabla...", they discussed it; we did an intense backlog in the following iteration, and redesigned the system* (IS analyst Regional Insurer - tr).

> *So, it's really up to them* [the users] *to tell us what we want to produce* [in terms of the developed system] (IS Analyst, Regional Insurer).

Thus, as depicted in Table 10, the Regional Insurer analyst closely followed a facilitation approach that relied on extensive Model 2 transmission activities with users in order to better understand their business processes, help them clarify their objectives, and reach consensus regarding their business needs, which in turn informed the system's design and features, resulting in a widely appreciated system.

The users of LifeSci also had divergent needs and methods of performing their business processes. However, in contrast to the Regional Insurer analyst, the LifeSci analyst's extensive Model 2 transmission activities were with the IT group and the ITPM, and oriented towards better understanding organizational systems (i.e., problem frame). As shown in Table 10, he engaged in only a few hours of transmission activities with users to learn about their business processes.

> *There's some discrepancies within the group. Some* [technicians…] *didn't have the need for the* [sensitivity analysis] *and some did* (LifeSci Technician 1).

Table 10. The Learning Models Used by the LifeSci and Regional Insurer Analyst

Phase	Transmission behavior	Object of learning			Number of hours (approximate)
		Users' information requirements	Users' business process	Problem frame – existing organizational systems	
LifeSci					
Scoping	Inquiry			Model 2	170
Sense-making	Inquiry	Model 2	Model 2		6
	Advocacy	Model 2	Model 1		8
				Total hours	**184**
Regional insurer					
Scoping	Inquiry		Model 2		56
Sense-making	Inquiry	Model 2	Model 2		180
	Advocacy	Model 2	Model 2		45
				Total hours	**281**

Table 11. Beliefs (Ideal Type Dimensions) Represented by Different Role Orientations

IS Analyst change agentry roles	Systems expert	Facilitator
Nature of system objectives	*Articulated and shared* (improve efficiency and effectiveness)	*Ambiguous and not shared* (unknown upfront, acceptance of change)
Responsibility for system implementation objectives	*Managers*	*Clients*
IS analyst function	*Designer/developer of technology* (technical expert)	*Process facilitator*

Thus, the LifeSci analyst closely followed a "systems expert" approach in that he viewed his primary responsibility to be a provider of technology that brings well-defined benefits as determined by the project leader and management. The LifeSci analyst clearly adhered to the system functionality and "project scope" that were identified by the IT project manager, which in turn resulted in a system that did not respond to the needs of all users. Hence, the case findings suggest that, when stakeholders have divergent or ambiguous needs, analysts may need to direct their transmission efforts more towards users' business processes if they are to develop systems that better support users' business process needs.

Proposition 2: *When user needs are divergent or ambiguous, Model 2 transmission activities of IS analysts that are oriented towards learning users' business processes will be more helpful than their Model 2 transmission activities that are oriented towards learning the problem frame.*

It is important to note that these findings may be limited to the context of the projects studied here. For example, Markus (2004) differentiates between business IS projects that impact users' business processes, such as the two projects examined in this paper, and technical IS projects that are intended to improve technical efficiency and reliability. Unlike technical IS projects, Markus (2004) argues that business IS projects influence users' processes and thus require "a different kind of attention to the features of the '*solution*' and a different change process" (p. 5, italics in original). Indeed, the analysis of the two cases suggested that a "facilitation" approach would be more suitable for business IS projects than a systems expert approach, as depicted in Table 12.

Even if users identify shared and well-articulated system needs during business IS projects, there are at least two reasons why a facilitation approach is still likely to be superior to a systems approach. First, a facilitation approach can educate users about the potential benefits and changes the system will bring to their work (Kirsch & Beath, 1996). Second, the facilitation approach enables users to be extensively engaged with the project, which in turn can increase their involvement and acceptance of the system (Hartwick & Barki, 1994). Hence, notwithstanding new product development research that suggests that simple codified information requires limited interactions with the source (Hansen, 1999), we believe that a facilitation approach is more likely to lead to success in business IS project contexts.

In contrast to business IS projects, technical projects are assessed based on their operational performance and adherence to budgets and schedules rather than user acceptance (Markus, 2004). Accordingly, systems experts are likely to be more suitable for technical projects than facilitators given that they direct their efforts to transmission activities about the technological environment needed to support the system objectives as 'pre-specified' by management (Hirschheim & Klein, 1989). Further, systems experts could engage in Model 1 or 2 transmission activities to identify the most efficient and effective technical means of achieving the technical project's shared objectives. For example, systems experts are likely to use Model 2 transmission activities when system development options are complex or numerous in order to learn about them, while relying on Model 1 activities when technical options are few. In either case, the systems expert's focus remains on the technical means of achieving a set objective.

Table 12. IS Analyst Role Orientation and Transmission Activities Associated with Type of IS Project and Nature of System Needs			
		Nature of system needs	
		Shared / unambiguous	**Shared / unambiguous**
Type of IS project	**Technical IS project**	Systems expert, Model 1	Systems expert, Model 2
	Business IS project	Facilitator	Facilitator

Thus, while the IS analyst change agentry research has noted the importance of "the basic orientation toward the goals and means of IS work that shapes what the practitioner does" (Markus & Benjamin, 1996, p. 387), the present study extends this literature by providing initial empirical evidence regarding the relationship between project outcomes and IS analysts' role orientation, communication transmission activities and their content when stakeholders' needs are divergent or ambiguous. In addition, the present study combines the role orientation literature with that of Model 1/2 learning to provide a conceptual framework of analysts' communication activities and role orientations that can be useful for generating multiple propositions for future research.

7. Conclusion

The present study's findings suggest that what IS analysts learn about users' information needs influences ISD outcomes. While past research has often suggested that analysts' knowledge of users' business processes results in system success (e.g., Byrd et al., 1992; Curtis et al., 1988; Gallivan & Keil, 2003; Markus & Mao, 2004), surprisingly little empirical evidence or theory exist in this regard. To address this important gap, the present study integrated the concepts of Model 1/2

learning with the communication and role orientation literatures to provide an initial empirical examination and theoretical groundwork of IS analysts' communication activities and their influence on IS project outcomes.

