

Conceptualizing Interpersonal Relationships in Agile IS Development

Madsen, Sabine; Matook, Sabine

Publication date:
2010

Document Version
Peer reviewed version

Citation for published version (APA):
Madsen, S., & Matook, S. (2010). *Conceptualizing Interpersonal Relationships in Agile IS Development*. Paper presented at International Conference on Information Systems (ICIS) 2010, Saint Louis, Missouri, United States. http://aisel.aisnet.org/icis2010_submissions/102

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain.
- You may freely distribute the URL identifying the publication in the public portal.

Take down policy

If you believe that this document breaches copyright please contact rucforsk@kb.dk providing details, and we will remove access to the work immediately and investigate your claim.

Conceptualizing Interpersonal Relationships in Agile IS Development

Journal:	<i>International Conference on Information Systems 2010</i>
Manuscript ID:	ICIS-0333-2010.R1
Track:	Human Behavior and IT
Keywords:	Relationship, Human factors, Agile software development
Abstract:	<p>Agile information systems development (ISD) is a people-centered approach that emphasizes frequent interaction and genuine co-operation between customers and developers. While business relationships are the norm in the workplace, agile ISD leads to the creation of close interpersonal relationships. Drawing on relationship theory and friendship literature we propose a theoretical framework of three types of workplace relationships. The framework is used for deriving theoretical preconceptions about agile relationships and their impact on the agile ISD team's ability to deliver valuable, working software frequently. We also present the interpretive approach we will apply. An understanding of the relationship specific aspects of agile ISD as well as of the importance and impact of certain relationship characteristics can help explain why some agile projects succeed, while others fail. Such knowledge can also be used to foster friends-like behavior for the benefit of the software outcome in future projects.</p>

CONCEPTUALIZING INTERPERSONAL RELATIONSHIPS IN AGILE IS DEVELOPMENT

Research-in-Progress

Sabine Madsen
Roskilde University
Department of Communication,
Business & IT
Universitetsvej 1
DK-4000 Roskilde, Denmark
sabinem@ruc.dk

Sabine Matook
The University of Queensland
UQ Business School
Information Systems
3 Blair Drive
Brisbane, 4072, QLD, Australia
s.matook@business.uq.edu.au

Abstract

Agile information systems development (ISD) is a people-centered approach that emphasizes frequent interaction and genuine co-operation between customers and developers. While business relationships are the norm in the workplace, agile ISD leads to the creation of close interpersonal relationships. Drawing on relationship theory and friendship literature we propose a theoretical framework of three types of workplace relationships. The framework is used for deriving theoretical preconceptions about agile relationships and their impact on the agile ISD team's ability to deliver valuable, working software frequently. We also present the interpretive approach we will apply. An understanding of the relationship specific aspects of agile ISD as well as of the importance and impact of certain relationship characteristics can help explain why some agile projects succeed, while others fail. Such knowledge can also be used to foster friends-like behavior for the benefit of the software outcome in future projects.

Keywords: Interpersonal relationships, agile IS development, friendships, interaction, helping behavior

Introduction

Recently, agile information systems development (ISD) has become widely diffused and used in practice (Dyba and Dingsoyr 2008). Agile ISD is a particular type of development that recommends that developers and customer representatives work closely together to deliver valuable, working software frequently (Fowler and Highsmith 2001). Agile ISD advocates the team members' physical proximity, frequent interaction, and genuine co-operation towards the shared goal, i.e. software releases (Cockburn 2007). Moreover, inherent in all agile values is a strong focus on skilled individuals and the importance of people in software development (Vidgen and Wang 2009). Agile ISD is therefore a people-centered approach.

The management of relationships between people in software teams has previously been recognized as key to facilitate the utilization of critical skills and expertise (Beath and Orlikowski 1994). However, in traditional waterfall ISD the relationships are rarely explicitly studied with regard to their impact on the software outcome (Sawyer et al. 2008). The agile paradigm now invites this kind of studies because agile ISD provides a number of techniques that rely on interpersonal relationships rather than written documents to produce the software. For example, pair programming in eXtreme Programming (XP) (Beck 1999; Beck and Andres 2005) requires two developers to write the software code (McMahon 2003). The two developers work closely together to complete their task, i.e. their story cards. In general, close relationships among team members have been found to contribute to a positive working atmosphere. Thus, agile ISD enables and requires close interpersonal relationships that can be conceptualized as workplace friendships to form in an agile ISD project.

