Knowledge Sharing and Trust in Collaborative Requirements Analysis

Authors: Luis F. Luna-Reyes, Laura J. Black, Anthony M. Cresswell, and Theresa A. Pardo

This is the peer reviewed version of the following article: [Luna-Reyes, Luis F., Laura J. Black, Anthony M. Cresswell, and Theresa A. Pardo. "Knowledge Sharing and Trust in Collaborative Requirements Analysis." [System Dynamics Review](#) 24, no. 3 (June 2008): 265–297. doi:10.1002/sdr.404.], which has been published in final form at <http://dx.doi.org/10.1002/sdr.404>. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Self-Archiving.

Luna-Reyes, Luis F., Laura J. Black, Anthony M. Cresswell, and Theresa A. Pardo. "Knowledge Sharing and Trust in Collaborative Requirements Analysis." *System Dynamics Review* 24, no. 3 (June 2008): 265–297. doi: 10.1002/sdr.404.

Made available through Montana State University's [ScholarWorks](http://scholarworks.montana.edu)
scholarworks.montana.edu

Knowledge sharing and trust in collaborative requirements analysis

Luis F. Luna-Reyes,^{a*} Laura J. Black,^b Anthony M. Cresswell^c and Theresa A. Pardo^c

Abstract

Many information technology projects fail due to problems in requirements definition. Possible leverage points in improving requirements analysis lie in collaborative processes crossing functional and organizational boundaries, in which stakeholders learn about the problem and together identify possible solution requirements. Establishing trust among parties is critical to collaborative work, particularly in the early stages of information systems projects. However, there are few guidelines on *how* to establish trust among project participants. This paper draws on empirical work from the Center for Technology in Government facilitating interagency groups and system dynamics to generate a simple model of the role of knowledge sharing in building trust during the requirements analysis phase of a complex information systems project. Analysis of the model suggests that trust can depend on the pace of knowledge sharing among participants. More broadly, this examination offers a closer look at some of the “soft” variable dynamics that play critical roles in project progress. Copyright © 2008 John Wiley & Sons, Ltd.

Introduction

Information technology (IT) projects are complex systems that incur a high level of risk (Abdel-Hamid and Madnick, 1990; DeMarco, 1982; Lyytinen and Robey, 1999; Raffo and Kellner, 2000; Sterman, 1992; Watson and Frolick, 1993; Yourdon, 2004), and requirements determination is considered the most important process in the software development life cycle but at the same time the most difficult (Davidson, 2002; Marakas and Elam, 1998). In fact, a Standish Group study found that more than half the projects perceived as failures were compromised because of shortcomings during the requirements definition stage (Stallinger and Grünbacher, 2001; Van Lamsweerde, 2000).

Defining requirements is an endeavor in which stakeholders with different skills, backgrounds, and objectives negotiate and make sense of the nature of the technology innovation in the context of their work (Byrd *et al.*, 1992; Davidson, 2002; Sambamurthy and Kirsch, 2000; Stallinger and Grünbacher, 2001). Stallinger and Grünbacher (2001), for example, establish that “requirements emerge in

^a Universidad de las Américas-Puebla, Business School, Sta. Catarina Mártir, Cholula, Puebla, Mexico 72820.

^b College of Business, Montana State University, PO Box 173040, Bozeman, MT, U.S.A.

^c Center for Technology in Government, University at Albany, 186 Wolf Road, Suite 301, Albany, NY, U.S.A.

* Correspondence to: Luis F. Luna-Reyes. E-mail: luisf.luna@udlap.mx

Received September 2007; Accepted April 2008

received an MBA from the University of Texas at Austin and her PhD from MIT's Sloan School of Management.

a process of co-operative learning in which stakeholders jointly develop a mutual understanding about the system and benefit from other stakeholders' experiences and insights. Requirements development is therefore a 'learning, rather than gathering' activity" (p. 311). Thus defining requirements for information systems is a social process that crosses the organizational boundaries of function, expertise, tools, and objectives that stakeholders' experiences represent. Collaboration is required from beginning to end in any particular IT project, but the determination of the system's requirements constitutes the project stage where collaboration is more intense and critical to the overall success of the project (Byrd *et al.*, 1992; Davidson, 2002; Jackson, 1997; Marakas and Elam, 1998; Stallinger and Grünbacher, 2001; Van Lamsweerde, 2000)—and perhaps hardest to establish, since the project may not yet have momentum, organizational acceptance, or well-established relationships among stakeholders.

Since the requirements analyses portion of IT initiatives is critical to success, and since research indicates that collaboration can strengthen both the processes and outcomes of the requirements phase, we need to sharpen collective understanding of how collaboration occurs, and what are the specific processes through which collaborative requirements analysis unfolds. Developing a meaningful theory of collaboration in information systems development requires the integration of many interdependent factors and themes, including interpersonal communication, trust, and knowledge sharing. Several significant challenges stem from this integration. Researchers consider that ongoing sense-making processes among stakeholders "can be chaotic, nonlinear, and continuous" (Davidson, 2002, p. 329). Thus in creating a robust theory of collaborative requirements analysis we must attend to the nonlinear and continuous (Davidson, 2002), iterative learning activity (Stallinger and Grünbacher, 2001) that builds trust, communication, and capacity for knowledge sharing across organizational boundaries (Byrd *et al.*, 1992; Davidson, 2002; Stallinger and Grünbacher, 2001).

This paper provides a first step in this direction by offering a theoretical model, grounded in empirical work, of how collaborative requirements analysis takes place. As suggested by the case data, collaborating in requirements definition is a process in which participants share specialized knowledge about their work, negotiating meanings of data that need to be included in system design. Moreover, trust among participants appears to be critical to succeeding in this collaborative knowledge-sharing process. This model resonates both with the literature on collaboration as a knowledge-sharing process (Bechky, 2003; Katsamakas, 2007; Levina, 2005; Sachs, 1995) and the literature relating trust and collaboration in a path-dependent reinforcing process promoting success and engagement in project development (Mattessich *et al.*, 2001; Mishra, 1996; Powell, 1996; Vangen and Huxham, 2003). This paper presents a preliminary theory that integrates these concepts by modeling the relationships among activities that create shared knowledge and the growth of trust among stakeholders involved in this knowledge-sharing process. The theory provides dynamic explanations of why some early-stage processes in

information system development are successful and why others fail. Using the simple theory emerging from this study, we discuss implications for information systems practitioners designing requirements-defining interactions, so they can increase the collaborative investment of stakeholders and the rigor of system requirements, and so increase the probability of successful development and deployment of IT.

The model presented in the paper builds on the system dynamics tradition in project management dynamics (Lyneis and Ford, 2007; Richardson and Pugh, 1981; Sterman, 2000). However, in situations of negotiation and sense making, when building trust and engagement is key for project success, project momentum may depend much more on project participants' sense of their own progress, rather than on management's allocation of resources. In an effort to improve our understanding of how to manage better the process of requirements analysis, we focus on these aspects.

Literature review

Applications of system dynamics include a long, rich history of exploring the challenges of project management, as recently summarized by Lyneis and Ford (2007). Since the foundational work of Roberts (1974) and Pugh Roberts Associates (e.g., Cooper, 1980b), analyses of system dynamics models have provided insights into basic feedback-driven behaviors resulting from the "canonical" rework cycle (first clearly articulated by Cooper, 1980b, and relied on by many others since) and managerial decision making focused on closing the gaps between project targets for cost, schedule, and quality and perceived progress toward these targets. Adapted to a variety of industries, from highway construction (Ford *et al.*, 2004) to information technology software (Abdel-Hamid and Madnick, 1991), most model representations portray central accumulations of work to do, work believed complete, undiscovered rework, and completed work (summarized by Sterman, 2000). They primarily explore actions that managers may take (hiring more staff, ordering people to work harder or longer, etc.) in order to assure that projects meet their performance metrics (e.g., Abdel-Hamid, 1989; Ford and Sterman, 1998; Graham, 2000).