The present study's findings indicate that IS analysts who use Model 2 transmission activities to learn about users' business processes are more likely to develop systems that meet users' task and organizational needs than analysts who use Model 1 transmission activities. More specifically, analysts who actively encourage the use of concrete examples, testing and validation, and feedback solicitation regarding the actions users undertake to execute their tasks are more likely to identify their own knowledge deficiencies about the users' tasks, which in turn can help them design and develop systems that meet users' needs. By investigating the influence of analysts' Model 2 transmission activities on their knowledge of users' needs and on project outcomes, the present study also extends past research that examined the influence of analyst behaviors on the detection of information errors (Salaway, 1987). Our findings generally support past research that suggests that it is the "nature and quality" of user-analyst communication that matters (Gallivan & Keil, 2003, p. 37; Boland, 1978; Majchrzak et al., 2005; Markus & Mao, 2004; Salaway, 1987), while extending it via the identification of a number of specific communication activities that analysts can adopt.

Further, our analysis also suggests that, when users' system needs are ambiguous or divergent, analysts' who use a facilitation approach are more likely to develop systems that meet users' needs than analysts who focus on understanding problem frame issues. These findings extend the IS analyst change agentry literature (Hirschheim & Klein, 1989; Markus & Benjamin, 1995) by further clarifying the relationship between analysts' communication content and project outcomes.

The present study's findings are also significant for IS practice and curriculum, since their training and education can be designed to emphasize the importance of learning users' business processes in achieving successful project outcomes, and to encourage IS analysts in organizations to view themselves as supporters of business processes rather than just technical designers.

7.1. Limitations and Suggestions for Future Research

The present paper's limitations also need to be acknowledged. Because our study relied on retrospective data, a potential limitation is respondents' imperfect recall of events. In order to minimize this validity threat, respondents were asked during the interviews about the project's factual events, and to mainly describe the activities undertaken by the analysts and the resulting outcomes. Further, the collected data was triangulated via multiple sources to validate "what had happened", which is likely to minimize this limitation. A second limitation is the different duration of each IS project. While the LifeSci development was initially a six-month project, the Regional Insurer development lasted 18 months. Hence, the lengthier project duration may have provided the Regional Insurer analyst with more time to adequately inquire and seek feedback about users' business processes. While the time difference does not threaten the validity of the present study's findings, it can help explain the conditions that are needed for adequate communication. Another potential limitation is that the selected case studies were conducted in the context of complex in-house ISD projects in large organizations. While this helped control several situational factors, it also might have limited the generalizability of the study findings. For example, Hansen (1999) found that, to understand complex tasks, a recipient "needs multiple opportunities to assimilate" and acquire knowledge, necessitating extensive interaction with the information source (p. 88). Thus, future research is needed to validate the present study's findings and to examine IS analysts' communication activities and outcomes in other contextual settings. Doing so could identify IS project contexts where Model 1 communication activities are more appropriate than Model 2 activities. Further, future studies controlling for contextual settings could also examine the impact of analysts' communication activities on different outcomes. For example, using Model 1 communication activities could result in shorter projects than the use of Model 2 communication activities.

Other potentially fruitful avenues for future research include extending the present study to other contexts. For example, the geographic and temporal distances that exist between analysts and users

in offshore software development sourcing and outsourcing contexts (Levina & Vaast, 2008) suggest that how analysts communicate to learn users' needs and how their learning influences ISD outcomes is likely to be different than the present study's context. Also, analyst communication activities may be facilitated via tools that provide observable and demonstrable information. It would therefore be interesting to examine the extent to which such tools can facilitate or hinder communication. Another future research opportunity would be to investigate the impact of institutional forces, such as the influence of contract types or management roles, on the roles and behaviors of analysts. It is hoped that the present study can pave the way for such research and provide a useful step towards a better understanding of how IS analysts think and behave, and how their thinking and actions influence ISD projects and their outcomes.

Acknowledgements

References

Agarwal, R., & Tanniru, M. R. (1990). Knowledge acquisition using structured interviewing: An empirical investigation. *Journal of Management Information Systems*, *7*(1), 123-140.

Agarwal, R., De, P., & Sinha, A. P. (1999). Comprehending object and process models: An empirical study. *IEEE Transactions on Software Engineering*, *25*(4), 541-556.

Al-Salem, L. S., & A. Samaha (2007). Eliciting web application requirements—an industrial case study. *The Journal of Systems and Software*, *80*(3), 294-294.

Andreou, A. S. (2003). Promoting software quality through a human, social and organisational requirements elicitation process. *Requirements Engineering, 8*, 85-101.

Appan, R., & Browne, G. J. (2010). Investigating retrieval-induced forgetting during information requirements determination. *Journal of the Association for Information Systems*, *11*(5), 250-275.

Argyris, C. (1976). Single-loop and double-loop models in research on decision making. *Administrative Science Quarterly, 21*(3), 363-375.

Argyris, C. (2002). Double-loop learning, teaching, and research. *Academy of Management Learning and Education*, *1*(2), 206-218.

Argyris, C., & Schon, D. A. (1974). *Theory in practice: Increasing professional effectiveness*. San Francisco, CA: Jossey-Bass Inc.

Argyris, C., & Schon, D. A. (1978). *Organizational learning: A theory of action perspective.* Reading, MA: Addison Wesley.

Argyris, C., & Schon, D. A. (1996). *Organizational learning II.* Reading, MA: Addison Wesley.

Argyris, C., Putnam, R., & McLain Smith, D. (1985). *Action science: Concepts, methods, and skills for research and intervention*. San Fransisco, California: Jossey Bass.

Arthur, J. D., & Gröner, M. K. (2005). An operational model for structuring the requirements generation process. *Requirements Engineering Journal, 10*(1), 45-62.

Arthur, J. B., & Aiman-Smith, L. (2001). Gainsharing and organizational learning: An analysis of employee suggestions over time. *The Academy of Management Journal, 44*(4), 737-754.

Athanassopoulou, P., & Johne, A. (2004). Effective communication with lead customers in developing new banking products. *The International Journal of Bank Marketing*, *22*(2/3), 100-125.

Baker, W. E., & Sinkula, J. M. (1999). The synergistic effect of market orientation and learning orientation on organizational performance. *Journal of the Academy of Marketing Science*, *27*(4), 411-427.

Baker, W. E., & Sinkula, J. M. (2002). Market orientation, learning orientation and product innovation: Delving into the organization's black box. *Journal of Market-Focused Management*, *5*, 5-23.

Banker, R. D., & Kauffman, R. J. (2004). The evolution of research on information systems: A fiftieth-year survey of the literature in management science. *Management Science*, *50*(3), 281-298.