The concept of friendships in business settings has been studied by examining friendships among colleagues and between customers and service providers (see for example, Berman et al. 2002; Grayson 2007; Haytko 2004; Price and Arnould 1999). A workplace friendship shows the characteristics of a personal friendship but is formed in a business setting. A personal friendship is a close relationship among two individuals that is characterized by, e.g., voluntary interactions among the friends. These interactions are motivated by the friends' communal concern and the expectation of an exclusively intrinsic orientation (Grayson 2007). When a workplace friendship emerges the personal friendship and business relationship between individuals co-occur. At certain times, one relationship type, either the friendship or the business relationship might even dominate the interaction between the individuals. Nevertheless, workplace friendships are characterized by high levels of mutual commitment, trust, reciprocal liking, and shared values or interests between individuals at work (Berman et al. 2002).

Prior research shows that workplace friendships provide a number of benefits. Workplace friendships increase the level of communication, task assistance and support provided, as well as the emotional support among colleagues or team members. These benefits in turn help people get the job done, improve the quality of the work, and reduce stress (Berman et al. 2002). While the friendship literature delineates many relationship characteristics, in this research, we focus primarily on helping behavior and its beneficial impact on the software outcome. A workplace friendship is particularly rewarding because of the relationship partner's co-operation and helpfulness in meeting one's specific day-to-day needs (Wright 1984). The helping behavior exhibited by the relationship partners is also important for the achievement of the shared goal.

The characteristics of agile ISD as a people-centered approach contribute to the formation of close interpersonal relationships, i.e. of workplace friendships among the developers and between the customer representatives and the developers. Therefore, it is important for managers involved in an agile ISD project to have knowledge about how to invoke and benefit from workplace friendships (Grayson 2007). In an agile context this motivates examining how close the interpersonal relationships among the developers and between the customer representatives and the developers are likely to become and which friendship aspects could be fostered and leveraged for the benefit of the software outcome. In this research-in-progress paper, we draw on relationship theory and friendship literature to conceptualize different types of workplace relationships that vary in closeness. Based on an understanding here of we derive theoretical preconceptions about interpersonal relationships in agile ISD and their impact on the developers' ability to deliver valuable, working software frequently.

Background

In this section, we first provide more detail about the relationships that agile ISD emphasizes as well as how agile methods encourage and support the formation and progression of these relationships. Subsequently, we review prior research about different types of relationships. On this basis we propose a theoretical framework of three types of workplace relationships and discuss their implications for the individual and the work.

Relationship-specific aspects of agile IS development

Agile ISD is a development method that encompasses a complete range of practices for designing, building, implementing, and maintaining an information system with a focus on the ability to create and respond to change (Conboy 2009). The agile approach emerged as a response to the shortcomings of traditional, plan-driven ISD (Truex et al. 2000) and is a type of development that is interaction rather than document driven. It emphasizes the developers' and the customers' co-creation of valuable, working software in accordance with verbally communicated, emerging, and changing requirements. As such, interpersonal relationships are the main carriers of the communication, coordination, and collaboration in agile ISD, thereby also enabling or inhibiting the swift reactions to change depending on the quality of the relationships. Even though various relationships among different stakeholders can emerge in ISD projects, agile ISD is primarily concerned with two types of interpersonal relationships, namely the relationships (a) among the developers, and (b) between the customer representatives and the developers. The first type refers to relationships among colleagues in the workplace, and the second type refers to relationships between people in a client-service provider relationship in which monetary interests often are involved. Agile methods support the interaction and collaboration between these two types of project participants in various ways. Below we use examples from XP (Beck and Andres 2005) and Scrum (Schwaber and Beedle 2005) because they are the two best known and most widely used agile methods (Conboy 2009; Dyba and Dingsoyr 2008).

XP's pair programming and Scrum's daily stand-up meetings are examples of practices that support the interaction among developers. In XP's pair programming intensive collaboration among the developers is achieved by putting two developers together to write code for one task on one workstation (Beck and Andres 2005). The aim is to simultaneously produce code and conduct quality checks. It is recommended that the developers mix as much as possible and empirical findings show that programming partners might be changed as frequently as every half hour (Vidgen and Wang 2009). However, the small team size means that the same pairs form often and that the developers therefore have a good understanding and overview of what each other are working on. Scrum's daily stand-up meetings also support interaction among the developers. These meetings are short meetings held each day at the same time and location. In these meetings each developer in the Scrum team answers three questions, namely what was done yesterday, what is planned today, and if there are any problems that block the accomplishment of the day's task (Schwaber and Beedle 2005).

