Issues and Strategy for Agile Global Software Development Adoption

FLORIN DUMITRIU  
Faculty of Economics and Business Administration  
Alexandru Ioan Cuza University  
22, Bvd. Carol I, Iasi  
ROMANIA  
fdumi@uaic.ro

DUMITRU OPREA  
Faculty of Economics and Business Administration  
Alexandru Ioan Cuza University  
22, Bvd. Carol I, Iasi  
ROMANIA  
doprea@uaic.ro

GABRIELA MESNITA  
Faculty of Economics and Business Administration  
Alexandru Ioan Cuza University  
22, Bvd. Carol I, Iasi  
ROMANIA  
gabim@uaic.ro

Abstract: - The necessity of finding right skilled people and limiting costs has leaded to wide adoption of global development in software industry. Agile method increases software quality and improves project performance and, for these reasons, in the last years many companies try to merge the two approaches. Despite many benefits, this combination raises various challenges. This paper explores the challenges of integrating the two approaches based on their conflicting characteristics with the aim to define a strategy for companies that wish to adopt agile global software development. The strategy we propose is founded on grouping the various solutions proposed by the literature in three categories: preparatory measures, technology and communication tools, tailoring agile method.

Key-Words: - global software development, agile software development, distributed XP, Scrum, collaboration technology, informal communication

1 Introduction

Global Software Development (GSD) and Agile Software Development (ASD) are two of the most prominent trends in the software industry in last decade. GSD is amplified by the globalization of IT and the improvement of telecommunication facilities. Within this trend, software development is increasingly going global, with development occurring in multiple sites located in different geographic regions. ASD is a reaction to too many failed projects and to rapid changes in the modern business and information technology domain. Both of them have benefits and challenges and in the last years there is a growing interest in combining them in order to mitigate the issues of global projects. This combination is often referred in the literature as Agile Global Software Development (AGSD).

Despite of the multiple benefits AGSD promises, their achievement is not easy due to the significant challenges it encounters. The most papers in the literature, both research and case study papers, treats the issues of merging GSD and ASD from one of two perspectives: the use of agile practices to mitigate global projects challenges and the adaptation of agile practices to cope with global conditions. In this context, our paper attempt to provide a way of applying AGSD so that to gain the potential benefits without the need of tailoring agile practices as much as possible.

The aim of our paper is to define a strategy that companies might use when to apply agile methods...
2 Agile global software development challenges

In this section we try to identify and categorize the challenges of merging agile and global development with the aim of explaining the reason for these. Before this we will shortly discuss global development issues and agile development characteristics.

2.1 Global software development issues

There are many reasons for the large adoption of GSD in the last years [13]: overcoming the shortage of IT professionals, the low cost opportunities of foreign software houses, global competition imposes businesses to be proactive and to take advantage of mergers, acquisitions and alliances, the advantages of proximity to the international market, improve time-to-market by using “round-the-clock” development.

The achievement of these benefits is not easy as there are diverse challenges that have to be deal within GSD environment. The main challenges that arise in practice refer to [13], [15]: inadequate communication and coordination between team members separated by distance and time-zone; the socio-cultural characteristics of particular location may affect people and this may condition their belief systems, reflect in their attitudes, and subsequently engender conflicts; lack of trust between remote team members that restrict communication; the distance between development team and business representatives may reduces the motivation of the development team, since they have no personal relationship to build on; lack of informal communication; loss of group awareness, which is useful for coordinating actions, managing coupling, discussing tasks, anticipating others’ actions, and finding help. To be successful, companies need to make fundamental changes on their approach of software development in order to support the GSD. One of such change might be the shifting to agile development.

The challenges of global projects are introduced by some features which distinguish GSD from traditional, centralized development. The most studies group the GSD characteristics in three categories of determinants [1]: geographical distance introduces the physical separation among team members, the software being developed in a multi-site environment; socio-cultural difference is a complex factor, involving organizational culture, national culture and language, politics, individual motivations and work ethics; temporal distance concerns the dislocation in time experienced by two members of a distributed team wishing to interact and it can be caused by time zone or time shifting work patterns and both of them can increase or decrease temporal distance.

2.2 Agile software development characteristics

Agile approach refers to a group of software development methodologies aiming to more flexible and lighter process development and to increase the frequency of feature iterations, making them more responsible to change. It is built around empowered and self-organizing teams that coordinate their work themselves, with a strong focus on collaboration.

Some of the agile key points, relevant to global projects, are: close collaboration among the members of the development team, based on face-to-face, informal communication; self-organizing teams are valued as a means to deliver better designs; frequent delivery of working software at short, regular intervals; on-site customer, facilitating close, daily cooperation between customer and developers; collaborative decision making; project are built around motivated individuals, who should be trusted, with limited control from management; regular feedback from customer; continuous planning with small timeboxes.

The software development activities are supported through various agile practices, including pair programming, test-driven development, continuous integration, planning game, collective ownership, and customer collaboration. In fact, an agile methodology is seen as a set of interdependent
practices, the team being responsible of selecting which of them to apply.