Barki, H., Rivard, S., & Talbot, J. (1993). Toward an assessment of software development risk. *Journal of Management Information Systems, 10*(2), 203-225.

Baskerville, R., Levine, L., & Pries-Heje, J. (2006). High-speed software development practices: What works, what doesn't. *IT Professional Magazine*, *8*(4), 29-36.

Beath, C. M., & Orlikowski, W. J. (1994). The contradictory structure of systems development methodologies: Deconstructing the IS-user relationship in information engineering. *Information Systems Research, 5*(4), 350-377.

Beecham, S., Hall, T., & Austen, R. (2005). Defining a requirements process improvement model. *Software Quality Journal, 13*(3), 247-279.

Berlo, D. K. (1960). *Process of communication: An introduction to theory and practice*. New York: Holt, Rinehart and Winston.

Boland Jr., R.J. (1978). The process and product of system design. *Management Science, 24*(9), 887-898.

Boland Jr., R. J., & Tenkasi, R.V. (1995). Perspective making and perspective taking in communities of knowing. *Organization Science, 6*(4), 350-372.

Bolman, L. G., & Deal, T. E. (2008). *Reframing organizations: Artistry, choice, and leadership* (4th Ed.). San Fransisco, CA: Jossey-Bass.

Bostrom, R. P. (1989). Successful application of communication techniques to improve the systems development process. *Information and Management*, *16*, 279-295.

Breitman, K. K., Leite, J. C. S. d. P., & Berry, D. M. (2005). Supporting scenario evolution. *Requirements Engineering Journal, 10*(2), 112-131.

Browne, G.J., & Rogich, M. B. (2001). An empirical investigation of user requirements elicitation: Comparing the effectiveness of prompting techniques. *Journal of Management Information Systems*, *17*(4), 223-249.

Browne, G. J., & Ramesh, V. (2002). Improving information requirements determination: A cognitive perspective. *Information and Management, 39*, 625-645.

Bryant, J. (1997). Requirements capture using SODA. *European Journal of Information Systems, 6*(3), 155-163.

Butler, T., & Fitzgerald, B. (2001). The relationship between user participation and the management of change surrounding the development of information systems: A European perspective. *Journal of Organizational and End User Computing, 13*(1), 12-25.

Byrd, T.A., Cossick, K. L., & Zmud, R. W. (1992). A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly*, *16*(1), 117-138.

Carte, T. A., & Russell, C. J. (2003). In pursuit of moderation: Nine common errors and their solutions1. *MIS Quarterly*, *27*(3), 479-501.

Chakraborty, S., Sarker, S., & Sarker, S. (2010). An exploration into the process of requirements elicitation: A grounded approach. *Journal of the AIS*, *11*(4), 212-249.

Choudhury, V., & Karahanna, E. (2008). The relative advantage of electronic channels: A multidimensional view. *MIS Quarterly*, *32*(1), 179-200.

Churchman, C., & Schainblatt, A. (1965). The researcher and manager: A dialectic of implementation. *Management Science*, *11*(4), B69-B87.

Cleland-Huang, J., Dumitru, H., Duan, C., & Castro-Herrera, C. (2009). Automated support for managing feature requests in open forums. *Communications of the ACM* ,52(10), 68-74.

Curtis, B., Krasner, H., & Iscoe, N. (1988). A field study of the software design process for large systems. *Communications of the ACM*, *31*(11), 1268-1287.

Damas, C., Lambeau, B., Dupont, P., & van Lamsweerde, A. (2005). Generating annotated behavior models from end-user scenarios. *IEEE Transactions on Software Engineering*, *31*(12), 1056-1073.

Darke, P., & Shanks, G. (1997). User viewpoint modeling: Understanding and representing user viewpoints during requirements definition. *Information Systems Journal*, *7*(3), 213-239.

Davis, C. J., Fuller, R. M., & Berndt, D. J. (2006). Communication challenges in requirements elicitation and the use of the repertory grid technique. *The Journal of Computer Information Systems*, *46*(5), 78-86.

Davis, G. (1982). Strategies for information requirements determination. *IBM Systems Journal*, *21*(1), 4-31.

Day, J. (2007). Strangers on the train. *Information Technology & People, 20*(1), 6-31.

Decker, B., Ras, E., Rech, J., Jaubert, P., & Rieth, M. (2007). Wiki-based stakeholder participation in requirements engineering. *IEEE Software*, *24*(2), 28-35.

Dieste, O., Juristo, N., & Shull, F. (2008). Understanding the customer: What do we know about requirements elicitation? *IEEE Software*, *25*(2), 11-13.

Dishaw, M., & Strong, D. M. (1999). Extending the technology acceptance model with task-technology fit constructs. *Information and Management*, *36*, 9-21.

Dubé, L., & Paré, G. (2003). Rigor in information systems positivist research: Current practices, trends, and recommendations. *MIS Quarterly*, *27*(4), 597-635.

Duggan, E. W. (2003). Generating systems requirements with facilitated group techniques. *Human-Computer Interaction*, *18*(4), 373-394.

Duggan, E. W., & Thachenkary, C. S. (2003). Higher quality requirements: Supporting joint application development with the nominal group technique. *Information Technology and Management*, *4*(4), 391-408.

Duggan, E. W., & Thachenkary, C. S. (2004). Supporting the JAD facilitator with the nominal group technique. *Journal of Organizational and End User Computing*, *16*(2), 1-19.

Edmondson, A. (1999). Psychological safety and learning behavior in work teams. *Administrative Science Quarterly*, *44*(2), 350-383.

Edstrom, A. (1977). User influence and the success of MIS projects: A contingency approach. *Human Relations, 30*(7), 589-607.

Eisenhardt, K. (1989). Building theories from case study research. *Academy of Management*, *14*(4), 532-550.

Eisenhardt, K.M., & Graebner, M. E. (2007). Theory building from cases: Opportunities and challenges. *Academy of Management Journal, 50*(1), 25-32.

Farooq, U., Ganoe, C. H., Carroll, J. M., Councill, I. G., & Giles, C. L. (2008). Design and evaluation of awareness mechanisms in Citeseer. *Information Processing & Management*, *44*(2), 596.

Frolick, M. N., & Robichaux, B. P. (1995). EIS information requirements determination—using a group support system to enhance the strategic business objectives method. *Decision Support Systems, 14*(2), 157-170.