XP's planning game, onsite customer practice as well as Scrum's sprint planning and review meetings facilitate collaboration between the developers and the customer representatives. In XP, both developers and customer representatives participate in the planning game, a meeting that occurs once per iteration (Martin 2003; Poole and Huisman 2001). The aim is to collaboratively determine which system requirements that should be included in the current and near-future releases. The requirements are written down on user story cards. Later the end result in the form of released, working software is matched with the original user stories as well as acceptance tested by the customer. Thus, the XP developers and customer representatives collaborate at the beginning and end of an iteration to identify and test high-priority functionality (Williams and Kessler 2002). Moreover, XP stresses that the developers should have easy and continuous access to information and feedback from the customer throughout the duration of each iteration. It is therefore recommended that the developers and one or more customer representatives should be collocated as this allows for informal face-to-face conversations and collaboration when the need arises (Beck and Andres 2005; Fruhling and De Vreede 2006). In line with XP's planning game, Scrum prescribes that each iteration (i.e., each sprint) starts with a joint meeting between the developers and the product owner (i.e. the customer representative). In the sprint planning meeting the customer representative shares the arguments for the priority of functionality with the developers, and the functionality to be included in the current sprint is chosen (Rising and Janoff 2000). At the end of the sprint, a sprint review meeting is held where the developers show the working software to obtain feedback about the product owner's perceptions and satisfaction with the released product. Even though Scrum does not provide a practice that aims to ensure developer-customer collaboration during the sprint, empirical findings suggest that frequent, informal communication between all developers and the product owner is the norm in agile ISD (Baskerville et al. 2010).

In sum, agile ISD emphasizes and provides practices that support frequent interactions among developers (i.e. among colleagues), and close collaboration between customer representatives and developers (i.e. between people in a client-service provider relationship). Relationship research shows that sheer frequency of interaction, the need to cooperate to produce the result, and shared high-pressure experiences promote the formation of close interpersonal relationships at work (Price and Arnould 1999). Therefore, we turn to relationship theory and the literature about workplace friendship to achieve a conceptual understanding of the relationships that are likely to form in agile ISD.

Characteristics and types of interpersonal relationships

Different types of relationships between individuals can be formed. Each type of relationship is governed by implicit relationship norms and expectations with regard to exchange, equity, intimacy, commitment, trust, coordination, and communication (Morgan and Hunt 1994; Reis et al. 2000; Sirdeshmukh et al. 2002). The relationship norms enable smooth interaction because much can be taken for granted while expectations provide guidelines that moderate individual behavior during an interaction so as to achieve the most favorable outcomes (Reis et al. 2002). One of the reasons that relationships might fail is due to the ignorance of, or disregard for the norms and expectations associated with the particular type of interpersonal relationship (Clark and Mills 1979). Social exchange theory (Emerson 1976; Thibaut and Kelley 1959) argues that individuals only pursue relationships for as long as the relationship is satisfactory in terms of the rewards and cost (Clark and Mills 1993). What an individual perceives to be a relationship reward and a relationship cost depends on the type of relationship. Relationships can be characterized as either communal or exchange oriented. These two relationship types define different norms and expectations between the relationship partners (Clark et al. 1986; Clark and Mills 1993).

Exchange relationships are transactional and utility-oriented in nature. Benefits are given with the expectation of receiving a comparable benefit in return. Exchange relationships are initiated because both relationship partners believe the relationship is of value for them; a means to an end. The interaction between partners is similar to the behavior between polite strangers (Tomiuk and Pinsonneault 2008). Each partner keeps score of what he has invested and neither of them wants to contribute more than the other (Clark et al. 1986). It is commonly assumed in the literature that workplace and business relationships are prototypical examples of an exchange relationship (Tomiuk and Pinsonneault 2008). In contrast, in communal relationships the norm is to give benefits in response to needs (offer help, gifts, and give emotional support) and to express a general concern for the other person. Interactions among partners are cooperative and joint efforts are made to achieve a common goal. In communal relationships, no immediate repay of a benefit is expected and the partners do not feel an obligation to return a benefit received, as in an exchange relationships. Relationships between friends and family members are examples of communal relationships (Clark and Mills 1993).

Despite the common assumption that all workplace and business relationships are exchange relationships some researchers draw on both the communal and exchange orientations to understand social interaction in work settings. For example, two types of business relationships have been identified (Heide and Wathne 2006), namely those of (1) a *friend* who follows the norm of “co-operation in principle”, and (2) a *business person* who makes decisions based on utility-maximization considerations and who will either co-operate or not depending on which action that will be most profitable for him as an individual. Similarly, based on a qualitative study of firm-to-firm relationships, the three types of *strictly business*, *business friend*, and *personal friend* relations are identified (Haytko 2004). The study demonstrates that account managers are willing to go out of their way to help relationship partners they like and feel comfortable with. The definition of the three types focuses on the amount of information disclosed between the partners, the resulting level of intimacy, and ultimately the trusting behavior between them.