Since Abdel-Hamid (1984) first focused the system dynamics approach to information technology software development, many have provided insights into how large software-intensive projects can be more astutely managed. For example, Abdel-Hamid and Madnick (1991) sought to integrate system dynamics modeling and project dynamics insights with traditional software development processes that underlie the Capability Maturity Model over the entire software life cycle. As another example, Rodrigues and Williams (1998) proposed methods by which system dynamics modeling can be integrated with principles of project management widely publicized by the Project Management Institute. Many project models, including those focused on software

management, have represented the project's staff as belonging to a single organization. Efforts to depict distinctions among project participants, when they exist, have focused primarily on skill level, particularly between long-term project participants and newly added staff (e.g., Cooper, 1994). Often, however, projects rely on different organizations with perhaps conflicting goals and work processes. Differences among project participants' goals may become less consequential in later phases of software development, in the light of well-defined development tasks in the earliest phases such as requirements analysis. Nevertheless, disparate organizational goals and work practices may pose barriers to establishing clearly articulated system requirements that can serve as the foundation to create a high-quality information system at acceptable cost in an acceptable timeframe (Byrd *et al.*, 1992).

Often during requirements analysis, neither stakeholders nor developers have a clear image of the system to be developed, or even a shared understanding of the problem that the information system is to address (Preston, 1991). Requirements are thus usefully understood as socially constructed through a continuous process of sense making and negotiation among stakeholders (Davidson, 2002; Kraut and Streeter, 1995; Levina, 2005; Levina and Vaast, 2005; Sommerville and Sawyer, 1997). Articulating system requirements often requires addressing many issues associated with knowledge sharing, collaboration, and communication problems among analysts and stakeholders, as they seek to expose tacit assumptions and reconcile differences in practices, languages, culture, or worldviews (Eriksson, 1992; Saiedian and Dale, 2000). Therefore dynamic explorations of the requirements analysis process might represent explicitly participants' knowledge of each others' reasons for project involvement and the activities that increase that knowledge, in order to facilitate collaboration and shared understanding of the work to be done.

Because the practice view of knowledge (Bourdieu, 1977; Lave, 1988) suggests that knowledge is distributed—that is, it resides partially in people's heads, but it also resides in the activities and objects that people use, create, and re-create when developing their work or during the act of “knowing”¹ or exercising their particular competence (Argyris and Schön, 1978; Bourdieu, 1977; Carlile, 2004; Cook and Brown, 1999; Henderson, 1991; Lave, 1988; Levina, 2005; Levina and Vaast, 2005; Polanyi, 1966; Suchman, 1988; Wenger, 1998)—there is a need for learning and translation among communities. In the context of requirements analysis, the social processes of sharing knowledge across boundaries are particularly relevant (Bechky, 2003; Black, 2002; Black *et al.*, 2004; Levina, 2005; Levina and Vaast, 2005; Zhang, 2003). A knowledge-sharing approach has framed efforts to develop tools and techniques to elicit users' knowledge to be formalized as a model of the system to be developed (Byrd *et al.*, 1992; Eriksson, 1992; Marakas and Elam, 1998; Saiedian and Dale, 2000; Watson and Frolick, 1993; Wetherbe, 1991). Requirements analysts have a long tradition of using different kinds of objects and models to facilitate the negotiation process (Neill and Laplante, 2003, made a good summary of current practices). Given

the widely recognized failures, however, and growing recognition that building robust requirements “is not simply one of richer notations or more ample resources” (Bannon, 1995, p. 66), we give heightened attention to the nature and quality of objects used in the interactions during requirements analysis.

Some of these objects, such as procedures and written policies, are used in the creation and re-creation of practices inside a particular community (Hildreth and Kimble, 2002; Suchman, 1988; Wenger, 1998) or across different community boundaries (Bechky, 2003; Black, 2002; Carlile, 2004; Levina, 2005; Star, 1989). To better understand and manage the knowledge-sharing and translation process, some authors have focused on “boundary objects”—tangible things that can potentially aid the transfer, translation, and transformation processes of knowledge generation and knowledge sharing across boundaries by making local knowledge “visible” to other communities (Carlile, 2002; Majchrzak *et al.*, 2004) through an “ongoing dialog about the practice” (Levina and Vaast, 2005, p. 340). Carlile (2002) proposed that three characteristics make objects effective in sharing knowledge across boundaries: boundary objects must be “representative” of the language and syntax of the different communities, “concrete” enough to depict effectively the consequences of interdependencies among the communities’ work, and “transformable” to help participants to re-create and transform their own practices in light of the interdependencies.

Additionally, trust has been shown to influence the effectiveness of information sharing and organizational learning (Dodgson, 1993; Dyer and Singh, 1998; Katsamakas, 2007; Levin *et al.*, 2002; McEvily *et al.*, 2003; Powell *et al.*, 1996; Szulanski, 2000; Uzzi, 1997), as well as in teamwork and collaboration (Jones and George, 1998; King, 2000; Kumar, 1996) and interorganizational relationships (Butler, 1999; Hart and Saunders, 1997; Sabherwal, 1999). In general, trust can be defined as “a psychological state comprising the intention to accept vulnerability based upon positive expectations of the intentions or behavior of another” (Rousseau *et al.*, 1998, p. 395). Trust includes accepting the risks associated with the type and degree of interdependence inherent in a given relationship (Sheppard and Sherman, 1998), and trusting behaviors increase one’s vulnerability to the trustee whose behavior is not under the trustor’s control (Zand, 1972). Trust can be viewed as the positive expectation that the trustee will not behave opportunistically, even if there are incentives to do so (Chiles and McMackin, 1996), and it is built on the basis of cognitive and affective components (McAllister, 1995).

In the process of analyzing requirements, then, the challenge is to understand what activities, objects, and interactions can accelerate the knowledge sharing and growth of trust so that collaborative sense making discovers and addresses effectively a proposed system’s critical issues during the requirements determination phase. Moreover, we are interested in understanding the leverage points in generating trust and shared understanding quickly enough that participants continue to invest in the project and complete the challenging tasks of requirements analysis.

Data and methods

The case

The context for building this model-based theory is a longitudinal case study of an intergovernmental effort to define the requirements for a prototype data warehouse system to be shared by several state, county, and city agencies and independent nonprofits: the Homeless Information Management System (HIMS). The project was designed to develop a new management information system to be shared by the New York State Bureau of Housing Services (BHS) and state-funded homeless shelter providers. The system was intended to assist in managing and evaluating client service programs. The project crossed levels of government, as well as organizations, because it involved state, county, and city regulatory agencies and the nonprofit and local government service providers that receive financial support and supervision from the State.

In 1999 federal, state, and local government programs for the homeless in New York State totaled approximately \$350 million, \$130 million of which was allocated to support services to clients. BHS's regulatory and administrative responsibility included writing regulations that govern the physical, financial, and program requirements for shelters, certifying shelter programs according to these requirements, and conducting periodic inspections. This responsibility covered the 80 percent of the New York State homeless population residing in New York City, Westchester County, and Suffolk County. BHS shared this regulatory and administrative role with the New York City Department of Homeless Services (DHS) for those providers that also received City funding. The overwhelming majority of service providers were nonprofit organizations, ranging from small operations serving only a few individuals or families at a time to large programs of well-established organizations like the American Red Cross. County social services agencies had similar roles *vis-à-vis* shelters and programs outside New York City.

