

Global Agile Team Configuration

Jason H. Sharp
Tarleton State University

Sherry D. Ryan
University of North Texas

Drawing upon extant literature, we first construct a framework consisting of three major dimensions (agility, virtualness, and structure) to address the question of how productive agile teams can be configured in globally distributed environments. We then propose a configurational pattern for global agile teams by taking each of the twelve principles of the Agile Manifesto and describe its intersection with each of the dimensions of the framework. Our examination indicates that eight principles are especially relevant for the configuration of global agile teams, while the four remaining principles are either unsupported by previous literature or do not constitute unique considerations.

INTRODUCTION

In recent years the use of agile software development methods, which describe ways of producing software in a lighter, quicker, more people-centered way, has been recommended to alleviate the traditional challenges associated with software development (Abrahamson, et al., 2003). Based upon the fact that many organizations have already been using distributed teams for some time, and in many cases globally distributed, there is an increasing stream of research examining the concept of distributed agile development (e.g., Agerfalk & Fitzgerald, 2006; Holmstrom, et al., 2006; Lee, et al., 2006; Lee & Xia, 2010; Ramesh, et al., 2006; Sarker, et al., 2009; Sarker & Sarker, 2009).

According to the Agile Manifesto, a fundamental principle of agile methods is the efficacy of collocated teams in order to enable daily, face-to-face (FTF) interaction between stakeholders (Fowler & Highsmith, 2001). Considering that within the context of global software development, virtual team members may never meet in person a potential dilemma exists for organizations that are considering the use of agile methods in a distributed environment (Gibson & Cohen, 2003). Fortunately, for such organizations a growing stream of research suggests that, although it is sometimes difficult and takes great care, it is possible through the modification of the agile method to the distributed setting (Agerfalk & Fitzgerald, 2006; Fitzgerald, et al., 2006; Holmstrom, et al., 2006; Kircher, et al., 2001; Sarker & Sarker, 2009; Sarker et al., 2009; Schummer & Schummer, 2001; Xiaohu, et al., 2004).

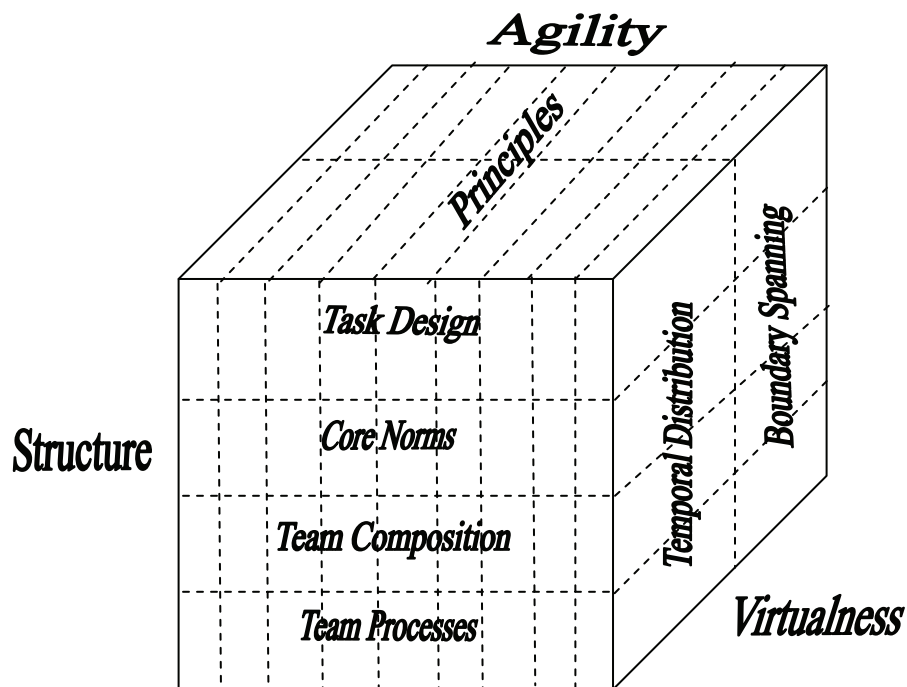
CONFIGURING GLOBAL AGILE TEAMS FRAMEWORK

Team configuration in global settings is a complex phenomenon. While it is true that globally distributed teams encounter many of the same challenges as colocated teams, these are often exacerbated

by physical distance and cultural issues (Komi-Sirvio & Tihinen, 2005; Shachaf, 2008). Thus, the actual configuration of agile teams in globally distributed environments appears to be a significant area of research that has currently received minimal attention. Drawing upon configurational theory, the software agility literature, work group design research, and team virtualness concepts, we propose that it is possible to successfully configure a global agile software development team, but that there are issues that must be well thought-out and that make this particular type of team different from other global teams based upon the use of an agile methodology.

A configuration may be “denote any multidimensional constellation of conceptually distinct characteristics that commonly occur together” (Meyer, et al., 1993, p. 1175). In other words, a configuration is a pattern that describes an entity. Although a considerable amount of work on configurations has been conducted at the organizational level (Miles & Snow, 1978; Mintzberg, 1979, 1983), much less has been done at the group level. A team is representative of a group level configuration. Therefore, we pose the following research question: How can a productive agile team be configured in a globally distributed environment? To address this question we review the literature and construct a framework consisting of three dimensions: team agility, virtualness, and structure as illustrated in Figure 1. Each of these major dimensions are discussed in the sections below, followed by a discussion of their intersections.