In exchange relationships the amount and depth of disclosed information between the partners is low. Consequently, they know very little about each other (Hays 1984). The limited personal disclosure in exchange relationships impairs the creation of trust between the partners (Wheless and Grotz 1976). These low levels of relationship intimacy and trust mean that customers feel a higher need for performance monitoring of service providers. Moreover, in exchange relationships it is uncommon to respond to and express social needs (e.g., emotional stress and feelings of self-worthiness) and to engage in helping behavior. Therefore, it is also not possible to ask a relationship partner who is strictly business for a favor or for the answer to a question that could be interpreted as “goofy”, or incompetent by others (Haytko 2004). Yet, relationships can evolve from non-intimate to intimate areas of communication (Cozby 1973). The more personal a relationship becomes the more people feel comfortable around each other. As a result, they also feel more able to ask for favors and to bring “thorny” issues up (Wright 1984). Thus, communal relationships are characterized by much helping behavior among the partners. The willingness to express and respond to social needs is also high in personal friend relationships.

Building on the dichotomy of communal and exchange relationships, the concept of communality has been suggested as a continuum that reflects the extent to which a workplace relationship resembles a friendship (Goodwin and Gremler 1996). Nonessential conversation (e.g., about the weather and broader work-related topics, such as the appointment of a new CEO, the firm’s internationalization strategy, etc.) and self-disclosure (e.g., about family, hobbies, and sicknesses) that are not directly linked to the delivery of the core service are indicators of the degree of communality in a relationship (Goodwin and Gremler 1996). The amount of nonessential information and depth of

disclosed information increases with higher levels of communality (Wheless and Grotz 1976). It has been found that customers who express communal feelings such as “he’s more like a friend” also report a desire to work to maintain the relationship, and overcome misunderstandings and failures and that similar commitment was expressed by service providers who identified their regular customers as friends (Goodwin and Gremler 1996). Thus, the degree of communality determines future interactions between the relationship partners and influences the partner’s perception of the other’s actions as either genuinely communal or instrumental (Clark and Mils 1993).

Drawing on several references (see e.g. Clark and Mils 1993; Hays 1984; Haytko 2004) Table 1 summarizes the abovementioned key characteristics of the three types of workplace relationships, i.e. business associate, business friend, and personal friend. The eight relationship characteristics are based on general friendship theory and literature about workplace friendship and as such they are all important. However, helping behavior is particularly crucial in business settings in which a shared goal is pursued and tangible outcomes are sought because giving and receiving help speeds up problem solving processes and improves the quality of the work, thereby also ensuring that the job gets done (Berman et al. 2002). The other relationship characteristics can be seen as prerequisites for helping behavior to occur (e.g., some level of trust is required) or as means that facilitate helping behavior depending on the type of relationship. For example, work-related but even more so nonessential conversations about hobbies and family lead to intimate self-disclosure which means that the individuals get to know each other. While it is rewarding to know personal things about your colleagues in itself it also creates an understanding of each others’ situations that makes it more likely that help will be asked for and provided as needed.

Table 1. Different Types of Relationships in the Workplace			
Relationship Characteristics	Business Associate	Business Friend	Personal Friend
Relationship Norms	Individual behavior guided by exchange norms	Individual behavior guided by co-operative and exchange norms	Individual behavior guided by co-operative norms
Degree of Communality	Low	Medium	High
Helping Behavior	No helping behavior	Medium helping behavior	Much helping behavior
Conversation	No nonessential conversation	Medium level of nonessential conversation	Much nonessential conversation
Amount and Depth of Self-Disclosure	No self-disclosure	Medium self-disclosure	Highly intimate self-disclosure
Trust	Low levels of trust that require substantial performance monitoring	Medium levels of trust with occasional performance monitoring	High levels of trust resulting in low performance monitoring
Intimacy	No personal knowledge about each other	Medium personal knowledge about each other	Much personal knowledge about each other
Need responses	No attention to social needs	Supportive responds to social needs	Supportive responds to social needs and emotional stress
Obligation	High level of obligation to return comparable benefit immediately	Some level of obligation	Low level of obligation for equal, immediate repayment

Communal relationships, such of those of personal friends and even business friends, offer various benefits for the projects and organizations that constitute the contexts in which the relationships are initially established and developed. Prior research on project team success has identified group process issues and technical issues as important to task team success (White and Leifer 1986). The following preconceptions will introduce the aspect of friendship relationships as an antecedent of the agile ISD outcome.