The project was a collaborative effort involving the BHS, DHS, and about 20 representatives of the more than 100 homeless service providers, along with the Center for Technology in Government (CTG). The role of CTG in the project was twofold: supporting and facilitating collaboration among the other participants, and providing expertise and a development environment for the prototype system. Creating a prototype Homeless Information Management System (HIMS), including data from multiple existing case management and financial systems, focused attention on the need for collaboration and development of common data standards and definitions. CTG also employed tools and artifacts representing dependencies as part of their facilitation, as well as group process methods and decision-making techniques, to support requirements analysis progress. To be successful, the project required participants from BHS responsible for shelter oversight to work in a highly collaborative way with managers from the wide range of homeless shelters in New York City and Westchester and Suffolk

counties. Over a period of more than two years, the project participants were able to achieve the necessary collaboration and share and integrate their understanding of the highly detailed and complex operational procedures and goals of different organizations. The result culminated in the design and development of a successful prototype shared information system (Cresswell *et al.*, 2002).

To achieve this success, participants had to overcome significant barriers. Part of the difficulty lay in political, social, and goal-oriented distinctions among the organizations involved. The oversight agencies, the State of New York's BHS and New York City's DHS, had substantial supervisory and funding authority over the shelter providers. Thus the proposed information system posed a substantial threat to providers' autonomy or even existence, since these oversight agencies could now have access to performance data for individual providers and could then use that data to control or even eliminate programs of those very same shelter providers helping to develop the information system. Further, since providers may be in competition with each other for funding or clients, sharing information could conceivably disadvantage some of them. It was therefore necessary for providers to develop sufficient trust in the oversight agencies before they would commit to building the proposed system and providing the necessary data. Moreover, the population of providers was diverse in number of clients, client types, programming options, and management practices. They had no standardized IT platforms, data definitions, or uniform business practices. It was therefore necessary for providers to learn many operational details about each other's business processes and to agree upon data definitions that could be useful and valid *across* the provider population. The evidence of feedback and learning in these processes, and the dynamics of how collaboration and trust developed, provide the focus of the model developed.

The team involved in the development of the HIMS prototype made intensive use of artifacts such as data-flow diagrams and stakeholder charts to share knowledge across boundaries (considering these objects as boundary objects (Carlile, 2002; Henderson, 1991; Levina and Vaast, 2005; Star and Griesemer, 1989) is an *a posteriori* interpretation that was not part of the original development plan). Data definitions, business rules, data dictionaries from the providers, and other objects used in the definition of the requirements were gradually transformed into a model for the data structure of the warehouse. Another important artifact in the interaction was the information system prototype itself. Figure 1 shows a screenshot of one of the menus of the HIMS prototype. By building it, BHS representatives learned about the limitations of their assumptions concerning data quality and their dependency on providers' gathered (and not yet gathered) data to create useful information. By interacting with the prototype during the evaluation, members from the provider group were also able to articulate an assessment of the benefits of the HIMS, suggesting different ways to continue with the project, assessing project feasibility, and asserting willingness to invest their own resources to make a full system real.

Fig. 1. HIMS prototype screenshot, taken from CTG's archives

Methods

To construct a dynamic model from the case study, we conducted group modeling sessions (Andersen and Richardson, 1997; Richardson and Andersen, 1995; Vennix, 1999) with the CTG's team of experienced qualitative researchers involved in the HIMS project. The main goals of the modeling sessions were to develop a dynamic theory of collaboration on the basis of project data and ongoing analyses at CTG and, at the same time, testing system dynamics as a method to support theory building. The particular scripts and processes used in the group modeling sessions, as well as our assessment of their usefulness, have already been reported (Luna-Reyes *et al.*, 2006). These sessions started with intensive use of storytelling and descriptions of the behaviors over time of various indicators identified by the team as key factors in project development and project outcomes. In these group modeling sessions, the researchers involved with the HIMS project generated the conceptual categories of trust,

collaboration, and the project's participants' growing understanding of others' work. Then, from their experiences and notes from working with the HIMS participants, they hypothesized causal relationships among these constructs to form a preliminary substantive theory (Glaser and Strauss, 1967) of how growing trust helped collaborative requirements analysis emerge in the HIMS project.

The formal model-building process, which took place largely outside of the group sessions, nevertheless included many iterative assessment and validation elements, seeking face validity, verifying model parameters, checking for dimensional consistency, running sensitivity tests, and qualitatively assessing the model behaviors (see Sterman, 2000, pp. 859–861). Given the lack of empirically easily quantifiable behaviors over time, running experiments and returning to discuss the plausibility of model behavior with the group was particularly important for the effort. Moreover, individual interviews with HIMS project participants were conducted to discuss weaknesses and strengths of both model structure and behavior. The following section describes the HIMS model as a representation of a formal theory (Glaser and Strauss, 1967) of collaborative requirements definition conceptualized as a social knowledge-sharing and trust-facilitated process aided by boundary objects.

Model description

Model overview

The formal theory was constructed to explore the dynamics observed in the HIMS case and more generally the possible dynamics from linking collaborative project work with knowledge, trust, and facilitative design. The model centers on reinforcing dynamics: working together builds knowledge of one's own work as well as knowledge of the other's work; as one knows the other's work better, it is possible to trust the other more; and as trust builds, parties share more information, making their collaborative work more effective. A key to collaborative work in this study lies in the facilitative tools, objects, and methods used by the Center for Technology in Government researchers (Dawes *et al.*, 2004). Stakeholder charts, diagrams, and other representations generated by CTG-facilitated conversations to identify the possible problems to be resolved, as well as specific service objectives, strategic frameworks, and diagrams that depicted shared (and sometimes disparate) understanding of data flows, all made interdependencies among the parties understandable and concrete. In turn, the growing common understanding reduced fears of embarrassment and threat and so advanced identifying and correcting errors and gaps (i.e., transforming the concrete representations) in the work done together.

In the model we make several significant simplifying assumptions to aid tractability of the simulated dynamics of building knowledge, trust, and

collaboration. First, we aggregate the service-provider agencies and represent them as a single Provider. We also aggregate the two phases of development into a single phase, considering the HIMS prototype as another artifact during the specification phase. Additionally, we operationalize CTG's facilitation as parameters representing attributes of effective boundary objects (Carlile, 2002) that therefore affect the quality and effectiveness of work undertaken together by the State and Provider. Although representing facilitation as parameters of boundary-object attributes limits the model's ability to show the iterative nature of boundary objects and the possibility that CTG facilitators can themselves learn from the representations they use, the simplification permits tractable experimentation with few variables that operationalize the effects of good (or poor) facilitation and nevertheless generate multiple distinct behavior modes. The model presented here therefore offers a useful, simplified representation of the collaboration dynamics observed in the case data and explored in an extended model (Luna-Reyes, 2004) that accounted for more of the actual project's complexities. Thus we can use it to explore the range of dynamics possible when two main participants, the State and the Provider, engage in the work of developing requirements and specifications for the proposed information system, supported by facilitation using boundary objects (model equations and documentation are available as an online companion at the journal website). It consists of three main parts: a simple project model to represent the dynamics of doing work; the participants' respective accumulations of knowledge of their own and the others' work and the resulting trust and collective engagement to continue specification development; and the influences of CTG's facilitative representations used in the meetings' process and content (see Figure 2).