Galliers, R. D., & Swan, J. A. (1997). Against structured approaches: Information requirements analysis as a socially mediated process. In Proceedings of The Thirtieth Annual Hawwaii International Conference on System Sciences, IEEE Computer Society, Hawwaii.

Gallivan, M. J., & Keil, M. (2003). The user-developer communication process: A critical case study. *Information Systems Journal, 13*(1), 37-68.

Garceau, L. R., Jancura, E. G., & Kneiss, J. (1993). Object-oriented analysis and design: A new approach to systems development. *Journal of Systems Management*, *44*(1), 25-25.

García-Duque, J., López-Nores, M., Pazos-Arias, J. J., Fernández-Vilas, A., Diaz-Redondo, R. P., Gil-Solla, A., Ramos-Cabrer, M., & Blanco-Fernandez, Y. (2006). Guidelines for the incremental identification of aspects in requirements specifications. *Requirements Engineering Journal*, 11(4), 239-263.

Gemino, A., & Parker, D. (2009). Use case diagrams in support of use case modeling: Deriving understanding from the picture. *Journal of Database Management*, *20*(1), 1-24.

Gill, T. G. (1995). High-tech hidebound: Case studies of information technologies that inhibited organizational learning. *Accounting, Management, and Information Technology*, *5*(1), 41-60.

Glass, R. L. (2008). Divided by a common language: The story of requirements triage. *Information Systems Management*, 25(2), 190.

Goodhue, D. L., & Thompson, R. L. (1995). Task-technology fit and individual performance. *MIS Quarterly, 19*(2), 213-236.

Guimaraes, T., Staples, D. S., & McKeen, J. D. (2003). Empirically testing some main user-related factors for systems development quality. *The Quality Management Journal, 10*(4), 39-54.

Habermas, J. (1984). The theory of communicative action (T. McCarthy, Trans., Vol. 1). Boston, Massachusetts: Beacon Press.

Habermas, J. (1987). The theory of communicative action. (T. McCarthy, Trans.). Volume 2: Lifeworld and system, a critique of functionalist reason. Boston, Massachusetts: Beacon Press.

Hagström, L., Ritzen, S., & Johansson, J. (2006). The use and implementation of CAD in the Swedish furniture industry. *Forest Products Journal*, *56*(1), 73-81.

Haley, C., Laney, R., Moffett, J., & Nuseibeh, B. (2008). Security requirements engineering: A framework for representation and analysis. *IEEE Transactions on Software Engineering*, *34*(1), 133-153.

Hansen, M. (1999). The search-transfer problem: The role of weak ties in sharing knowledge across organization subunits. *Administrative Science Quarterly*, *44*(1), 82-111.

Hartwick, J., & Barki, H. (1994). Explaining the role of user participation in information system use. *Management Science, 40*(4), 440-465.

Hartwick, J., & Barki, H. (2001). Communication as a dimension of user participation. *IEEE Transactions on Professional Communication*, *44*(1), 21-36.

Haumer, P., Pohl, K., & Weidenhaupt, K. (1998). Requirements elicitation and validation with real world scenes. *IEEE Transactions on Software Engineering*, *24*(12), 1036-1054.

Hevner, A. R., & Mills. H. D. (1995). Box-structured requirements determination methods. *Decision Support Systems*, *13*(3-4), 223-239.

Hickey, A. M., & Davis, A. M. (2004). A unified model of requirements elicitation. *Journal of Management Information Systems*, *20*(4), 65-65-84.

Hirschheim, R., & Klein, H. K. (1989). Four paradigms of information systems development. *Communications of the ACM, 32*(10), 1199-1216.

Hong, D., Chiu, D. K. W., Shen, V. Y., Cheung, S. C., & Kafeza, E. (2007). Ubiquitous enterprise service adaptations based on contextual user behavior. *Information Systems Frontiers*, *9*(4), 343-358.

Hostick, C. J., Billo, R. E., & Rucker, R. H. (1991). Making the most of structured analysis in manufacturing information system design: Application of icons and cycle-time. *Computers in Industry*, *16*(3), 267-267.

Huber, G. P. (1991). Organizational learning: The contributing processes and the literatures. *Organization Science, 2*(1), 88-115.

Jarke, M., Bui, X. T., & Carroll, J. M. (1998). Scenario management: An interdisciplinary approach. *Requirements Engineering Journal, 3*(3-4), 155-173.

Jirotka, M., & Luff, P. (2006). Supporting requirements with video-based analysis. *IEEE Software*, *23*(3), 42-44.

Jones, R. M., Candy, L., & Edmonds, E. A. (1993). Knowledge-based system requirements. *Knowledge-Based Systems, 6*(1), 31-37.

Joshi, K. D., Sarker, S., & Sarker, S. (2007). Knowledge transfer within information systems development teams: Examining the role of knowledge source attributes. *Decision Support Systems*, *43*(2), 322-335.

Jwo, J.-S., & Cheng, Y. C. (2010). Pseudo software: A mediating instrument for modeling software requirements. *The Journal of Systems and Software*, *83*(4), 599.

Kaiser, K., & Srinivasan, A. (1982). User-analyst differences: An empirical investigation of attitudes related to systems development. *Academy of Management Journal, 25*(3), 630-630.

Kaiya, H., Shinbara, D., Kawano, J., & Motoshi, S. (2005). Improving the detection of requirements discordances among stakeholders. *Requirements Engineering Journal, 10*(4), 289-303.

Kazman, R., & Hong-Mei, C. (2005). From requirements negotiation to software architecture decisions. *Information and Software Technology*, *47*(8), 511-520.

Keil, M., & Carmel, E. (1995). Customer-developer links in software development. *Communications of the ACM*, *38*(5), 33-44.

Kensing, F., & Munk-Madsen, A. (1993). PD: Structure in the toolbox. *Communications of the ACM, 36*(6), 78-78.

Kim, J., Kim, J., Park, S., & Sugumaran, V. (2004). A multi-view approach for requirements analysis using goal and scenario. *Industrial Management + Data Systems*, *104*(8/9), 702-711.

Kirsch, L. J., & Beath, C. M. (1996). The enactments and consequences of token, shared, and compliant participation in information systems development. *Accounting, Management and Information Technologies*, *6*(4), 221-254.

Ko, D.G., Kirsch, L. J., & King, W. R. (2005). Antecedents of knowledge transfer from consultants to clients in enterprise system implementations. *MIS Quarterly*, *29*(1), 59-85

Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. *Communications of the ACM*, *38*(3), 69-81.