FIGURE 1
GLOBAL AGILE TEAM CONFIGURATION FRAMEWORK



Team Agility

The first major dimension of our framework is team agility. Agile methods embrace the values of individuals and interactions, working software, customer collaboration, and responding to change. Traditional methodologies, on the other hand, emphasize comprehensive documentation, contract negotiation, and following a plan (Fowler & Highsmith, 2001). The Agile Manifesto outlines twelve specific principles that have been established to guide agile development as listed in Table 1. These principles emphasize the need for early and continuous delivery of software, openness to changing requirements, delivering working software on a frequent basis, strong interaction between stakeholders,

supporting and motivating team members, promoting sustainable development, fostering technical excellence, and regular feedback.

TABLE 1
SUMMARY OF THE PRINCIPLES BEHIND THE AGILE MANIFESTO

Principles
1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Business people and developers must work together daily throughout the project.
3. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
4. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
5. The best architectures, requirements, and designs emerge from self-organizing teams.
6. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.
7. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
8. Simplicity—the art of maximizing the amount of work not done—is essential
9. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
10. Working software is the primary measure of progress.
11. Continuous attention to technical excellence and good design enhances agility
12. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Team Virtualness

The second major dimension of our framework is team virtualness. Bell and Kozlowski (2002) proposed a typology of virtual teams that consisted of boundary spanning, temporal distribution, life cycles and member roles. Boundary spanning indicates that virtual teams can cross functional, organizational, and cultural boundaries. Temporal distribution denotes that a virtual team is distributed across time. Virtual team members may be colocated in time, separated by only a few hours, or separated by many hours. Members may also be temporally synchronized, e.g., located in different time zones, but are still working off of the same time reference. The life cycle dimension suggests that virtual teams may not follow the traditional life cycle that occurs in traditional teams. Finally, the member roles dimension implies that members may participate in multiple roles within a team. As shown in Fig. 1, we did not include these later two dimensions in our framework because we think they overlap with, and are more appropriately included in, the third dimension in our framework, team structure. We argue that these characteristics are directly affected by the team structure, which is one of the reasons it is so important to carefully consider its design (Hackman, 2002).

Team Structure

The third major dimension of our framework is team structure. Meyer et al. (1993) suggested that work group design represents a possible group level configurational approach (Hackman, 2002; Hackman & Oldham, 1980; Hackman & Walton, 1986). In work group design research, the structural elements play an important role in the effectiveness of the team. Hackman (2002, p. 13) stated, “it is a fantasy – a tempting and pervasive one, but a fantasy nonetheless – that it is possible to have great teams without the bother of creating enabling team structures”. Although work group design research addresses teams in general, “the structural conditions that foster effectiveness of face-to-face teams are just as critical for virtual teams – but with one caveat: *it is much harder to create those conditions in virtual teams*” (Hackman, 2002, p. 131). Unfortunately, the design of the structural characteristics of virtual teams often appears as if it were only an afterthought (Powell, et al., 2004). As a dimension of the proposed

framework, team structure includes the sub-dimensions of task design, core norms of conduct, team composition, and team processes. In sum, Powell et al. (2004, p. 6) stated, “we believe that investigation of team structure in the virtual environment holds significant promise for research and practice because it represents perhaps the most controllable and influential aspect of virtual team design”.

Team Processes

In regard specifically to global virtual team structure, Prasad and Akhilesh (2002) also suggested that team processes are an important structural element and consist of several considerations. The first deals with the mechanism for making decisions in terms of its centralization and formality. The decision-making process dovetails with task design and the autonomy of the team. The second process addresses the degree of information sharing between the members of the team and their participation in the long-range planning of projects. The third process deals with the modes of control and communication and coordination. The fourth process involves the degree of commonality in work process and technology infrastructure. This is an important process due to the potential problems caused by strategic issues such as division of work, time zone differences, and technological inconsistencies between distributed sites.

Core Norms

Core norms of conduct indicate the acceptable and unacceptable behaviors of the team (Hackman, 2002). An important element of virtual team design is the establishment of a shared set of norms which direct the individual and corporate behavior of members (Sarker, et al., 2001; Suchan & Hayzak, 2001). During the formation of the team there may be much ambiguity about member roles, overall goals, and the rules which will govern the actions of the team. As such, the team leader or project manager will be called upon to begin the process of defining these areas. It is important that each team member positively internalizes this set of rules and in essence "buys-in" to their use (Sepulveda, 2003).

Team Composition

Team composition addresses the elements of size, mix, interpersonal skills, and task-related knowledge and skill. Hackman (2002, p. 118) advocated having as few team members as possible to accomplish the task, in fact, “a team may function better when it has slightly fewer members than the task actually requires”. Determining the size of the team, therefore, is dependent on the complexity of the task. Powell et al. (2004) stated that to their knowledge no specific study to date has been conducted that explicitly examined virtual team size as a variable during the team design phase.

Ideally, teams should be composed of members who meet a predefined standard for interpersonal skills, “some people just are not cut out to be team players” (Hackman, 2002, p. 125). Similarly, Suchan and Hayzak (2001) argued that virtual team members must possess excellent interpersonal and conflict management skills. The importance of addressing interpersonal skills within the context of team structure cannot be overstated. Hackman (2002) contended that in an appropriately structured team the number of interpersonal conflicts will be less than in a team for which the task, norms, and composition were given little or no thought.