Fig. 2. Model overview

Fig. 3. Project work

Project work

The model sector representing project work progress is shown in Figure 3. We adapted here the rich body of system dynamics literature on dynamics of project management (Cooper, 1980a; Lyneis and Ford, 2007; Richardson and Pugh, 1981; Sterman, 2000). When the specification-development phase begins, no work has yet been done, so all tasks (the initial Project Definition) are in the Work to Do accumulation. As participants perform work—and we represent, for simplicity, that all requirements analysis and specification-discovery work occurs in meetings between the State and Provider—tasks move to the stock Work Really Done, with a probability of $1 - \text{Error Fraction}$. Tasks performed incorrectly require rework, and additional meetings by the State and the Provider identify which tasks must be redone. Hence as the State and Provider perform Work to Do, tasks enter the accumulation Undiscovered Rework with a probability of Error Fraction. Similarly, redoing problematic tasks can be completed correctly (moving from the accumulation of Known Rework to Work Really Done) or incorrectly (returning to Undiscovered Rework), based on the error fraction.

Unlike other project management models in which managers allocate human resources to work activities, this model of collaboration represents that activity rates are influenced by the participants themselves. Specifically, Participants' Collective Engagement, or willingness to continue working on the project, depends on their sense of progress in the project work. Sense of Progress (Figure 4) relies on three main components: the perception of the fraction of tasks undertaken to that point that participants believe they have

Fig. 4. Sense of progress

completed correctly (Reported Progress on Work Done So Far, which is $[\text{Work Really Done} + \text{Undiscovered Rework} - \text{Known Rework}] / [\text{Project Definition} - \text{Work to Do}]$); the fraction of the *entire* project that participants believe has been done correctly (Perceived Progress on Overall Project, which is $[\text{Work Really Done} + \text{Undiscovered Rework} - \text{Known Rework}] / \text{Project Definition}$); and a sense of how hard the participants have been working lately (Perceived Work Rate, which is a smooth function of the actual work rates).

These progress measures reflect an optimistic view, portraying a common bias (observed in the field during this case study and in everyday life) toward believing that any work done is work done correctly, until the rework is discovered. Moreover, in addition to offering a participant-driven rather than management-allocated approach to project activities, these measures provide a more operational view of participants' assessing progress than measures based on work rates, which have often been used in project models. Both Perceived Progress on Work Done So Far and Perceived Progress on Overall Project rely on accumulations (which participants can more easily observe), rather than rates. Additionally, both variables are decreased by tasks identified as requiring rework. This formulation technically double-counts project tasks (which must exist in only one of the four accumulations of work in Figure 3 at a single point in time), but it accurately reflects the sense of discouragement and *lack* of progress that participants feel when they become aware that much of what they have done is not adequate. Therefore to omit the identified rework accumulation from the participants' sense of progress would be less correct, since it indeed affects whether participants want to continue working on the project. While Perceived Progress on Work Done So Far focuses on the tasks believed correct relative only to the number of tasks undertaken to that point, Perceived Progress on Overall Project provides a measure of the tasks believed correct relative to the entire project scope. Sense of progress combines these measures (with an explicit variable for varying the weight participants give to each) with the effect of recent work rates.

Collaboration, engagement, trust, and knowledge

Although, as an authority, BHS might have shaped the relationships according to a control and coordination structure, those at BHS in positions to exercise that authority did not do so, choosing to rely instead on collaboration based on informal norms and allowing trust to build over time. In the model, collaboration is represented by Collective Engagement, the sum of participants' respective engagement. As participants are more engaged, their productivity increases. A party's engagement in the project depends on its sense of project progress and on its trust in the other party (modeled as a weighted average of both factors; $\text{Participant's engagement} = \text{Participant's trust} \times \text{weight on trust} + \text{sense of progress} \times [1 - \text{weight on trust}]$). While the literature on trust frequently focuses on how trust influences information sharing, based on the case data, our understanding of reinforcing processes, and the cognitive basis of trust reported in the literature (McAllister, 1995), we hypothesized that trust also depends on how much a party knows about the other participant's roles, needs, objectives, and constraints. Thus this model theory depicts that trust initially depends on one's accumulation of knowledge of the other's work relating to the project and its implementation; as knowledge of the other's involvement in the project increases, one becomes capable of trusting the other more. As participants do the project work, they learn more about the range of possibilities for their own involvement, as well as more about the other's involvement in the project, which in turn influences trust.

Therefore the model portrays that the State and Provider can each possess two kinds of knowledge, which grow as they interactively do the work of creating information system requirements and specifications. Each party in the interaction can accumulate knowledge of its own roles, needs, objectives, and constraints in the project (State's Knowledge of State's Role in the Project, and Provider's Knowledge of Provider's Role in the Project). Each can also accumulate understanding of the other's roles, objectives, constraints, and needs in the project (State's Knowledge of Provider's Role in the Project, and Provider's Knowledge of State's Role in the Project). In the model, these stocks of knowledge are dimensionless variables (defined over [0,1]) and increase as people learn more about the project and how it can benefit (and perhaps hinder) their work (see Figure 5). The model represents that, as knowledge of one's own work in the project increases, the probability of error in the project work decreases. When both participants possess a lot of knowledge about their respective roles in the project, the probability of making mistakes as they work together to develop system requirements is lower.

CTG facilitative methods and tools

The CTG facilitators' work is operationalized through parameters representing the effectiveness of cross-boundary communication objects and tools: Concreteness, Transformability, and Potential Accuracy in Representing State's

Fig. 5. Knowledge, engagement, and trust

(Provider's) Point of View (all defined on the interval $[0,1]$). Concreteness (Carlile, 2002) refers to the specificity of the representations in cross-boundary work on requirements discovery. Through visible-to-all diagrams and note-taking, CTG's facilitative team helped direct the State and Provider in listing specific stakeholders, for example, or in identifying particular issues in sharing certain kinds of data. In the model, the ability to do new work and to identify problems in work done to date depends on the Concreteness of the content. Transformability (Carlile, 2002) refers to the ability of the State and Provider to address and correct any problems identified in the concrete representations. If participants in the project work were unable to voice their opinions about how

Fig. 6. Accuracy in representing a participant's point of view

the work done to date should be different, or if their suggestions for resolving problems were ignored by the other party and the facilitators, then the work would be “untransformable” to them (in this case Transformability would have a very low value). CTG’s facilitative methods built on a philosophy of helping parties be explicit about their concerns and hopes for joint work, and therefore high CTG involvement in a project suggests that the parameters Concreteness and Transformability take on high values.

CTG staff had no authority (or desire) to coerce parties to share information. Therefore, CTG involvement affected the Potential Accuracy for representing each party’s point of view, but the actual accuracy depended also on each party’s trust in the other (see Figure 6). As a participant grows in trust of the other, he is likely to share more information, increasing the accuracy of the representation of his point of view in the project work. As the accuracy of one participant’s point of view increases, the other grows in knowledge of that participant’s role in the project. In the model we represent, too, that the CTG staff, through preliminary phone calls and meetings with each party in preparing for the project, have the ability to inform each party of the other’s interest in collaborating and so establish in each an initial modicum of trust in the other (i.e., CTG’s involvement creates a non-zero initial value for the stocks State’s Knowledge of Provider’s Role in the Project and Provider’s Knowledge of State’s Role in the Project).