Development of theoretical preconceptions

In this section we draw on the three types of workplace relationships and their characteristics to develop an understanding of interpersonal relationships in agile ISD teams and their impact on the team's ability to deliver valuable, working software on a regular basis. The section is structured in two parts where we first address the relationships among developers and subsequently look at the relationships between customer representatives and developers. In each part we summarize our understandings as theoretical preconceptions, i.e. as concise statements.

Relationships among developers

Agile ISD prescribes and provides methodical support for frequent interaction and collaboration among developers. This is exemplified by various practices, such as XP's pair programming, Scrum's daily stand-up meetings, and the self-organizing nature of agile teams. Scrum is a particularly good example of the heightened norms of informality, and self-organization that agile ISD advocates. In Scrum there is no project manager. Instead the Scrum master's task is to make sure that the developers have everything they need (information, software tools, etc.) to get the job done. As such, there is, at least in principle, no formal hierarchy in agile ISD teams; rather the developers and the Scrum master collaborate towards the shared goal of delivering the functionality that is a part of the particular sprint. Due to the frequent face-to-face interactions, and genuine co-operation between equal peers prescribed by agile ISD it seems likely that personal friend relationships among the developers will form and that their actions and interactions therefore will be guided by the norm of co-operation as a matter of principle. Thus, our first theoretical preconception is:

TP 1: In agile ISD, the frequency of opportunities for interaction and co-operation among the developers support the formation of personal friend relationships.

The literature describes personal friend relationships as characterized by a high level of communality. This more specifically means that relationship partners engage in much nonessential and self-disclosing conversations, thereby increasing the knowledge they have about each other as a person. This in turn leads to enhanced trust and a low level of performance monitoring. In practice, the intimacy of the self-disclosure and the extent to which people actually like each other, share similar values and interests, and engage in voluntary interactions outside the office depend on many aspects including company culture and personality. However, in an agile ISD context, getting to know each other and creating trust-based relationships are important aspects as these aspects makes it possible for the team members to voice concerns and ask for help. It also makes people more willing to help. In this way, knowing and trusting each other are prerequisites for the team to fully benefit from the key agile practices for example of pair programming and daily standup meetings. In particular, the daily standup meeting, and especially question three, explicitly invites the team members to voice concerns and to ask for and provide help as needed. Therefore, it seems reasonable to assume that the amount of helping behavior that the team members show towards each other is high. Indeed, high levels of helping behavior correspond with the assumption that the developers have friendship bonds. Thus, our second theoretical preconception is:

TP 2: The personal friend relationships among the developers are in particular characterized by a high level of helping behavior.

Prior research shows that interacting in person makes people work harder to connect and empathize, solves problems faster and arouses performance instincts (Sapsed and Salter 2004). In line with this, we assume that the cumulative and collaborative efforts of the developers to get the day's tasks done, i.e. to produce integrated, tested code each day, and to help each other do so increase the likelihood that the agile ISD team is able to deliver working software to the customer at regular, short intervals. Thus, our third theoretical preconception is:

TP 3: The personal friends relationships and in particular the high level of helping behavior impact positively the developers' ability to deliver working software to the customer frequently.

Relationships between developers and customer representatives

At the general level, the relationship between the developers and the customer representatives is that of a client – service provider relationship, which means that the partners are in an instrumental (and often monetary) relationship towards each other. Yet, at the interpersonal level, agile ISD recommends that the developers and the customer representatives work together often and in a collaborative way to produce the result. Moreover, if the developers and

customer representatives are collocated the two types of project partners might think of themselves and be seen by others as a team (Cohen & Bailey, 1997). However, in practice the actual level of informal interaction and co-operation between the developers and customer representative can vary despite the use of agile practices (Conboy 2009). The balance between exchange and co-operative norms might therefore also vary from project to project, thereby making the relationship more or less communal in nature depending on the actual display of norms. It also seems reasonable to assume that certain situations might call more naturally for co-operative norms (e.g., the planning game and sprint planning meetings), while other situations might cause exchange norms to come into play (e.g., the review meetings and acceptance test). Thus, although agile ISD values and practices strongly encourage and support frequent interaction and co-operation, the exchange orientation is inherent to the developer-customer relationship. It therefore also seems likely that the interpersonal relationships between the developers and the customer representatives will be that of business friends and that their actions and interactions will be guided by both co-operative and exchange norms. Hence, our fourth theoretical preconception is:

TP 4: In agile ISD, the frequency of opportunities for interaction and co-operation between developers and customer representatives support the formation of business friend relationships.