Task Design

Task design deals with the construction of the work itself. Team structure is dependent on the work performed (Prasad & Akhilesh, 2002). According to Powell et al. (2004), significant attention has been paid to the design of virtual team interaction, but much less attention has been given to the design of the work unit itself. The overall goal of a well constructed design is to facilitate collective internal work motivation. In order to reach this goal, task design seeks to provide team members with a meaningful challenge, to offer regular performance assessments, and to allow the team to practice significant autonomy (Hackman, 2002).

Summary of Framework Dimensions

Overall the choice to include these three dimensions into our framework was based upon a thorough review of the literature in regard to the agile methodology, virtualness, and team design. In terms of agility, the framework incorporates the principles of the Agile Manifesto. The typology set forth by Bell and Kozlowski (2002) appears to encapsulate the primary elements pertaining to the virtualness of globally distributed teams as described in the virtual team literature. For example, Prasad and Akhilesh (2002) emphasized that an important contextual aspect of a virtual team was its degree of virtualness which included such characteristics as the measure of geographically dispersion and the temporal nature of the team. In regard to team structure, the dimensions provided by Hackman (2002) and Prasad and Akhilesh (2002) cover a broad range of sub-dimensions that relate to all teams, but also to virtual teams in particular. Prasad and Akhilesh (2002) addressed the characteristic of team composition and membership directly as it related to virtual teams suggesting its importance in any definition of a virtual team. Kirkman, Rosen, Gibson, Tesluk, and McPherson (2002) suggested that identifying virtual team members who have a healthy balance of technical and interpersonal skills as one of the primary challenges related to virtual team success. Finally, in terms of team processes, Lurey and Raisinghani (2001) found that there was a strong relationship between the team's processes and team performance and team member satisfaction.

CONFIGURATIONAL PATTERN: INTERSECTION OF FRAMEWORK DIMENSIONS

According to Bose (2008, p. 626) the Agile Manifesto is a “well accepted benchmark to judge agile projects”. In a review of the literature he identified twelve case studies on the successful implementation of distributed agile software projects. He then analyzed and synthesized the findings from the perspective of the values and principles of the Agile Manifesto. Bose indicated that in the case studies examined, 11 of the 12 principles of the Agile Manifesto were evidenced to some degree. The result of his analysis was that not all principles enumerated in the Agile Manifesto were considered to be important. Subsequently, in this section we propose a configurational pattern for global agile teams by taking each principle of the Agile Manifesto and describe its intersection with the other dimensions and applicable subdimensions of our framework as shown in Table 2 (see appendix). We argue that eight agile principles are especially relevant for consideration of global agile teams and that the remaining four principles are either unsupported by previous literature or do not engender unique considerations for globally distributed teams. Where differences exist between Bose's analysis and our own, in regard to the inclusion or exclusion of specific principles, a justification is provided under that principle.

Principle 1 - Our Highest Priority is to Satisfy the Customer Through Early and Continuous Delivery of Valuable Software

Satisfying the customer is more challenging in a global, distributed environment because of both boundary spanning and temporal distribution challenges. In terms of boundary spanning issues, differences in national cultures pose challenges to conveying information in which the meaning and priorities of that information are shared. Culture can have a great impact on how individuals interpret and react to various situations (Kotlarsky & Oshri, 2005). Factors such as the need for structure, attitude toward organizational hierarchy, sense of time, language barriers, and overall attitude toward international development all come into play in some way (Battin, et al., 2001; Herbsleb & Moitra, 2001).

Temporal distribution also leads to difficulty in requirements engineering and can potentially impact the communication processes in which timely feedback is received, especially when the timeframe for development is short and the time zone difference is large. In a case study of requirements engineering with users and developers in the UK and New Zealand, Hanisch and Corbitt (2007) found that distribution among the development team members as well as the distance between the developers and the users created problems. Difficulties arose because of the complexities of communicating detailed design requirements through a simple electronic medium (email). Therefore, as with other studies (e.g., Carmel, 1999), the loss of communication richness resulting from cyberinfrastructure tools usage to compensate

for physical separation has been documented as a problem. Communication difficulties between the developers and the users revolved around getting the users to answer questions or clarify issues.

As development teams seek customer satisfaction through early and continuous delivery of valuable software, all four structural configurational characteristics should be considered. First, team processes should be designed so that there is regular feedback, both from team members and customers. Planning is necessary for ensuring that the appropriate cybertools with an adequate level of media richness are used so that feedback is effectively provided. The agile practice of short iterations requires that the technology tools be available both to the developers as well as the customers so that prototypes can be evaluated. Some research has indicated that “small releases” may not always be feasible throughout the development effort, based on local constraints (Fitzgerald, et al., 2006). Yet, the task design done prior to the inception of the project would ideally identify these constraints and provide the needed feedback. Another important aspect in designing feedback loops is identifying and gaining access to key users, especially early in the project. This means that regular communications flows should be established.

The core norms of the team must address cultural differences in which misunderstandings can lead to low satisfaction in customers as well as among team members. Cultural training at the project’s inception can aid in the alignment of cultural frames of reference. For example, in a study of US and Norwegian systems development teams (Sarker & Sahay, 2004), the US members thought the Norwegians were “abrupt and blunt” in their communication style. Cultural understanding is required so that misunderstandings or hurt feelings won’t ensue. Additional examples include awareness of national holidays and festivals and recognition of differences in colloquial language usage.