Fig. 7. Main reinforcing processes in the model

Summary of model description

The primary feedback loops formed by the connections described above are reinforcing. The causal loop diagram in Figure 7 depicts that the CTG research staff used carefully chosen facilitative methods and tools to increase the clarity and usefulness of communication across the BHS–Provider boundary. As participants began to work together and make progress, they became more engaged (R1 and R2) to do the work and in turn learned more about their roles, constraints, objectives, and needs in continuing participation in the project and its implementation (R3, through Provider's Knowledge about Provider's Work, and R4, through State's Knowledge about State's Work). As participants' knowledge of their own work increased, the error rate decreased, and as their knowledge of the other's work increased, the productivity rate (not shown in the causal loop diagram, for the sake of visual simplification) increased. By engaging in work together—i.e., collaboratively—they also learned about the other's role in the project and so grew in trust (R5 and R6). As trust increased, each party shared more information, thus clarifying its point of view for the other

and so allowing the other to learn more about its objectives and constraints, and thereby trusting more (R7). “Collaboratively Doing Work”, at the center of Figure 7, is not a single variable in the model; rather, it represents at an abstract level the effects of collaboration operationalized in the model through the participants’ decreased error rate and increased productivity rate, as they affect each of the “doing work” flows because of the changes in knowledge accumulations (see flows in Figure 3).

As any other reinforcing process, the feedback loops portrayed in Figure 7 can also work in vicious cycles; when participants do not perceive progress from their work together, they become less engaged and do not learn more about their own work and others’; trust does not increase, and individuals do not share additional information, and learning about one another’s work slows or ceases, further slowing or even halting the project work.

Model simulations

HIMS simulation

Although a simplified representation of the actual interactions observed during the HIMS project work, model experiments help identify points of leverage in several distinct behavior modes of collaboration and non-collaboration, requirements analysis progress, and requirements analysis failure. The base scenario simulated with the model represents the requirements analysis and specification stage of the HIMS prototype development. The initial parameter values were based on the qualitative data available in the case. Given that the CTG team sustained previous iterative interactions with the State team in preparation for this IT initiative, the State’s knowledge about its own objectives and role in the project is set to 0.8 initially, and the State’s knowledge about the Provider’s work is set to 0.3. On the other hand, both accumulations of the Provider’s knowledge were initially low (0.1 in both kinds of knowledge). Additionally, CTG participation is represented by the use of effective objects and methods during the process (0.8 in Concreteness, Transformability, and Potential Accuracy). As shown in Figure 8, project work is completed in about 42 months in this base scenario, which is longer than observed in the case. (More detailed models produced by the authors reproduce with more precision the case data, but the dynamics of this parsimonious representation are similar.)

In the base simulation, during the early stages of the project Work Really Done and Undiscovered Rework grow together. Because the Provider’s low initial knowledge creates a high error rate, the accumulation of Undiscovered Rework is greater than the accumulation of Work Really Done during these first months. The Provider’s knowledge of its own role and objectives in the project increases as the collaborative work takes place (see Figure 9(a)), however,

Fig. 8. Project work development in the HIMS case

gradually reducing the error rate and increasing the fraction of work done correctly (reinforcing process R3 described in the previous section). Undiscovered Rework is identified through the reviewing activities (Recognizing Problems), which begin when the participants in the project perceive they have undertaken about 20 percent of the project, and reviewing activities increase gradually towards the end of the project.

Because of the effectiveness of the facilitation methods reflected in the values of Concreteness, Transformability, and Potential Accuracy, the four accumulations of knowledge move from their initial conditions to values close to the maximum (Figure 9(a)). In this base scenario, all the reinforcing processes depicted in the previous section work in a virtuous manner to promote less error, more engagement, and growing knowledge and trust during the project. Although at the end of the simulation all four levels of knowledge have a value close to the maximum, participants end the project with knowledge about themselves slightly higher than knowledge about their partners.

Figure 9(b) shows the dynamics of trust and engagement in the project. The State's and Provider's trajectory of growing trust is the same qualitatively and quantitatively as their respective knowledge about the other's role and objectives in the project (while simplistic, it helps underscore our hypothesis that trust is driven by, as well as drives, sharing knowledge, and it helps distinguish among the four accumulations of knowledge in the simulations). Collaborative work is explained partially by these two trajectories of trust, but it is influenced, too, by participants' collective engagement in the project. The collective

Fig. 9. (a) Knowledge levels and (b) trust and collective engagement in the HIMS scenario

(a)

(b)

engagement is higher than the level of trust while project work is being developed, and decreases to the same value of trust at the end of this project stage because engagement falls when there is no more work to do. We interpret this final value of collective engagement as momentum to be carried to the next stages of the project or to different initiatives started by the same partnership, or as ongoing monitoring of or communication about the HIMS project among the participants.

Under base scenario initial conditions, the model produces dynamic behaviors consistent with the substantive theory and actual behaviors and outcomes observed in the case study project. The simulated scenarios below explore the usefulness of the formal theory represented in the model constructs and relationships by analyzing the behaviors resulting from parameters and initial conditions different from those of the base case. In order to explore the effectiveness of the facilitation design in the promotion of knowledge sharing, trust, and engagement in the context of a project, two experiments are shown. In the first, effective facilitation objects were used in scenarios with symmetric but different levels of initial knowledge. That is, high values in Concreteness, Transformability and Potential Accuracy were used in simulations where:

- both participants in the project begin with high knowledge about their own role in the project but low knowledge about the other participant (generating initially low trust);
- both participants start with low knowledge about their own role but high knowledge about the other's (generating initially high trust); and
- both participants begin with low levels in both accumulations of knowledge.

In the second experiment, average, rather than excellent, facilitation artifacts were tested in the same three scenarios.

Good facilitation design with several initial knowledge conditions

Figure 10(a) shows the completion of tasks really done in the three scenarios, comparing them to the base case (the HIMS case), in which high-quality facilitation objects are used. In the cases where the two participants started with a high level of any kind of knowledge, the project work finishes earlier than in the base case, and when both levels of knowledge start low, the work gets done a little later than in the base case. Moreover, an interesting pattern is observed in the cases in which both participants start with one kind of knowledge. In the scenario in which there is a high level of knowledge about their *own* roles and objectives, the trajectories initially climb faster than in the scenario in which there is a high level of knowledge about the *other's* role and objectives, but in the latter scenario the project ends earlier. The initial knowledge

Fig. 10. Work Really Done and Collective Engagement with several initial knowledge conditions for a good facilitation scenario

of participants' own purposes drives a lower error rate and a faster start-up, but high initial knowledge about the other leads to higher levels of trust and engagement. Being highly engaged in the project makes the learning-by-doing reinforcing process work faster, promoting quick learning in both participants and an even more rapid reduction in the error fraction. The higher engagement also promotes more intensive interactive work (Figure 10(b)) and thus an earlier conclusion to the project.

The symmetry of the initial values of the stocks of knowledge makes the State's and Provider's knowledge behave identically in the scenarios depicted in Figure 11(a) and (b). The Provider's knowledge of its own role and objectives (Figure 11(a)) grows faster in the scenario of high trust and low self-knowledge for the same reasons described above. Interestingly, in this scenario both the Provider and the State end up knowing slightly more about themselves (Figure 11(a) and (b)). In this case, low initial values of knowledge of their own work promote higher error rates at the beginning of the project. The additional interactive meetings necessitated by recognizing and solving these additional errors promote more learning and higher levels of both kinds of knowledge (of their own and the other's work) at the end of the project.

Good facilitation artifacts prove to be a powerful influence even in the case in which four stocks of knowledge start at a low level. In the three first scenarios, the reinforcing processes in the model behave in a virtuous mode. The initial work and the initial sense of progress promote increasing engagement, and doing more interactive work drives learning and increases the four stocks of knowledge.