Kujala, S. (2003). User involvement: A review of the benefits and challenges. *Behaviour & Information Technology*, *22*(1), 1-16.

Kwan, I., Schroter, A., & Damian, D. (2011). Does socio-technical congruence have an effect on software build success? A study of coordination in a software project. *IEEE Transactions on Software Engineering*, *37*(3), 307-324.

Laporti, V., Borges, M. R. S., & Braganholo, V. (2009). Athena: A collaborative approach to requirements elicitation. *Computers in Industry*, *60*(6), 367-380.

Lee, A. S. (1989). A scientific methodology for MIS case studies. *MIS Quarterly*, *13*(1), 33-50.

Lee, W. J., Cha, S. D., & Kwon, Y. R. (1998). Integration and analysis of use cases using modular petri nets in requirements engineering. *IEEE Transactions on Software Engineering*, *24*(12), 1115-1130.

Leifera, R., Leeb, S., & Durgeea, J. (1994). Deep structures: Real information requirements determination. *Information & Management*, *27*(5), 275-285.

Leite, J., & Freeman, P. A. (1991). Requirements validation through viewpoint resolution. *IEEE Transactions on Software Engineering*, *17*(12), 1253-1269.

Levina, N., & Vaast, E. (2008). Innovating or doing as told? Status differences and overlapping boundaries in offshore collaboration. *MIS Quarterly*, *32*(2), 307-332.

Lichter, H., Schneider-Hufschmidt, M., & Zullighoven, H. (1994). Prototyping in industrial software projects—bridging the gap between theory and practice. *IEEE Transactions on Software Engineering*, *20*(11), 825-832.

Liker, J., Fleischer, M., Nagamachi, M., & Zonnevylle, M. (1992). Designers and their machines: CAD use and support in the US and Japan. *Communications of the ACM*, *35*(2), 77-95.

Lind, M. R., & Zmud, R. W. (1991). The influence of a convergence in understanding between technology providers and users on information technology innovativeness. *Organization Science*, *2*(2), 195-217.

Lussier, S. (2004). New tricks: How open source changed the way my team works. *IEEE Software*, *21*(1), 68-72.

Maiden, N. A. M., & Hare, M. (1998). Problem domain categories in requirements engineering. *International Journal of Human-Computer Studies, 49*(3), 281-304.

Majchrzak, A., Beath, C. M., & Lim, R. A. (2005). Managing client dialogues during information systems design to facilitate client learning. *MIS Quarterly*, *29*(4), 653-672.

Marakas, G. M., & Elam, J. J. (1998). Semantic structuring in analyst acquisition and representation of facts in requirements analysis. *Information Systems Research*, *9*(1), 37-63.

Markus, M. L. (2004). Technochange management: Using IT to drive organizational change. *Journal of Information Technology*, *19*, 4-20.

Markus, M. L., & Mao, J.-Y. (2004). Participation in development and implementation—updating an old, tired concept for today's IS contexts. *Journal of the AIS*, *5*(11/12), 514-544.

Markus, M. L., & Benjamin, R. I. (1996). Change agentry—the next IS frontier. *MIS Quarterly*, *20*(4), 385-407.

Martin, M. P. (1995). The case against CASE. *Journal of Systems Management*, *46*(1), 54-54.

Mathiassen, L., Saarinen, T., Tuunanen, T., & Rossi, M. (2007). A contingency model for requirements development. *Journal of the AIS, 8*(11), 569-597.

McKeen, J. D., & Guimaraes, T. (1997). Successful strategies for user participation in systems development. *Journal of Management Information Systems*, *14*(2), 133-150.

McKeen, J. D., Guimaraes, T., & Wetherbe, J. C. (1994). The relationship between user participation and user satisfaction. *MIS Quarterly*, *18*(4), 427-427.

Moody, D. (2009). The "physics" of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering*, *35*(6), 756-779.

Moody, J. W., Blanton, J. E., & Cheney, P. H. (1998). A theoretically grounded approach to assist memory recall during information requirements determination. *Journal of Management Information Systems*, *15*(1), 79-98.

Moynihan, T. (1996). An experimental comparison of object-orientation and functional-decomposition as paradigms for communicating system functionality to users. *The Journal of Systems and Software*, *33*(2), 163-163.

Neill, C. J., & Laplante, P. A. (2003). Requirements engineering: The state of the practice. *IEEE Software*, *20*(6), 40-45.

Nelson, K. M., & Cooprider, J. G. (1996). The contribution of shared knowledge to IS group performance. *MIS Quarterly*, *20*(4), 409-432.

Nelson, R. (2007). IT project management: Infamous failures, classic mistakes, and best practices. *MISQ Exec*, *6*(2), 67-78.

Newman, M., & Noble, F. (1990). User involvement as an interaction process: A case study. *Information Systems Research, 1*(1), 89-113.

Nichols, M. L. (1981). A behavioral analysis for planning MIS implementation. *MIS Quarterly*, *5*(1), 57-66.

Nordberg Iii, M. E. (2003). Managing code ownership. *IEEE Software, 20*(2), 26-33.

Nuseibeh, B., Kramer, J., & Finkelstein, A. (1994). A framework for expressing the relationships between multiple views in requirements specification. *IEEE Transactions Software Engineering*, *20*(10), 760-773.

Orlikowski, W. J., & Gash, D. C. (1994). Technological frames: Making sense of information technology in organizations. *ACM Transactions on Information Systems*, *12*(2), 174-207.

Pai, J.-C., & Yeh, C.-H. (2008). Factors affecting the implementation of e-business strategies: An empirical study in Taiwan. *Management Decision*, *46*(5), 681-690.

Patnayakuni, R., Ruppel, C., & & Rai, A. (2006). Managing the complementarity of knowledge integration and process formalization for systems development performance. *Journal of the AIS*, *7*(1), 545-567.

Patton, M. Q. (2002). *Qualitative research and evaluation methods* (3rd Ed.). Thousand Oaks, CA: Sage Publications.

Pitts, M.G., & Browne, G. J. (2004). Stopping behavior of systems analysts during information requirements elicitation. *Journal of Management Information Systems*, *21*(1), 203-226.

Pitts, M. G., & Browne, G. J. (2007). Improving requirements elicitation: An empirical investigation of procedural prompts. *Information Systems Journal*, *17*, 89-110.