The literature describes business friend relationships as of medium level of nonessential conversation, self-disclosure, and intimacy. Consequently, the partners have some personal knowledge about each other and trust is at a medium level and therefore occasional performance monitoring is conducted. In agile ISD the customer representative serves as an onsite or easily available contact for the developers because of their domain knowledge. Professional and behavioral differences contribute to increased need for and occurrence of nonessential conversations and self-disclosure. This leads to reciprocal knowledge gains about each other and each others' broader work situation (e.g., the customer representatives' needs with regard to internal politics, and the developers' needs for information relevant to estimation and subsequent development of user stories). In this way, the developers get a better understanding of the customer's organization and the context in which the IS will be used, and at the same time, they might benefit from the customer representative's increased willingness to engage in helping behavior. Moreover, the opportunities that agile ISD creates for both essential and nonessential communications to take place between customers and developers can help ensure that the customer representative feels needed and as a part of the team. However, despite that mutual knowledge is achieved and helping behavior exercised the developer-customer relationships are also characterized by the customer's occasional performance monitoring. The performance monitoring typically takes place during joint review meetings and when the software is released to the customer for user acceptance testing. Thus, our fifth theoretical preconception is:

TP 5: The business friend relationships between developers and customer representatives are in particular characterized by a medium level of helping behavior and occasional performance monitoring.

Certain agile practices such as, e.g., XP's planning game and Scrum's sprint meetings create the setting and methodical structure for the customer representative to explain requirements and their priorities and to provide answers to the developers' questions in the beginning of an iteration. According to the agile principles the customer representative should also be available to provide help as needed during the iteration. Moreover, the performance monitoring that takes place at the end of each iteration (e.g., in the review meetings) helps ensure that the developers receive feedback on whether the produced software meets the requirements. The performance monitoring also ensures that the developers stay aware of the exchange oriented nature of the relationship they have with the customer, and therefore the obligation they have to deliver value; even when the relationship with the customer representative is otherwise close. Both the help that the developers receive from the customer representative in order to come to understanding of the requirements and their priorities as well as the customer's occasional performance monitoring impact positively the developers' ability to deliver valuable software frequently. Thus, our sixth theoretical preconception is:

TP 6: The business friend relationships between the developers and customer representatives and in particular the combination of medium level of helping behavior with occasional performance monitoring impact positively the developers' ability to deliver valuable software to the customer frequently.

Method

The theoretical preconceptions presented above constitute our current conceptualization of how different interpersonal relationships impact the agile ISD team's ability to release valuable, working software frequently. Together the theoretical framework and derived preconceptions (Klein and Myers 1999) form our starting point for conducting interpretive case study research (Creswell 2003; Walsham 1995). Interviews with developers and customer representatives will be the main data collection method and are planned to take place in July 2010. A semi-structured interview guide that contains questions about relationship type, characteristics, expectations, norms, implications, and impact will be developed. We will in particular ask questions that help establish and explain the link between different types of relationships and their characteristics and the success criteria of agile ISD.

According to the agile manifesto, the objective and success criteria of agile ISD is to deliver valuable, working software to the customer frequently. 'Frequently' means that software should be released every few weeks rather than months. The concept of 'working software' is the principal measure of progress from a development point of view and refers to tested and executable code that can produce an output. We will use iteration velocity and burn-down charts as measures for expected and actual production of working software per iteration. 'Valuable software' summarizes the agile principle that states that customer satisfaction should be achieved by delivery of software that the customer considers useful. We will measure the satisfaction by building on discrepancy theory to compare the expected and perceived performance of story cards per iteration.

The theoretical framework and preconceptions will also be used during data analysis, as a means for coding the data, identifying categories, and for structuring the presentation of the case study account. At the same time, we will strive to stay open to emerging codes, and categories. By the time of the conference, we will have completed the data analysis and shared our empirical findings with colleagues to obtain external validation. The objective of the interpretive study is to arrive at a nuanced understanding of different types of relationships in agile ISD as well as of the importance and impact of certain friendship characteristics, and in particular of helping behavior, on the software outcome. Also, the objective is to create an empirical foundation that can help further advance the theoretical ideas. In a subsequent research project, we will use this nuanced understanding to develop hypothesis and related constructs to capture the topic of interpersonal relationships in agile ISD with a positivist research approach.