Average facilitation design with several initial knowledge conditions

As mentioned above, the second experiment involved the use of the same initial knowledge scenarios with average facilitation artifacts (0.6 for the initial values of Concreteness, Transformability, and Potential Accuracy). In this second set of explorations, the project is completed only in the case where project participants begin with a high level of trust (high knowledge about the other's involvement in the work) (Figure 12(a)).

In both cases in which the project is not completed, the reinforcing processes work as a trap, or in a vicious mode. Low levels of initial trust, based on knowledge of the other's work, result in low activity rates in doing work (correctly or incorrectly), as indicated by loops R5 and R6, R1 and R2 in Figure 7. Low work rates yield little sense of progress, based on low accumulations of tasks believed correctly completed, and little learning by doing and therefore very low increases in participants' knowledge of their own or the other's role in the project, as portrayed by loops R3 and R4. The lack of achievement and perpetuated lack of trust make the participants become even less engaged and spend even less effort working on the project (Figure 12(b)). Continuing low rates of doing work limit the learning of both participants, maintaining the

Fig. 11. Comparison of Provider's knowledge of its own work and the State's work on the project under several initial knowledge conditions for a good facilitation scenario

Fig. 12. Work Really Done and Collective Engagement with several initial knowledge conditions for an average facilitation scenario

four knowledge stocks at the same level during the simulation (Figure 13(a) and (b)).

This set of simulations reveals that an imperfect (but not bad) facilitation design, as operationalized through moderately effective (0.6, on a scale of 0 to 1) artifact concreteness and transformability, can contribute to both success and failure in projects and highlights the importance of trust in cross-boundary collaborative efforts. When facilitation methods are not reliably strong, project success hinges more critically on beginning with a team in which members already have significant levels of trust, or knowledge of the other's role in the project. Pre-project activities, in which participants learn about the other's needs and objectives, may also be important determinants of project success or failure. Additionally, other simulations (not shown here) indicate that demonstrating progress early and fast, by rapidly doing work early in the project to increase accumulations of tasks believed done correctly (and perhaps, not finding errors very quickly in the tasks performed) could be another way to reverse the reinforcing processes to work toward project progress and completion.

Discussion and conclusions

In inter-organizational projects necessitating software requirements analysis, it is critical to consider how different organizations establish trust in order to make progress. Building knowledge of one another's work is a necessary part of the process, and well-designed facilitation tools for representing explicitly project participants' understanding and the consequences of dependencies—i.e., for creating effective boundary objects—can help participants learn more about their own and others' work as they undertake analysis tasks. This work contributes to the system dynamics tradition examining project dynamics by

- representing explicitly different organizations working on the project, rather than a homogeneous project staff;
- focusing on the collaboration-intensive early requirements analysis phase of software development;
- focusing on how project progress affects the project participants themselves, rather than managers, and their subsequent engagement;
- representing explicitly accumulations of knowledge about the other participants' objectives and customary means of accomplishing work; and
- representing explicitly the role of facilitation and the quality of facilitative artifacts often used in early system development projects such as requirements analysis.

Together, these advance our understanding of “soft” concepts such as trust and collaboration by operationalizing their roles in a successful requirements analysis project. The formal theory portrayed here suggested that trust, rather

Fig. 13. Comparison of State's Knowledge of its own work and the Provider's work on the project under several initial knowledge conditions and average facilitation scenarios

than only fueling knowledge sharing, also grows as a result of effectively sharing knowledge across organizational boundaries (McAllister, 1995), which results from actually undertaking and making progress in project work. Because doing the work, sharing knowledge, and building trust are nested reinforcing processes, requirements analysis is more likely to produce useful, robust outcomes when participants begin with a modicum of trust. The quality of facilitation and facilitative artifacts can play a pivotal role in requirements analysis success. In situations of poor or mediocre facilitative representations, success will hinge on starting with the “right” participants—people who already know and trust each other’s work. In the absence of already-established trust, excellent boundary objects as part of facilitation can provide powerful points of leverage in the requirements analysis and specification process, making possible interactions that allow a number of trust-building mechanisms to come into play, even in an initially low-trust situation.

Robust boundary objects (Carlile, 2002; Star, 1989) used in facilitation operationally aid participants’ abilities to learn about their own and others’ work. Learning about their own work increases participants’ effectiveness by reducing errors during the analysis phase, effectively addressing functional group obstacles (Byrd *et al.*, 1992). Boundary objects’ concreteness helps address many challenges in sharing knowledge across functional and role boundaries by clarifying definitions, explicating rationales for current information processes and hopes for future processes, and (by providing visible documentation of participants’ thoughts) offering managers the chance to review and refine the emerging ideas (Eriksson, 1992). Boundary objects’ depictions of interdependencies among the participants’ work aid cross-organizational communication about differing perspectives and vocabularies, help participants to focus on the salient aspects of their interdependent work, and educate technical analysts and developers about the nature and objectives of the work to be improved by the new information system (Davidson, 2002; Eriksson, 1992; Saiedian and Dale, 2000).

The process of transforming jointly held representations of the interdependent work aids the social construction of requirements, creating collaborative momentum for parties to remain engaged and fully participating in the work (Davidson, 2002). Diagrams, workshops, and tools used in the HIMS project provided, quite literally, spaces to discuss differences in practices and to build a shared vision of services, data definitions, and initial business rules to develop the prototype. Furthermore, good boundary objects aid progress in the work itself, which also helps generate engagement and, through learning about one’s own and the other’s work, further reduces error and builds trust. Trust facilitates candid, constructive discussions of multiple concerns and differences in practices that appear as barriers to collaborative work on the information system (Eriksson, 1992; Saiedian and Dale, 2000; Stallinger and Grünbacher, 2001). Trust also leads to further engagement and sharing more information more effectively, accelerating productivity in the work. The transformability of high-quality boundary objects is key to generating this trust; as participants

observe that their input and insight change how the interdependent work is represented, they grow in the belief that they can provide more information and it will be given attention and allowed to change others' perspectives on the work. Hence transformable boundary objects can help participants increase the accuracy of information they are providing and so reduce errors in the work at hand.

Many tools and methods for structured analysis in the requirements definition phase of information system development (Sommerville and Sawyer, 1997; Van Lamsweerde, 2000) can serve as excellent boundary objects, but too often they are not used as such. When analysts do not explain the grammar or terms used in diagrams or other tools, the artifacts appear mysterious to users—not concrete, relative to the users' experiences, and certainly not transformable. Similarly, when users' or other stakeholders' comments and questions are dismissed by technical experts, the analysts miss opportunities to understand salient dependencies in the work among participants and among the users and the developers' understanding of the system to be developed. We believe practitioners can significantly improve the quality of requirements analysis by using tools and methods in this early phase as boundary objects, in order to build knowledge, trust, and collaborative engagement among all parties.

Although practitioner literature on organizational change has frequently discussed the value of “quick wins”, little theoretical argument has been made for its importance in information systems development. Through this empirical investigation, we discovered that a sense of progress was necessary for participants to remain engaged in the joint project work. Indeed, participants in the HIMS project were adamant that visible, early progress in generating agreements on data definitions played a significant role in encouraging provider participants and in their decisions to remain engaged and share additional knowledge. The role of Sense of Progress in the model presented above suggests that the pacing of project work and communicating its results to project participants (as well as to other stakeholders) merit additional empirical and theoretical attention and exploration.