Also, the team composition should carefully be considered. Team members must be selected with the appropriate type of interpersonal skills. Miscommunication and erroneous interpretation of requirements can occur when team members have poor communication skills or are inexperienced, especially in a globally distributed team (Hanisch & Corbitt, 2007).

Finally, task design should be configured so that developers have meaningful challenges and regular assessments of their performance. Dividing tasks among developers in multiple locations can be a complex. However, to ensure customer satisfaction with the development effort, and completion of tasks in a timely manner, planning in this regard is imperative. Team processes and task design somewhat intersect in that team processes must incorporate feedback loops, and that feedback can provide individual developers assessments of their performance.

Principle 2 - Business People and Developers Must Work Together Daily Throughout the Project

In a study conducted by Komi-Sirvio and Tihinen (2005), participants involved in global software development were provided with a list of eight different problem areas identified from prior research and ask to indicate which problems they had experienced. Communication and contacts ranked second (74%), with cultural differences (included in boundary spanning issues) and physical distances (included in temporal distribution) repeatedly occurring. Temporal distribution makes it challenging for business people and developers to work together on a daily basis (Hanisch & Corbitt, 2007). It is easier if at least some of the team members are on time schedules that are not polar opposites. This implies that ideally, at least part of the team should be in time zones where at least some of the standard business day overlaps.

Team processes, in terms of when and how to communicate, are critical (Rennecker & Goodwin, 2005; Ocker & Fjermestad, 2008). With respect to the issues of distance and communication, Herbsleb and Mockus (2003, p. 481), stated that, “in contrast to the frequent interaction of colocated work, there is very convincing evidence that the frequency of communication generally drops off sharply with physical separation among coworkers’ offices and that the sphere of frequent communication is surprisingly small”. Daily stand-up meetings are consistently emphasized in agile methodologies. Even within a global agile team daily (or periodic) stand-ups can be implemented through the use of synchronous, visual technologies such as videoconferencing. A practical consideration is attempting to schedule these meetings during the overlapping time during the work day, which affords the opportunity for each member to briefly summarize what they have accomplished or are planning on doing for the day.

In addition to scheduling meetings as a form of communication and coordination, core norms must be established. For example, a norm might be instituted as to how often each day messages are checked and responded to. This might mean that a message is sent that states the member is working on a response, even if the response is not completed. Explicitly stated norms help avert problems arising from geographical separation (Sarker & Sahay, 2004).

Team composition is also a consideration in which team members should be chosen that have a bent for collaboration, which is critical for business people and developers as they work together as stated by this principle. Similar to colocated agile teams, it is also recommended that the size of global agile teams be kept as small as possible based upon the scope and complexity of the project. Due to the distributed nature of the team, configuring a team that is too large may decrease the level of communication and coordination. This may be evidenced in the daily stand-up meetings by lack of involvement, loss of focus, or simply by the meeting going on for too long.

Principle 3 - The Most Efficient and Effective Method of Conveying Information to and Within a Development Team is Face-to-Face Conversation

It is widely accepted that cultural differences present a significant challenge to global systems development and that those differences can be exacerbated in a non-face-to-face environment (e.g., Carmel, 1999; Damian & Moitra, 2006; Evaristo, et al., 2004; Herbsleb & Moitra, 2001). Cultural differences and lack of shared meaning (Carmel, 1999; Hanisch & Corbitt, 2007; Herbsleb & Moitra, 2001) are common problems. Fortunately, a study by Shachaf (2008, p. 139) indicated that information and communications technologies “mitigated the negative impact of cultural diversity on team effectiveness while supporting the positive impact”.

In a study conducted by Herbsleb and Mockus (2003) it was found that work distributed across sites appears to take two and one-half times longer than similar projects where the entirety of the work is done in a colocated environment. The study showed that that size, diffusion, and number of people were all directly related to the delay. Interestingly, however, there was no direct link between the amount of delay and the distributed nature of the work. This suggested that the number of people working on the project had a more significant influence on the amount of delay than the geographic distance. There is evidence that suggested that as a team grows in size, team productivity actually decreases rather than increases (Hackman, 2002). Therefore, as stated earlier, keeping the team size small is recommended.

Configuring team processes that utilize technology to compensate for lack of FTF efficiencies is critical, but can also be a source of problems. Technologies that communicate not only informational content, but also mediums such as video conferencing in which not verbal meaning can be derived, are important. The loss of “communication richness” is a significant problem often caused by the physical distance and time zone differences (Evaristo, et al., 2004). As such, it is extremely important that protocols be established for facilitating both official and informal communication. Official communication may be distributed via the organization’s intranet, email, and in some cases, telephone calls. Informal communications, which often takes place around the coffee pot or across a cubicle, is much more difficult to imitate in a globally distributed environment. This lack of informal communication may lead to issues that “go unrecognized or lie dormant and unresolved for extended periods of time” (Herbsleb & Moitra, 2001, p. 18). Fortunately, as advances in information and communication technologies such as instant messaging, audio and video conferencing, and groupware applications continue, the difficulties encountered in informal communication may be alleviated. By using such technologies the team can implement communication based agile practices such as the daily stand-up, iteration planning, iteration demos, iteration retrospectives, and pair programming.