Conclusion

In this paper we address the people-centered aspects of agile ISD. The aim is to understand how the closeness of the interpersonal relationships between key project participants impacts the agile ISD team's ability to deliver valuable, working software frequently. Agile ISD provides various practices that encourage frequent interaction and collaboration among developers and between customer representatives and developers, thereby also supporting the formation of close working relationships among these project participants. Drawing on relationship theory and friendship literature a theoretical framework about different types of workplace relationships was developed and used for deriving theoretical preconceptions about agile relationships and their impact. An understanding of the relationship specific aspects of agile ISD as well as of different types of workplace relationships can be used by researchers to help explain why some agile projects succeed, while others fail. Moreover, knowledge about the importance and impact of certain relationship characteristics can be used by agile project managers, Scrum masters, and others (e.g. the customer representative) to take action that helps foster friends-like behavior that influences the software outcome positively.

References

- Baskerville, R., Pries-Heje, J., and Madsen, S. "From exotic to mainstream: A 10-year odyssey from internet speed to boundary spanning with Scrum," in: *agile software development: current research and future directions*, (eds.) Dybå T., Dingsøy T., Moe, N.B., Springer, 2010, Chapter 5.
- Beath, C., and Orlikowski, W. "The contradictory structure of systems development methodologies: Deconstructing the IS-user relationship in information engineering," *Information Systems Research* (5:4) 1994, pp. 350-377.
- Beck, K. "Embracing change with extreme programming," *Computer* (32:10) 1999, pp. 70-77.
- Beck, K., and Andres, C. *Extreme programming explained: Embrace change*, (2 ed.) Addison-Wesley Boston, MA, 2005.
- Berman, E., West, J., and Richter Jr, M. "Workplace relations: Friendship patterns and consequences (According to managers)," *Public Administration Review* (62:2) 2002, pp. 217-230.
- Clark, M., and Mills, J. "Interpersonal attraction in exchange and communal relationships," *Journal of Personality and Social Psychology* (37:1) 1979, pp. 12-24.
- Clark, M., Mills, J., and Powell, M. "Keeping track of needs in communal and exchange relationships," *Journal of Personality and Social Psychology* (51:2) 1986, pp. 333-338.
- Clark, M., and Mils, J. "The difference between communal and exchange relationships: What it is and is not," *Personality and Social Psychology Bulletin* (19:6) 1993, p 684.
- Cockburn, A. *Agile software development: the cooperative game* Addison-Wesley, 2007.
- Cohen, S.G., and Bailey, D.E. "What makes teams work: Group effectiveness research from the shop floor to the executive suite," *Journal of Management*, (23:3) 1997, 239-290.
- Conboy, K. "Agility from first principles: Reconstructing the concept of agility in information systems development," *Information Systems Research* (20:3) 2009, pp. 329-354.
- Cozby, P. "Self-disclosure: A literature review," *Psychological Bulletin* (79:3) 1973, pp. 73-91.
- Creswell, J. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* Sage, Thousand Oaks, CA, 2003.
- Dyba, T., and Dingsoyr, T. "Empirical studies of agile software development: A systematic review " *Information and Software Technology* (50:9-10) 2008, pp. 833-859.
- Emerson, R. "Social exchange theory," *Annual review of sociology* (2:1) 1976, pp. 335-362.
- Fowler, M., and Highsmith, J. "The Agile Manifesto," *Software Development* (9:8) 2001, pp. 28-35.
- Fruhling, A., and De Vreede, G.-J. "Field experiences with eXtreme programming: developing an emergency response system," *Journal of Management Information Systems* (22:4) 2006, pp. 39-68.
- Goodwin, C., and Gremler, D.D. (eds.) *Friendship over the counter: How social aspects of service encounters influence consumer service loyalty*. JAI, Greenwich, 1996.
- Grayson, K. "Friendship versus business in marketing relationships," *Journal of Marketing* (71:4) 2007, pp. 121 - 139.
- Hays, R.B. "The development and maintenance of friendship," *Journal of Social and Personal Relationships* (1:1) 1984, pp. 75-98.
- Haytko, D. "Firm-to-firm and interpersonal relationships: Perspectives from advertising agency account managers," *Journal of the Academy of Marketing Science* (32:3) 2004, pp. 312-328.
- Heide, J.B., and Wathne, K.H. "Friends, businesspeople, and relationship roles: A conceptual framework and a research agenda," *Journal of Marketing* (70:3) 2006, pp. 90-103.
- Hinds, P., and Bailey, D. "Out of sight, out of sync: Understanding conflict in distributed teams," *Organization Science* (14:6) 2003, pp. 615-632.
- Hinds, P., and Mortensen, M. "Understanding conflict in geographically distributed teams: The moderating effects of shared identity, shared context, and spontaneous communication," *Organization Science* (16:3) 2005, pp. 290-307.
- Hirschheim, R., and Klein, H. "Four paradigms of information systems development," *Communications of the ACM* (32:10) 1989, pp. 1199-1216.
- Janis, I. *Victims of groupthink: A psychological study of foreign-policy decisions and fiascoes* Houghton Mifflin Boston, 1972.