From the emerging formal theory described here, we propose that effective cross-boundary knowledge sharing and perception of project progress are both sufficient and necessary conditions for collaborative requirements analysis to take place. While the theory, as represented by the model, helps identify facilitative boundary objects as a point of leverage in collaborative requirements analysis, more research on the use of tools, drawings, diagrams, and working and non-working prototypes in requirements analysis is needed. Among scholars there is ample room for additional research on the uses of specific tools and their role and the conditions under which they act as effective boundary objects in information systems development. A prototype of a system, for example, is a very concrete tool, yet it may embed assumptions about the future system that are difficult to alter, even if the feedback generated by the prototype's concreteness greatly enhances understanding of the future system's

needs and uses. Of course, much work remains to be done. This model, for example, portrays that once participants accumulate knowledge, it is not forgotten or obsolesced. Furthermore, for simplicity's sake, the model portrays trust as only increasing with participants' knowledge of the other, when in fact it is possible that trust may actually decline as knowledge of another increases, if that knowledge reveals large gaps between what the other says and what he or she does (Luna-Reyes, 2004). Therefore, as we continue this research, we want to explore ways that knowledge, trust, and trade-offs between increasing concreteness of a boundary object and its transformability interact to help or hinder collaborative work in information systems.

Note

1. According to Polanyi (1966), the most important object that people use in the acts of knowing is the human body, considering other objects extensions of the human body.

Supporting information

Supporting information may be found in the online version of this article.

Acknowledgements

The research reported here was supported by National Science Foundation grant #SES-9979839. The views and conclusions expressed in this paper are those of the authors alone and do not reflect the views or policies of the National Science Foundation. This paper is based on earlier work contributed to by David F. Andersen, Donna Canestraro, Meghan Cook, Ignacio J. Martínez-Moyano, Fiona Thompson, and George P. Richardson. An early version of this paper was presented at the 2003 Hawaii International Conference of System Sciences (HICSS-36), sponsored by the College of Business, University of Hawaii at Mānoa (January 6–9, 2003).

References

- Abdel-Hamid TK. 1984. The dynamics of software development project management: an integrative system dynamics perspective. PhD thesis, MIT Sloan School of Management: Cambridge, MA.
- . 1989. The dynamics of software project staffing: a system dynamics-based simulation approach. *IEEE Transactions on Software Engineering* **15**(2): 109–119.
- Abdel-Hamid TK, Madnick SE. 1990. The elusive silver lining: how we fail to learn from software development failures. *Sloan Management Review* **32**: 39–48.

- . 1991. *Software Project Dynamics: An Integrated Approach*. Prentice-Hall: Englewood Cliffs, NJ.
- Andersen DF, Richardson GP. 1997. Scripts for group model building. *System Dynamics Review* **13**(2): 107–129.
- Argyris C, Schön DA. 1978. *Organizational Learning: A Theory of Action Perspective*. Addison-Wesley: Reading, MA.
- Bannon L. 1995. The politics of design: representing work. *Communications of the ACM* **38**(9): 66–68.
- Bechky B. 2003. Sharing meaning across occupational communities: the transformation of understanding on a production floor. *Organization Science* **14**(3): 312–330.
- Black LJ. 2002. Collaborating across boundaries: theoretical, empirical, and simulated explorations. PhD dissertation, Sloan School of Management, MIT: Cambridge, MA.
- Black LJ, Carlile PR, Repenning NP. 2004. A dynamic theory of expertise and occupational boundaries in new technology implementation: building on Barley's study of CT scanning. *Administrative Science Quarterly* **49**(4): 572–607.
- Bourdieu P. 1977. *Outline of a Theory of Practice*. Cambridge University Press: Cambridge, U.K.
- Butler JK. 1999. Trust expectations, information sharing, climate of trust, and negotiation effectiveness and efficiency. *Group and Organization Management* **24**(2): 217–238.
- Byrd TA, Cossick KL, Zmud RW. 1992. A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly* **16**(1): 117–138.
- Carlile P. 2002. A pragmatic view of knowledge and boundaries: boundary objects in new product development. *Organization Science* **13**(4): 442–455.
- . 2004. Transferring, translating and transforming: an integrative framework for managing knowledge across boundaries. *Organization Science* **15**(5): 555–568.
- Chiles TH, McMackin JF. 1996. Integrating variable risk preferences, trust, and transaction cost economics. *Academy of Management Review* **21**(1): 73–99.
- Cook S, Brown J. 1999. Bridging epistemologies: the generative dance between organizational knowledge and organizational knowing. *Organization Science* **10**(4): 381–400.
- Cooper KG. 1980a. *Strategic Analysis for Program Management: A Computer-Based Aid for the Senior Management of Large-Scale Design and Construction Programs*. Pugh-Roberts Associates: Cambridge, MA.
- . 1980b. Naval ship production: a claim settled and a framework built. *Interfaces* **10**(6): 20–36.
- . 1994. The \$2,000 hour: how managers influence project performance through the rework cycle. *Project Management Journal* **25**(1): 11–24.
- Cresswell AM, Pardo TA, Thompson F, Canestraro DS, Cook M, Black LJ, Luna LF, Martinez IJ, Andersen DF, Richardson GP. 2002. Modeling intergovernmental collaboration: a system dynamics approach. In *Proceedings of the Hawaiian International Conference on System Sciences* 35, Hawaii.
- Davidson E. 2002. Technology frames and framing: a socio-cognitive investigation of requirements determination. *MIS Quarterly* **26**(4): 329–358.
- Dawes SS, Pardo TA, Simon S, Cresswell AM, LaVigne MF, Andersen DF, Bloniarz PA. 2004. *Making Smart IT Choices: Understanding Value and Risk in Government IT Investments*. University at Albany: Albany, NY.
- DeMarco T. 1982. *Controlling Software Projects: Management, Measurement and Estimation*. Yourdon Press: New York.

- Dodgson M. 1993. Learning, trust, and technological collaboration. *Human Relations* **46**(1): 77–95.
- Dyer J, Singh H. 1998. The relational view: cooperative strategy and sources of inter-organizational competitive advantage. *Academy of Management Review* **23**(4): 660–679.
- Eriksson H. 1992. A survey of knowledge acquisition techniques and tools and their relationship to software engineering. *Journal of Systems and Software* **19**: 97–107.
- Ford DN, Sterman JD. 1998. Dynamic modelling of product development processes. *System Dynamics Review* **14**(1): 31–68.
- Ford DN, Anderson S, Damron A, de Las Casas R, Gokmen N, Kuennen S. 2004. Managing constructability reviews to reduce highway project durations. *ASCE Journal of Construction Engineering and Management* **130**(1): 33–42.
- Glaser BG, Strauss AL. 1967. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine: Chicago, IL.
- Graham AK. 2000. Beyond pm 101: lessons for managing large development programs. *Project Management Journal* **31**(4): 7–18.
- Hart P, Saunders C. 1997. Power and trust: critical factors in the adoption and use of electronic data interchange. *Organization Science* **8**(1): 23–42.
- Henderson K. 1991. Flexible sketches and inflexible data bases: visual communication, conscription devices, and boundary objects in design engineering. *Science, Technology, and Human Values* **16**(4): 448–473.
- Hildreth P, Kimble C. 2002. The duality of knowledge. *Information Research* **8**(1): 24.
- Jackson M. 1997. The meaning of requirements. *Annals of Software Engineering* **3**: 5–21.
- Jones G, George J. 1998. The experience and evolution of trust: implications for co-operation and teamwork. *Academy of Management Review* **23**(3): 531–546.
- Katsamakas E. 2007. Knowledge processes and learning options in networks: evidence from telecommunications. *Human Systems Management* **26**(3): 181–192.
- King S. 2000. Building trust among members of a work team: one facilitator's experiences. *Group Facilitation* **2**(2): 15–21.
- Kraut RE, Streeter LA. 1995. Coordination in software development. *Communications of the ACM* **38**(3): 69–81.
- Kumar N. 1996. The power of trust in manufacturer-retailer relationships. *Harvard Business Review* **74**(6): 92–105.
- Lave J. 1988. *Cognition in Practice*. Cambridge University Press: Cambridge, U.K.
- Levin D, Cross R, Abrams L, Lesser E. 2002. *Trust and Knowledge Sharing: A Critical Combination*. IBM Institute for Knowledge-Based Organizations: Somers, NY.
- Levina N. 2005. Collaborating on multiparty information systems development projects: a collective reflection-in-action view. *Information Systems Research* **16**(2): 109–130.
- Levina N, Vaast E. 2005. Competence in practice: Implications for implementation and use of information systems. *MIS Quarterly* **29**(2): 335–363.
- Luna-Reyes LF. 2004. Collaboration, trust and knowledge sharing in information-technology-intensive projects in the public sector. PhD dissertation, School of Information Science and Policy, University at Albany: Albany, NY.
- Luna-Reyes LF, Martinez-Moyano JJ, Pardo TA, Cresswell AM, Andersen DF, Richardson GP. 2006. Anatomy of a group model-building intervention: building dynamic theory from case study research. *System Dynamics Review* **22**(4): 291–320.
- Lyneis JM, Ford DN. 2007. System dynamics applied to project management: a survey, assessment, and directions for future research. *System Dynamics Review* **23**(2/3): 157–189.