Principle 4 - Build Projects Around Motivated Individuals. Give Them the Environment and Support They Need, and Trust Them to Get the Job Done

Trust is a key component, which can be more difficult with temporal distribution issues in which team members have never had FTF interaction (Hanisch, et al., 2001; Mitchell & Zigers, 2009). A major concern is choosing team members that are motivated individuals. While this is not unique to global

distributed environments, it appears more critical because motivation would not come from physical presence of having other team members “looking over one’s shoulder.”

The task design must facilitate collective internal work motivation. The tasks should be meaningful to the team member who is assigned the task. This means that the tasks should be significant and challenging. Strategically, the ideal arrangement would, to a large degree, allow each site to work independently while still fostering flexible and effective communication (Herbsleb & Moitra, 2001). Another aspect of task design is ensuring that team members clearly understand how their part of the project fits into the larger project scheme. While team members should be “self-starters”, and should be given the latitude and the resources to accomplish the tasks assigned, they must also be able to work well with others (Hackman, 2002).

The concepts of short iterations and small releases are important to the design of the tasks and relate to the areas of meaningfulness, autonomy, and feedback. In regard to meaningfulness members are able to see the result of their work within a short period of time, such as a matter of weeks rather than months, which may promote a stronger sense accomplishment and thus increase motivation. Allowing the members a high- to moderate level of autonomy within each iteration empowers them to apply the practices they deem most suitable to meet the requirements of the iteration as well as employ personal creativity to ensure the needed requirements are complete by the end of the iteration. Finally, regular assessments provided during and after each iteration (i.e., stand-up meeting, iteration retrospective), in weeks rather than months, enables members to make necessary changes and respond more quickly to customer requests.

Principle 5 - The Best Architectures, Requirements, and Designs Emerge from Self-Organizing Teams

Self-organization is more difficult in a virtual, globally distributed environment in which members derive rules of behavior and come up with a self-organizing team hierarchy and values. Two of the critical areas in which boundary spanning and temporal distribution create difficulty are coordination and control. Due to the importance of coordination in globally distributed teams, Kotlarsky, vanFenema, and Willcocks (2008) suggested that a knowledge-based perspective be developed in order to inform teams in regard to not only information flows, but also in terms of knowledge process. Coordination may be defined as the integration of “each task with each organizational unit, so the unit contributes to the overall objective”; whereas, “control is the process of adhering to goals, policies, standards, or quality levels”; and “communication is a mediating factor affecting both coordination and control” (Carmel & Agarwal, 2001, p. 23). Thus, team processes which establish effective communication modes serve as a crucial intermediary between coordination and control in globally distributed environments. According to Carmel and Agarwal (2001, p. 23), therefore, the main challenge in global software development is summarized as follows: “distance negatively affects communication, which in turn reduces coordination effectiveness”. Similarly, Herbsleb and Moitra (2001) argued that because multi-site development typically takes significantly longer than comparable colocated tasks, the interplay between communication, coordination, and control is extremely important in the amount of delay caused by geographic distance.

When establishing team processes, organizations must evaluate the trade-offs between the advantages and disadvantages of synchronous and asynchronous communication (Carmel & Agarwal, 2001). Asynchronous communication typically includes email, voice mail, discussion groups, and groupware. Synchronous communication consists of telephone calls, audio and video conferencing, application sharing, and instant messaging. Although asynchronous communication technologies have become a part of normal life, Carmel and Agarwal (2001) argued that there are numerous advantages for incorporating synchronous communication into the GSD process. These include the resolution of miscommunications and misunderstandings, as well as taking care of small problems before they become big problems. Increased delays and making problems more complicated are major problems with asynchronous communication. Espinosa and Pickering (2006, p. 1) argued the following in terms of the impact of temporal distribution on distributed teams:

With today's availability of sophisticated collaboration technology and team's increasing experience working globally, geographic distance is no longer a substantial problem for teams as it used to be, whereas time separation remains a great challenge, particularly as more time zones are represented in a team.

Although this statement is potentially true of all types of globally distributed teams, it may be especially true for globally distributed agile teams. Carmel and Agarwal (2001) noted, however, that reducing temporal distance via synchronous communication is "no panacea", because it too has its own limitations.

Finally, integrating globally distributed teams into a coherent team and instilling a sense of "teamness" is a challenging endeavor (Battin, et al., 2001; Carmel, 1999; Herbsleb & Mockus, 2003). Allowing the team to formulate its own set of core norms rather than strictly imposing organizational policy and procedure may contribute to the development of this sense of "teamness". Due to inherent challenges of communication, culture, and conflict within a distributed team, the freedom to develop strategies for addressing these types of issues helps to build cohesion and oneness.