Potts, C., Takahashi, K., & Antón, A. I. (1994). Inquiry-based requirements analysis. *IEEE Software*, *11*(2), 21-32.

Qurban, M. H., & Austria, R. D. (2009). Improving the communication skills of is developers during requirements elicitation using experiential learning. *Journal of Information Systems Education*, *20*(3), 301-311.

Ramos, I., Berry, D. M., & Carvalho, J. A. (2005). Requirements engineering for organizational transformation. *Information and Software Technology*, *47*(7), 479-495.

Ravid, A., & Berry, D. M. (2000). A method for extracting and stating software requirements that a user interface prototype contains. *Requirements Engineering Journal*, *5*(4), 225-241.

Rexfelt, O. (2009). Consumer acceptance of product-service systems. *Journal of Manufacturing Technology Management*, *20*(5), 674-699.

Robertson, S., & Robertson, J. (2006). *Mastering the requirements process.* Upper Saddle River, NJ: Addison-Wesley.

Saiedian, H., Kumarakulasingam, P., & Anan, M. (2005). Scenario-based requirements analysis techniques for real-time software systems: A comparative evaluation. *Requirements Engineering Journal*, *10*(1), 22-33.

Salaway, G. (1987). An organizational learning approach to information systems development. *MIS Quarterly*, *11*(2), 245-264.

Saleem, N., Jones, D. R., & Moses, B. (2006). Forming design teams to develop healthcare information systems. *Hospital Topics*, *84*(1), 22-30.

Sardinha, J., Choren, R., da Silva, , V., Milidiú, R. & de Lucena, C. (2006). A combined specification language and development framework for agent-based application engineering. *The Journal of Systems and Software*, *79*(11), 1565-1577.

Sawyer, S., & Guinan, P. J. (1998). Software development: Processes and performance. *IBM Systems Journal,* *37*(4), 552-569.

Shen, V. (1989). Faults, failures, and a metrics revolution. *IEEE Software,6*(2), 85, 91.

Shin, J. E., Sutcliffe, A. G., & Gregoriades, A. (2005). Scenario advisor tool for requirements engineering. *Requirements Engineering Journal*, *10*(2), 132-145.

Shirani, A. I. (2006). Sampling and pooling of decision-relevant information: Comparing the efficiency of face-to-face and GSS supported groups. *Information & Management*, *43*(4), 521-529.

Siau, K., & Tan, X. (2006). Using cognitive mapping techniques to supplement UML and up in information requirements determination. *The Journal of Computer Information Systems*, *46*(5), 59-66.

Siau, K., & Tan, X. (2008). Use of cognitive mapping techniques in information systems development. *The Journal of Computer Information Systems*, *48*(4), 49-49-57.

Sinkula, J. M., Baker, W. E., & Noordewier, T. (1997). A framework for market-based organizational learning: Linking values, knowledge, and behavior. *Journal of the Academy of Marketing Science*, *25*(4), 305-318.

Soh, C., Kien, S., & Tay-Yap, J. (2000). Enterprise resource planning: Cultural fits and misfits: Is ERP a universal solution? *Communications of the ACM*, *43*(4), 47-51.

Some, S. S. (2006). Supporting use case based requirements engineering. *Information and Software Technology*, *48*(1), 43-58.

Spinellis, D. (2009). Start with the most difficult part. *IEEE Software*, *26*(2), 70-70-71.

Stallinger, F., & Grunbacher, P. (2001). System dynamics modelling and simulation of collaborative requirements engineering. *Journal of Systems and Software*, *59*(3), 311-321.

Standish Group. (2009). CHAOS reports on IT project failures. Retrieved February 12, 2010, from http://www1.standishgroup.com/newsroom/chaos_2009.php

Stein, E. W., & Vandenbosch, B. (1996). Organizational learning during advanced system development: Opportunities and obstacles. *Journal of Management Information Systems*, *13*(2), 115-136.

Stingel, J. D., & Componation, P. J. (2006). The utilization of modeling and simulation as a supply chain management tool for a recapitalization program. *Engineering Management Journal*, *18*(2), 44-50.

Stobart, S. C., Thompson, J. B., & Smith, P. (1991). Use, problems, benefits and future direction of computer-aided software engineering in united kingdom. *Information and Software Technology*, *33*(9), 629-629.

Sustar, H., Pfeil, U., & Zaphiris, P. (2008). Requirements elicitation with and for older adults. *IEEE Software*, *25*(3), 16-17.

Szajna, B., & Scamell, R. W. (1993). The effects of information system user expectations on their performance and perceptions. *MIS Quarterly*, *17*(4), 493-493.

Telem, M. (1988). Information requirements specification I: Brainstorming a collective decision-making approach. *Information Processing & Management*, *24*(5), 549-557.

Thew, S., Sutcliffe, A., Procter, R., de Bruijn, O., McNaught, J., Venters, C. C., & Buchan, I. (2009). Requirements engineering for e-science: Experiences in epidemiology. *IEEE Software*, *26*(1), 80-87.

Topi, H., & Ramesh, V. (2002). Human factors research on data modeling: A review of prior research, an extended framework and future research directions. *Journal of Database Management*, *13*(2), 3-19.

Toth, K. (2006). Experiences with open source software engineering tools. *IEEE Software*, *23*(6), 44-52.

Tucker, A. L., Edmondson, A. C., & Spear, S. (2002). When problem solving prevents organizational learning. *Journal of Organizational Change Management*, *15*(2), 122-137.

Ullah, A., & Lai, R. (2011). Modeling business goal for business/IT alignment using requirements engineering. *The Journal of Computer Information Systems*, *51*(3), 21-28.

van Lamsweerde, A., & Letier, E. (2000). Handling obstacles in goal-oriented requirements engineering. *IEEE Transactions on Software Engineering*, *26*(10), 978-1005.

Vessey, I., & Sravanapudi, A. P. (1995). CASE tools as collaborative support technologies. *Communications of the ACM, 38*(1), 83-83.

Viega, J., Kohno, T., & Potter, B. (2001). Trust (and mistrust) in secure applications. *Communications of the ACM*, *44*(2), 31-36.

Vitalari, N. P. (1985). Knowledge as a basis for expertise in systems analysis: An empirical study. *MIS Quarterly, 9*(3), 221-241.

Vitalari, N. P., & Dickson, G. W. (1983). Problem solving for effective systems analysis: An experimental exploration. *Communications of the ACM*, *26*(11), 948-956.