- Lyytinen K, Robey D. 1999. Learning failure in information systems development. *Information Systems Journal* **9**(2): 85–102.
- Majchrzak A, Cooper LP, Neece OA. 2004. Knowledge reuse for innovation. *Management Science* **50**(2): 174–188.
- Marakas G, Elam J. 1998. Semantic structuring in analyst acquisition and representation of facts in requirement analysis. *Information Systems Research* **9**(1): 37–63.
- Mattessich PW, Murray-Close M, Monsey BR. 2001. *Collaboration: What Makes It Work*. Amherst H. Wilder Foundation: Saint Paul, MN.
- McAllister D. 1995. Affect- and cognition-based trust as foundations for interpersonal cooperation in organizations. *Academy of Management Journal* **38**(1): 24–59.
- McEvily B, Perrone V, Zaheer A. 2003. Trust as an organizing principle. *Organization Science* **14**(1): 91–103.
- Mishra AK. 1996. Organizational responses to crisis: the centrality of trust. In *Trust in organizations: Frontiers of theory and research*, Tyler T, Kramer RM (eds). Sage: Thousand Oaks, CA; 261–287.
- Neill CJ, Laplante PA. 2003. Requirements engineering: the state of the practice. *IEEE Software* **20**(6): 40–45.
- Polanyi M. 1966. *The Tacit Dimension*. Doubleday: Garden City, NY.
- Powell W. 1996. Trust-based forms of governance. In *Trust in Organizations: Frontiers of Theory and Research*, Tyler T, Kramer RM (eds). Sage: Thousand Oaks, CA; 51–67.
- Powell W, Koput KW, Smith-Doerr L. 1996. Interorganizational collaboration and the locus of innovation: networks of learning in biotechnology. *Administrative Science Quarterly* **41**(1): 116–145.
- Preston AM. 1991. The “Problem” in and of management information systems. *Accounting, Management, and Information Technology* **1**(1): 43–69.
- Raffo DM, Kellner M. 2000. Empirical analysis in software process simulation modeling. *Journal of Systems and Software* **53**(1): 31–41.
- Richardson GP, Andersen DF. 1995. Teamwork in group model building. *System Dynamics Review* **11**(2): 113–137.
- Richardson GP, Pugh AL III. 1981. *Introduction to System Dynamics Modeling with Dynamo*. Productivity Press: Cambridge, MA. (Now available from Pegasus Communications, Waltham, MA).
- Roberts EB. 1974. A simple model of R&D project dynamics. *R&D Management* **5**(1): 1–15.
- Rodrigues AG, Williams TM. 1998. System dynamics in project management: assessing the impacts of client behaviour on project performance. *Journal of Operational Research Society* **49**(1): 2–15.
- Rousseau DM, Sitkin SB, Burt RS, Camerer C. 1998. Not so different after all: a cross-discipline view of trust. *Academy of Management Review* **23**(3): 393–404.
- Sabherwal R. 1999. The role of trust in outsourced IS development projects. *Communications of the ACM* **42**(2): 80–86.
- Sachs P. 1995. Transforming work: collaboration, learning and design. *Communications of the ACM* **38**(9): 36–44.
- Saiedian H, Dale R. 2000. Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology* **42**(6): 419–428.
- Sambamurthy V, Kirsch LJ. 2000. An integrative framework of the information systems development process. *Decision Sciences* **31**(2): 391–411.

- Sheppard BH, Sherman DM. 1998. The grammars of trust: a model and general implications. *Academy of Management Review* **23**(3): 422–437.
- Sommerville I, Sawyer P. 1997. *Requirements Engineering: A Good Practice Guide*. Wiley: Chichester.
- Stallinger F, Grünbacher P. 2001. System dynamics modelling and simulation of collaborative requirements engineering. *Journal of Systems and Software* **59**(3): 311–321.
- Star SL. 1989. The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving. In *Distributed Artificial Intelligence*, Gasser L, Huhns MN (eds). Morgan Kaufmann: San Mateo, CA; 37–54.
- Star SL, Griesemer JR. 1989. Institutional ecology, “translations” and boundary objects: amateurs and professionals in Berkeley’s Museum of Vertebrate Zoology. *Social Studies of Science* **19**(3): 387–420.
- Sterman JD. 1992. System dynamics modeling for project management. MIT: Cambridge, MA.
- . 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Irwin/McGraw-Hill: Boston, MA.
- Suchman L. 1988. Representing practice in cognitive science. *Human Studies* **11**(2/3): 305–325.
- Szulanski G. 2000. The process of knowledge transfer: a diachronic analysis of stickiness. *Organizational Behavior and Human Decision Processes* **82**(1): 9–27.
- Uzzi B. 1997. Social structure and competition in interfirm networks: the paradox of embeddedness. *Administrative Science Quarterly* **42**(1): 35–67.
- Van Lamsweerde A. 2000. Requirements engineering in the year 00: a research perspective. In *Proceedings of the 22nd International Conference in Software Engineering*, Limerick, Ireland.
- Vangen S, Huxham C. 2003. Nurturing collaborative relations: building trust in interorganizational collaboration. *Journal of Applied Behavioral Science* **39**(1): 5–31.
- Vennix JAM. 1999. Group model-building: tackling messy problems. *System Dynamics Review* **15**(4): 379–401.
- Watson HJ, Frolick MN. 1993. Determining information requirements for an EIS. *MIS Quarterly* **17**(3): 255–269.
- Wenger E. 1998. *Communities of Practice: Learning, Meaning and Identity*. Cambridge University Press: Cambridge, U.K.
- Wetherbe JC. 1991. Executive information requirements: getting it right. *MIS Quarterly* **15**(1): 51–66.
- Yourdon E. 2004. *Death March*. Pearson Education: Upper Saddle River, NJ.
- Zand DE. 1972. Trust and managerial problem solving. *Administrative Science Quarterly* **17**(2): 229–239.
- Zhang J. 2003. Cross-boundary knowledge sharing: a case study of building the multi-purpose access for customer relations and operational support (macros) system. PhD dissertation, School of Information Science and Policy, University at Albany: Albany, NY.