Principle 6 - At Regular Intervals, the Team Reflects on How to Become More Effective, then Tunes and Adjusts its Behavior Accordingly

The existing literature identifies regular feedback as a crucial aspect of a successful global agile team configuration (e.g., Fowler, 2006; Layman, et al., 2006; Nisar & Hameed, 2004; Ramesh, et al., 2006; Sepulveda, 2003; Yap, 2005). Multiple agile practices such as daily stand-up meetings, iteration planning sessions, iteration demos, and iteration retrospectives can potentially contribute to the feedback loop in a very positive manner by providing the members with a constant stream of interaction with their colleagues and allowing for consistent feedback on how the project is progressing. Because of the emphasis on regular and effective communication in agile methodologies regular feedback can greatly enhance the agility of the team. A primary benefit is that global agile teams are not communicating every few weeks, but more likely every day. While it is true that the use of multiple ICT can help facilitate these practices for providing a mechanism for reflection and adjustment, the configuration of the team must again take into consideration the issue of temporal distribution. When there is a lack of overlapping work hours to due time zone differences the use of ICT is greatly hindered. If there is not someone on the other end of the communication to receive it, these technologies become useless for synchronous communication and cannot remedy the challenge of significant temporal distribution (Espinosa & Pickering, 2006). This suggests that teams should not simply be configured based upon the lowest cost locations, but also by considering the time zone differences. Moreover, by emphasizing the practices of short iterations and small releases the team is able to see each of the smaller pieces as they come along as providing a sense of the overall project purpose and goal, i.e., the "big picture".

Principle 7 - Deliver Working Software Frequently, from a Couple of Weeks to a Couple of Months, with a Preference to the Shorter Timescale

Even within global agile teams iterations can be scheduled in terms of weeks rather than in months and allow the customer to take an active and regular part in the design of the software. Through the use of multiple types of ICT, the customer can then see the progress of the project early and often and decide whether or not changes need to be made. Similarly, with small releases, the customer can actually see how parts of the project are functioning much sooner than with a traditional waterfall approach. Short iterations, small releases, and iteration demos have the potential to keep the customer satisfied during the entire project development lifecycle due to their regular involvement and ability to track the progress from requirements gathering to implementation. By carefully structuring the task design and the team processes, the team increases its ability to deliver software on a shorter timescale, thus decreasing the boundaries between it and its customer.

Principle 8 - Simplicity—The Art of Maximizing the Amount of Work Not Done—is Essential

Although this principle received little support in Bose's (2008) study in that only one case supported it, while the other cases indicated insufficient or no information, we argue that it should be included for

consideration when global teams are configured. We propose that if the simplicity principle were enacted, there would be less boundary and temporal issues because there would be less work done and therefore fewer misunderstandings due to cultural issues and less work to coordinate across multiple time zones.

Principle 9 - Welcome Changing Requirements, Even Late in Development. Agile Processes Harness Change for the Customer's Competitive Advantage

Bose (2008) found that this principle was not evidenced in any of the case studies, either because of insufficient or no information. We therefore omit it from our framework, but suggest that future empirical research validate its omission.

Principle 10 - Working Software is the Primary Measure of Progress; Principle 11 - Continuous Attention to Technical Excellence and Good Design Enhances Agility; Principle 12 - Agile Processes Promote Sustainable Development. The Sponsors, Developers, and Users Should be Able to Maintain a Constant Pace Indefinitely

Although Bose (2008) found at least some support for principles 10-12 we argue that they are true for colocated as well globally distributed teams. While these principles are important to successful implementation, they are broad in scope. For example, principle 10 states that working software is the primary measure of progress. If agile principles were followed, this principle would be true in both a local-only environment as well as in a globally distributed environment; likewise, for principles 11 and 12.

FUTURE RESEARCH

Organizations are concerned with delivering software in a timely, cost-effective fashion. Agile methods have been proposed to meet the on-time, within budget dilemma. Organizations have been concerned with leveraging global assets, but there is still considerable concern as to whether agile methods can be adapted to the global environment. There appears to be a growing interest among the academic community in regard to the area of agility and distributed work (e.g., Agerfalk & Fitzgerald, 2006; Flor, 2006; Holmstrom, et al., 2006; Ramesh, et al., 2006; Sarker & Sarker, 2009; Sarker, et al., 2009). This paper should be of interest to organizations and academicians alike.

Our framework provides practical insights for challenging the way organizations think about configuring global agile teams. In this way, this framework can be used to potentially alter the way in which team structuring decisions are made. As organizations become more attentive to leveraging global assets, the topic of global development is of contemporary interest.

Based on prior literature, our framework includes three dimensions: agility, virtualness, and team structure. It is possible that other dimensions may exist that we have not included in our framework. Future empirical research should explore these dimensions, subdimensions, and the resulting configurational interactions. Case study research would be appropriate to provide rich descriptive data about best practices, pitfalls and successes experienced by organizations that have already begun to venture down this path.

CONCLUSION

Based upon the literature exploring agility, virtualness, and team structure it is the belief of the authors that agile methodologies can be successfully applied in global software development projects. If research on the configuration of global agile teams is not conducted, researchers and practitioners alike will not have a clear understanding if there are truly significant differences between how agile teams and non-agile teams are designed in globally distributed settings. As such, we believe that this paper contributes to the information systems field by providing a framework based on extant literature that indicates these three major dimensions are appropriate. Furthermore, our research shows that particular intersections of the framework are especially important when configuring agile teams in a global environment. As of this time no known research framework exists which incorporates these three dimensions. The utilization of

global agile teams has the potential to significantly impact the field of software development. Our hope is that this framework will serve as a building block for further research in this important area.