Walz, D. B., Elam, J.J., & Curtis, B. (1993). Inside a software design team: Knowledge acquisition, sharing, and integration. *Communications of the ACM, 36*(10), 63-77.

Watson, H. J., & Frolick, M. N. (1993). Determining information requirements for an EIS. *MIS Quarterly, 17*(3), 255-269.

Weick, K. E., & Ashford, S. J. (2001). Learning in organizations. In F. M. Jablin and L. L. Putnam, *The new handbook of organizational communication: Advances in theory, research, and methods* (pp. 704-731).Thousand Oaks, CA: Sage.

Wetherbe, J. C. (1991). Executive information requirements—getting it right. *MIS Quarterly*, *15*(1), 51-65.

Wolf, T., Schroter, A., Damian, D., Panjer, L. D., & Nguyen, T. H. D. (2009). Mining task-based social networks to explore collaboration in software teams. *IEEE Software*, *26*(1), 58-66.

Wu, I.-L., & Hung, C.-Y. (2009). A strategy-based process for effectively determining system requirements in ecrm development. *Information and Software Technology*, *51*(9), 1308-1318.

Wu, I.-L., & Shen, Y.-C. (2006). A model for exploring the impact of purchasing strategies on user requirements determination of E-SRM. *Information and Management, 43*(4), 411-422.

Yan, W., Chen, C.-H., & Khoo, L. P. (2005). Creams: A customer requirements elicitation and management system for product conceptualization. *International Journal of Product Development*, *2*(4), 303-320.

Yin, R. (2003). *Case study research: Design and methods* (3rd Ed.). Thousand Oaks, California: Sage Publications.

Zimmermann, T., Premraj, R., Bettenburg, N., Just, S., Schroter, A., & Weiss, C. (2010). What makes a good bug report? *IEEE Transactions on Software Engineering*, *36*(5), 618-643.

Appendices

Appendix A. Literature Review Identification and Selection Process

Our objective of the review outlined in Table A-1 was to perform a broad search of the literature which examined IS analysts' communication activities with organizational stakeholders. To do so, we searched the ABI/Inform online database by using the terms "developer communication" or "analyst communication" in all search fields and in each text. We also searched the database by using the terms "requirements determination" or "requirements elicitation" from May 15 2004 onward, and drew on the requirements determination literature review of Mathiassen et al. (2007), which covered the period from 1980 to May 15, 2004. Our search was conducted in September 2011. From Mathiassen et al. (2007) we only selected articles that focused on requirements discovery and experimentation techniques because the former relies "on personal contact between developers and potential users" and the latter involves end users as part of the communication (p. 576). In other words, we excluded from the Mathiassen et al. (2007) and IRD literature reviews those articles addressing the formal process of graphically or textually specifying requirements, as well as the articles that provided techniques for prioritizing requirements because these articles addressed how requirements are structured or represented without any reference to the communication process between analysts and project stakeholders.

As a result, 32 articles that did not focus on IS analysts' communication activities to acquire information about system needs were excluded from the analysis. These included editorials, articles investigating communication regarding new financial product development, development team communication, literature reviews, and curriculum development articles.

Finally, relevant articles not captured in our earlier search were added based on articles identified by the authors of the selected papers, as well as a forward and backward search of the literature. This yielded a total of 90 papers (listed in Table A-2) as being relevant to the present study.

Table A-1. Steps Performed in Identifying the Relevant Literature			
Selection step	**Analyst communication**	**Requirements determination/elicitation**	**Total number of articles**
Identification of articles based on search terms	"developer communication" or "analyst communication" N = 41 articles	"requirements determination" or "requirements elicitation" From May 15th, 2004 to September 14th, 2011 which yielded 34 articles. Two of these were excluded because they were already identified in the analyst communication literature. N = 32 articles.	73
Identification of potentially relevant articles from Mathiassen et al. (2007)		Articles that address techniques related to the discovery of requirements. N = 39 articles.	112
Exclusion of articles not relevant to the present study	Exclusion Criteria; RD/RE articles about requirements specification/documentation techniques, editorials, development team communication, non-IS articles (e.g. new product development), curriculum development articles, and literature reviews of IS research (e.g. literature review of IS research published in Management Science). N = 32 excluded		80
Identification of additional relevant articles	Backward and forward search for additional relevant literature that has addressed communications between users and analysts. N = 10 articles		90