- Klein, H., and Myers, M. "A set of principles for conducting and evaluating interpretive field studies in information systems," *MIS Quarterly* (23:1) 1999, pp. 67-93.
- Martin, R. *Agile software development: principles, patterns, and practices* Prentice Hall PTR Upper Saddle River, NJ, USA, 2003.
- McAvoy, J., and Butler, T. "The role of project management in ineffective decision making within Agile software development projects," *European Journal of Information Systems* (18:4) 2009, pp. 372-383.
- McMahon, J. "Five lessons, from transitioning to eXtreme programming: Using two person teams to write software code is one aspect of eXtreme Programming (XP). Here's what Citect learned when it implemented the XP method," *Control Engineering* (50:3) 2003, pp. 59-60.
- Morgan, R.M., and Hunt, S.D. "The commitment-trust theory of relationship marketing," *The Journal of Marketing* (58:3) 1994, pp. 20-38.
- Poole, C., and Huisman, J. "Using extreme programming in a maintenance environment," *IEEE Software* (18:6) 2001, pp. 42-50.
- Price, L.L., and Arnould, E.J. "Commercial friendships: Service provider-client relationships in context " *Journal of Marketing* (63:4) 1999, pp. 38-56.
- Reis, H.T., Capobianco, A., and Tsai, F.-F. "Finding the person in personal relationships," *Journal Of Personality* (70:6) 2002, pp. 813-850.
- Reis, H.T., Collins, W.A., and Berscheid, E. "The relationship context of human behavior and development," *Psychological Bulletin* (126:6) 2000, pp. 844-872.
- Rising, L., and Janoff, N. "The Scrum software development process for small teams," *IEEE Software* (17:4) 2000, pp. 26-32.
- Sapsed, J., and Salter, A. "Postcards from the edge: Local communities, global programs, and boundary objects," *Organization Studies* (25:9) 2004, pp. 1515-1534.
- Sawyer, S., Guinan, P., and Coopriider, J. "Social interactions of information systems development teams: a performance perspective," *Information Systems Journal* (20:1) 2008, pp. 81-107.
- Schwaber, K., and Beedle, M. *Agile software development with Scrum* Prentice Hall, Upper Saddle River, NJ, 2005.
- Sirdeshmukh, D., Singh, J., and Sabol, B. "Consumer trust, value, and loyalty in relational exchanges " *The Journal of Marketing* (66:1) 2002, pp. 15-37.
- Thibaut, J., and Kelley, H.H. *The social psychology of groups* Wiley, New York, 1959.
- Tomiuk, D., and Pinsonneault, A. "Applying relationship theories to web site design: development and validation of a site-communality scale," *Information Systems Journal* (19:4) 2008, pp. 413-435.
- Truex, D., Baskerville, R., and Travis, J. "Amethodical systems development: The deferred meaning of systems development methods," *Accounting, Management and Information Technologies* (10:1) 2000, pp. 53-79.
- Vidgen, R., and Wang, X. "Coevolving systems and the organization of agile software development," *Information Systems Research* (20:3) 2009, pp. 355-376.
- Walsham, G. "Interpretive case studies in IS research: Nature and method," *European Journal of Information Systems* (4:2) 1995, pp. 74-81.
- Wheless, L.R., and Grotz, J. "Conceptualization and measurement of reported self-disclosure," *Human Communication Research* (2:4) 1976, pp. 338-346.
- White, K.B., and Leifer, R. "Information systems development success: Perspectives from project team participants," *MIS Quarterly* (10:3) 1986, pp. 215-223.
- Williams, L., and Kessler, R. *Pair programming illuminated* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2002.
- Wing, M. "In Praise of SE Geeks," *ACM SIGSOFT Software Engineering Notes* (30:4) 2005, pp. 8-9.
- Wright, P.H. "Self-referent motivation and the intrinsic quality of friendship," *Journal of Social and Personal Relationships* (1:1) 1984, pp. 115-130.