REFERENCES

- Abrahamson, P., Warsta, J., Sippon, S.T., & Ronkainen, J. (2003). *New directions on agile methods: A Comparative Analysis*. Proceedings of the 25th International Conference on Software Engineering, pp. 244-254.
- Agerfalk, P.J., & Fitzgerald, B. (2006). Flexible and distributed software processes: Old petunias in new bowls?. *Communications of the ACM*, 49(10), 27-34.
- Battin, R.D., Crocker, R., Kreidler, J., & Subramanian, K. (2001). Leveraging resources in global software development. *IEEE Software*, 18(2), 70-77.
- Bell, B.S., & Kozlowski, S.W.J. (2002). A typology of virtual teams: Implications for effective Leadership. *Group & Organization Management*, 27(1) 14-49.
- Bose, I. (2008). Lessons learned from distributed agile software projects: A case-based analysis. *Communications of the Association for Information Systems*, 23(34), 619-632.
- Carmel, E. (1999). *Global software teams: Collaborating across borders and time zones*. Upper Saddle River, NJ: Prentice-Hall.
- Carmel, E., & Agarwal, R. (2001). Tactical approaches for alleviating distance in global software development. *IEEE Software*, 18(2), 22-29.
- Damian, D., & Moitra, D. (2006). Global software development: How far have we come? *IEEE Software*, 23(5), 17-19.
- Espinosa, J.A., & Pickering, C. (2006). *The effect of time separation on coordination processes and outcomes: a case study*, Proceedings of the 39th Hawaii International Conference on System Sciences, pp. 1-10.
- Evaristo, R.J., Scudder, R., & Desouza, K.C. (2004). A dimensional analysis of geographically distributed project teams: A case study. *Journal of Engineering Technology Management*, 21(3), 175-189.
- Fitzgerald, B., Hartnett, G., & Conboy, K. (2006). Customising agile methods to software practices at Intel Shannon. *European Journal of Information Systems*, 15(2), 200-213.
- Flor, N.V. (2006). Globally distributed software development and pair programming. *Communications of the ACM*, 49(10), 57-58.
- Fowler, M. (2006). Using agile software process with offshore development. Retrieved from <http://martinfowler.com/articles/agileOffshore.html>
- Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Agilemanifesto.org*. Retrieved April 27, 2006, from <http://www.agilemanifesto.org>

- Gibson, C.B., & Cohen, S.G. (2003). In the beginning: Introduction and framework. Gibson, C.B., and Cohen, S.G. (eds.), *Virtual teams that work: Creating conditions for virtual team effectiveness*, San Francisco, CA: Jossey-Bass.
- Hackman, J.R. (2002). *Leading teams: Setting the stage for great performances*. Boston, MA: Harvard Business School Press.
- Hackman, J.R., & Walton, R.E. (1986). Leading groups in organizations. P. S. Goodman, eds. *Designing effective work groups*. San Francisco, CA: Jossey-Bass.
- Hackman, J.R., & Oldham, G.R. (1980). *Work redesign*. Reading, MA: Addison-Wesley.
- Hanisch, J., & Corbitt, B. (2007). Impediments to requirements engineering during global software development. *European Journal of Information Systems*, 16(6), 793-805.
- Hanisch, J., Thanasankit, T., & Corbitt, B. (2001). Exploring the cultural and social impacts on the requirements engineering process - highlighting some problems challenging virtual team relationships with clients. *Journal of Systems and Information Technology*, 5(2), 1-19.
- Herbsleb, J.D., & Mockus, A. (2003). An empirical study of speed and communication in globally distributed software development. *IEEE Transactions on Software Engineering*, 29(6), 481-494.
- Herbsleb, J.D., & Moitra, D. (2001). Global software development. *IEEE Software*, 18(2), 16-20.
- Highsmith, J., & Cockburn, A. (2001). Agile software development: The business of innovation. *IEEE Computer*, 34(9), 120-122.
- Holmstrom, H., Fitzgerald, B., Agerfalk, P.J., & Conchuir, E.O. (2006). Agile practices reduce distance in global software development. *Information Systems Management*, 23(3), 7-18.
- Kircher, M., Prashant, J., Corsaro, A., & Levine, D. (2001). Distributed eXtreme programming.
- Marchesi, M., Succi, G., Wells, D., Williams, L., & Wells, J.D., eds. *Extreme Programming Perspectives*. Addison-Wesley, Reading, MA. Retrieved September 18, 2006, from <http://www.kircher-schwanninger.de/michael/publications/xp2001.pdf>
- Kirkman, B.L., Rosen, B., Gibson, C B., Tesluk, P., & McPherson, S. (2002). Five challenges to virtual team success: Lessons from Sabre, Inc. *Academy of Management Executive*, 16(3), 67-79.
- Kotlarsky, J., & Oshri, I. (2005). Social ties, knowledge sharing and successful collaboration in globally distributed system development projects. *European Journal of Information Systems*, 14(1), 37-48.
- Kotlarsky, J., van Fenema, P.C., & Willcocks, L.P. (2008). Developing a knowledge-based perspective on coordination: The case of global software projects. *Information & Management*, 45(2), 96-108.
- Komi-Sirvio, S., & Tihinen, M. (2005). Lessons learned by participants of distributed software development. *Knowledge and Process Management*, 12(2), 108-122.
- Layman, L., Williams, L., Damian, D., & Bures, H. (2006). Essential communication practices for Extreme Programming in a global software development team. *Information and Software Technology*, 48(9), 781-794.