Table A-2. Full List of Selected Articles	
Category	**References**
Analyst communication, empirical	Agarwal et al. (1999), Butler and Fitzgerald (2001), Duggan and Thachenkary (2003), Duggan and Thachenkary (2004), Gemino and Parker (2009), Guimaraes et al. (2003), Kaiser and Srinivasan (1982), Licther et al. (1994), Majchrzak et al. (2005), McKeen et al. (1997), McKeen et al. (1994), Moynihan (1999), Saleem et al. (2006), Stobart et al. (1991), Szajna et al. (1993).
Analyst communication, non-empirical	Baskerville, Levine, and Pries-Heje (2006), Beecham, Hall, and Austen (2005), Davis, Fuller, and Berndt (2006), Garceau, Jancura, and Kneiss (1993), Hostick, Billo, and Rucker (1991), Kensing and Munk-Madsen (1993), Moody (2009), Thew et al. (2009), Topi and Ramesh (2002), Banker and Kauffman (2004).
Requirements determination / elicitation, empirical	Laporti et al. (2009), Pitts and Browne (2004), Pitts and Browne (2007), Shirani (2006).
Requirements determination / elicitation, non-empirical	Damas, Lambeau, Dupont, and van Lamsweerde (2005), Decker, Ras, Rech, Jaubert, and Rieth (2007), Dieste, Juristo, and Shull (2008), Haley, Laney, Moffett, and Nuseibeh (2008), Hong, Chiu, Shen, Cheung, and Kafeza (2007), Jirotka and Luff (2006), Sustar et al. (2008), Kim, Kim, Park, and Sugumaran (2004), Ramos, Berry, and Carvalho (2005), Sardinha et al. (2006), Siau and Tan (2006), Some (2006), Sustar et al. (2008), Ullah and Lai (2011).
Mathiassen et al. (2007), empirical	Watson and Frolick (1993), Keil and Carmel (1995), Andreou (2003), Marakas and Elam (1998), Duggan (2003), Kaiya et al. (2005), Maiden and Hare (1998), Jones, Candy, and Edmonds (1993), Haumer et al. (1998), Leifera et al. (1994), Darke and Shanks (1997), Potts et al. (1994), Browne and Rogich (2001), Bostrom (1989), Arthur and Groner (2005), Saiedian et al. (2005), Shin et al. (2005), Bryant (1997), Salaway (1987), Sawyer and Guinan (1998)
Mathiassen et al. (2007), non-empirical	Byrd et al. (1992), Davis (1982), Wetherbe (1991), Hickey and Davis (2004), García-Duque et al. (2006), Kujala (2003), Stallinger and Grunbacher (2001), Telem (1988), Hevner and Mills (1995), Leite and Freeman (1991), Nuseibeh, Kramer, and Fikelstein (1994), van Lamsweerde and Letier (2000), Frolick and Robichaux (1995), Browne and Ramesh (2002), Lee, Cha, and Kwon (1998), Breitman, Leite, and Berry (2005), Jarke, Bui, and Carroll (1998), Jirotka and Luff (2006), Ravid and Berry (2000).
Additional articles identified	Agarwal and Tanniru (1990), Moody et al. (1998), Boland (1978), Nichols (1981), Lind and Zmud (1991), Gallivan and Keil (2003), Kraut and Streeter (1995), Hartwick and Barki (2001), Joshi et al. (2007), Ko et al. (2005).
Analyst communication articles that were excluded	Carte and Russell (2003), Day (2007), Hagström et al. (2006), Kwan, Schroter, and Damian (2011), Liker et al. (1992), Lussier (2004), Martin (1995), Shen (1989), Nordberg (2003), Pai and Yeh (2008), Stingel and Componation (2006), Athanassopoulou and Johne (2004), Vessey and Sravanapudi (1995), Viega, Kohno, and Potter(2001), Wolf, Schroter, Damian, Panjer, and Nguyen (2009), Zimmerman et al. (2010).
Requirements determination / elicitation articles that were excluded	Al-Salem and Abu-Samaha (2007), Cleland-Huang, Dumitru, Duan, and Castr-Herrera (2009), Farooq, Ganoe, Carroll, Councill, and Giles (2008), Jwo and Cheng (2010), Kazman and Hong-Mei (2005), Wu and Hung (2009), Wu and Shen (2006), Choudhury and Karahanna (2008), Davis et al. (2006), Duggan and Thachenkary (2004), Glass (2008), Qurban and Austria (2009), Rexfelt (2009), Siau and Tan (2008), Spinellis (2009), Toth (2006), Yan et al. (2005).

Appendix B. Semi-Structured Interview Guideline (IS Analyst)

Part I: Implementation Overview

1. Implementation overview and system objectives
   a. Why was the project initiated? What were the objectives of the implementation?

Part II: Learning Activities

2. What specific activities did you undertake to understand users' needs? ......What specific objects? ......For how long? ......From whom?
   a. What information did you need to design and develop the system?
   b. How did you assess the accuracy of your knowledge? How did you know your information was right or wrong?
   c. How were the system requirements obtained? Type of information?
   d. Can you describe to me in detail your interactions with the users to exchange knowledge and information?
   e. How did you know when to stop collecting information?
   f. Did the users' requirements change during the ISD?
   g. How were the users' requirements validated?
3. Implementation activities
   a. How were the users trained on the system? By who? What methods were used (actual use, screen shots)? How long? What was your role? How often did you meet the users? How often did you meet other group members?
   b. Did you encounter requirement changes/modifications during this phase? If so, how were those identified, and how did you deal with them?

Part III: Task Characteristics

4. Concerning the users' tasks that are supported by the system, how would you evaluate these tasks in terms of …
   a. Variability: in terms of how the tasks are routine
   b. Analyzability: There are standard procedures that guide the work and processes of the users
   c. Interdependence: the level of interdependence of these tasks with other tasks, or with other processes within the organization.
5. In your opinion, did the technology dramatically change the users business processes and the way they carry-out their tasks? (technology, new business processes, incentives)
   a. How?
   b. What was your role in these changes?

Part IV: IS development and Implementation Outcome

6. Did the system meet users' needs to perform their tasks?
7. Budget: How did the implementation fair according to the budget?
8. Time: As to the project timeline, was the system implementation on-time?
9. Use and User satisfaction:
   a. How would you characterize the current use of the system? (if known)
   b. What were the users' reactions to the system? How would you characterize the satisfaction with the system (by users, management)?
10. Project constraints:
    a. What would you say were the major obstacles or constraints that you faced during the implementation? (e.g. top management support, user participation, system / technology complexity or rigidity, dynamic business environment, resource limitations - people or time)

Part V: Personal and Contextual Information

11. Project information
    a. What was the size of the project team? From what functions?
    b. What was the duration of the development and implementation?
12. Participant:_____ Job title_____
13. Years of experience as IS analyst:
14. Formal training: Degree(s) _____   Discipline_____ Other_____
15. Previous work experience: Previous job(s)/no. of years_____
16. Organization/client information: Industry sector_____
    No. employees_____
17. Extent of experience performing such implementations_____? Experience implementing systems for this specific area/dept?

About the Authors

Shadi SHURAIDA is an instructor in the Computer Information Systems Department at Baruch College, City University of New York, and a doctoral candidate in Information Systems at HEC Montréal. His main research interests mainly focus on the implementation and impact of information systems in organizations. His research has been published or is forthcoming in the *European Journal of Information Systems, Journal of the Association for Information Systems,* and in the proceedings of the *Administrative Sciences Association of Canada (ASAC) Conference*.

Henri BARKI is Canada Research Chair in Information Technology Implementation and Management at HEC Montréal. A member of the Royal Society of Canada since 2003, he has served on the editorial boards of *MIS Quarterly*, *Data Base for Advances in Information Systems*, *Canadian Journal of Administrative Sciences*, *Gestion*, and *Management International*. His main research interests focus on the development, introduction and use of information technologies in organizations. Journals where his research has been published include *Canadian Journal of Administrative Sciences, Data Base for Advances in Information Systems, IEEE Transactions on Professional Communication, Information Systems Journal, Information Systems Research, Information & Management, INFOR, International Journal of Conflict Management, International Journal of e-Collaboration, International Journal of e-Government Research, Journal of the AIS, Journal of Information Technology, Journal of MIS, Management Science*, *MIS Quarterly, Organization Science,* and *Small Group Research.*