- Lee, G., Delone, W., & Espinosa, J.A. (2006). Ambidextrous coping strategies in globally distributed software development projects. *Communications of the ACM*, 49(10), 35-40.
- Lee, G., & Xia, W. (2010). Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility. *MIS Quarterly*, 34(1), 87-114.
- Lurey, J.S., & Raisinghani, M.S. (2001). An empirical study of best practices in virtual teams. *Information & Management*, 38(8), 523-544.
- Meyer, A.D., Tsui, A.S., & Hinings, C.R. (1993). Configurational approaches to organizational analysis. *Academy of Management Journal*, 36(6), 1175-1195.
- Miles, R.E., & Snow, C.C. (1978). *Organizational strategy, structure, and process*. New York: McGraw-Hill.
- Mintzberg, H.T. (1983). *Structure in fives: Designing effective organizations*. Englewood Cliffs, NJ: Prentice-Hall.
- Mintzberg, H.T. (1979). *The structuring of organizations*. Englewood Cliffs, NJ: Prentice-Hall.
- Mitchell, A., & Zigurs, I. (2009). Trust in virtual teams: Solved or still a mystery? *The DATA BASE for Advances in Information Systems*, 40(3), 61-83.
- Nisar, M.F., & Hameed, T. (2004). *Agile methods handling offshore software development issues*. Proceedings of the 8th International Multitopic Conference, pp. 417-422.
- Ocker, R.J., & Fjermestad, J. (2008). Communication differences in virtual design teams: Findings from a multi-method analysis of high and low performing experimental teams. *The DATA BASE for Advances in Information Systems*, 39(1), 51-67.
- Powell, A., Piccoli, G., & Ives, B. (2004). Virtual teams: A review of current literature and directions for future research. *The DATA BASE for Advances in Information Systems*, 35(1), 6-36.
- Prasad, K., & Akhilesh, K.B. (2002). Global virtual teams: What impacts their design and Performance? *Team Performance Management*, 8(5/6), 102-112.
- Ramesh, B., Cao, L., Mohan, K., & Xu, P. (2006). Can distributed software development be agile? *Communications of the ACM*, 49(10), 41-46.
- Rennecker, J., & Goodwin, L. (2005). Delays and interruptions: A self-perpetuating paradox of communication and technology use. *Information and Organization*, 15, 247-266.
- Sarker, S., Lau, F., & Sahay, S. (2001). Using an adapted grounded theory approach for inductive theory building about virtual team development. *DATA BASE for Advances in Information Systems*, 32(1), 38-56.
- Sarker, S., Munson, C.L., Sarker, S., & Chakraborty, S. (2009). Assessing the relative contribution of the facets of agility to distributed systems development success: An Analytic Hierarchy Process approach. *Journal of Information Systems*, 18, pp. 285-299.

- Sarker, S., & Sahay, S. (2004). Implications of space and time for distributed work: An interpretive study of US-Norwegian systems development teams. *European Journal of Information Systems*, 13(1), 3-20.
- Sarker, S., & Sarker, S. (2009). Exploring agility in distributed information systems development teams: An interpretive study in an offshoring context. *Information Systems Research*, 20(3), 440-461.
- Sepulveda, C. (2003). *Agile development and remote teams: Learning to love the phone*. Proceedings of the Agile Development Conference, pp. 1-6.
- Schummer, T., & Schummer, J. (2001). Support for distributed teams in eXtreme programming. Succi, G., Marchesi, M., eds. *Extreme Programming Examined*, Reading, MA: Addison-Wesley.
- Shachaf, P. (2008). Cultural diversity and information and communication technology impacts on global virtual teams: An exploratory study. *Information & Management*, (45)2, 131-142.
- Suchan, J., & Hayzak, G. (2001). The communication characteristics of virtual teams: A case study. *IEEE Transactions on Professional Communication*, 44(3) 174-186.
- Xiaohu, Y., Bin, X., Zhijun, X.H., & Maddineni, S.R. (2004). *Extreme programming in global software development*. Canadian Conference on Electrical and Computer Engineering, pp. 1845-1848.
- Yap, M. (2005). *Follow the sun: Distributed Extreme Programming development*. Proceedings of the Agile Development Conference, pp. 218-224.

APPENDIX

TABLE 2
AGILE PRINCIPLES AND DIMENSIONS UNIQUE TO GLOBAL AGILE TEAMS

Agile Principle	Virtual Challenge		Structure Configuration Characteristic
	Boundary Spanning	Temporal Distribution	
1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.	X	X	<ul style="list-style-type: none"> • Team Processes • Core Norms • Team Composition • Task Design
2. Business people and developers must work together daily throughout the project.		X	<ul style="list-style-type: none"> • Team Processes • Core Norms • Team Composition
3. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	X	X	<ul style="list-style-type: none"> • Team Processes • Core Norms • Team Composition
4. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.		X	<ul style="list-style-type: none"> • Task Design
5. The best architectures, requirements, and designs emerge from self-organizing teams.	X	X	<ul style="list-style-type: none"> • Team Processes • Core Norms
6. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.		X	<ul style="list-style-type: none"> • Team Processes • Core Norms
7. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale	X		<ul style="list-style-type: none"> • Team Processes • Task Design
8. Simplicity – the art of maximizing the amount of work not done – is essential	X	X	<ul style="list-style-type: none"> • Team Processes • Core Norms • Task Design
9. Welcome changing requirements, even late in the development. Agile processes harness change for the customer's competitive advantage	-	-	-
10. Working software is the primary measure of progress	-	-	-
11. Continuous attention to technical excellence and good design enhances agility	-	-	-
12. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace	-	-	-