2.3 Conflicting nature of agile and global software development

Addressing the collaboration, coordination and communication challenges is the primary reason for the growing interest in exploring the applicability of agile in GSD settings. Despite of the common declared objectives, looking at the values and principles of ASD and the environment of global projects, the two approaches seem to be opposite extreme on a continuum. Thus, anyone who tries to investigate this subject has to start by understanding the fundamental differences between the two approaches. Table 1 summarizes the conflicting characteristics of GSD and ASD. Next, we will briefly discuss each one with the aim at point out the challenges of combining them.

Table 1 The contradictory characteristics of agile and global software development

<table>
<thead>
<tr>
<th>Project Environment Category</th>
<th>Global Development Characteristic</th>
<th>Agile Development Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Development Team</td>
<td>Team structure</td>
<td>Self-organizing</td>
</tr>
<tr>
<td></td>
<td>Communication frequency</td>
<td>Only when necessary</td>
</tr>
<tr>
<td></td>
<td>Communication style</td>
<td>Face-to-Face</td>
</tr>
<tr>
<td></td>
<td>Synchronization</td>
<td>Synchronous</td>
</tr>
<tr>
<td></td>
<td>Communication cardinality</td>
<td>Formal</td>
</tr>
<tr>
<td></td>
<td>Management culture</td>
<td>Command</td>
</tr>
<tr>
<td>Project Management</td>
<td>Decision making</td>
<td>Centralized</td>
</tr>
<tr>
<td></td>
<td>Coordination mechanisms</td>
<td>Plan-driven</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
<td>Up front</td>
</tr>
<tr>
<td></td>
<td>Process</td>
<td>Disciplined</td>
</tr>
<tr>
<td>The Customer</td>
<td>Involvement</td>
<td>During</td>
</tr>
<tr>
<td></td>
<td>Availability</td>
<td>Easily</td>
</tr>
<tr>
<td></td>
<td></td>
<td>analysis</td>
</tr>
</tbody>
</table>

**The Development Team**

Communication is the main issue in GSD settings due to geographical, temporal and cultural distance. For agile teams, communication is one of the fundamental values and they consider that the most efficient and effective method of conveying information to and within a development team is face-to-face conversation. This is not possible in global teams where their members are distributed and the communication between on-site and offshore teams is possible only through technology mediation. Another limiting factor may be the time difference due to different time zones that hinders synchronous communication even with the usage of telephones or video conferencing systems.

The geographical and temporal distance prohibits informal communication \[6\]. In collocated projects, informal communication helps easy dissemination of project knowledge, familiarizes individuals with the working styles of others, and fosters greater understanding among team members. It has been argued that informal communication plays a critical role in coordination activities for collocated software development, especially when size and complexity of development increase \[9\]. The lack of informal communication may lead to a lower team spirit \[12\].

Another conflicting characteristic is the team structure. The global development is based upon the condition of working with specialists and this will lead to an architecture which will start reflecting the team’s geographical distribution. Thus it will become difficult to complete the features within an iteration as different locations will become overspecialized in a particular component. Within agile projects the solution is to use a self-organizing team.

**Project Management**

Global process presents considerably more challenges than a collocated process because of the loss of the opportunities for rich interaction and informal communication among members of distributed team. Due to these challenges, some authors suggest the need of using a heavy, plan driven process as a way to coordinate GSD projects in order to have a significant payoff in terms of success \[3\], \[4\]. Greater discipline may compensate for the loss of informal communication. A key enabler is a consistent set of processes and standards. As software development processes cross locations, time zones and socio-cultural differences, the practices and methods must be consistent and the standards for each work result explicit in order to reduce the variability of software development task outputs.

Embracing such opinion leads to adopt software process frameworks like Software Engineering Institute’s Capability Maturity Model Integration (CMMI) and not to incorporate agile in GSD projects. Moreover, some specialists are concerned
with enriching the CMMI framework with KPAs for globally distributed development [11].

Unlike plan-based methods, agile methods deal with unpredictability by relying on people and their creativity rather than formalized processes [2]. Agile methods operate on the principle of “just enough method” because they seek to avoid prescribing cumbersome and time-consuming processes that add little value to the software product and increase the time of development process.

As a consequence of the need for heavy process and work product standardization, global managers tend to rely on plan-driven coordination mechanisms, in which the life cycle model specify the task to be performed and the desired outcomes of each project phase. Also, managers are required to perform detailed upfront planning and check adherence to the process through supervision. On the other side, agile approach states the priority of responding to change over following a plan as a fundamental value in software development. This agile value is difficult to apply in GSD settings as it is not easy to implement feedback mechanisms in an environment with geographical and temporal distances. All agile methodologies have built-in process to change their plans at regular intervals based on feedback from the customer. Such an approach requires replacing up-front planning with continuous one with small timeboxing.

**Customer**

Agile approach requires the involvement of customer throughout the lifecycle of the project and easy accessible to team members. The customer involvement is necessary to acquire feedback on the current implementation of the system and to further clarify the requirements of the system to be built. In GSD settings is nearly impossible to apply customer on-site, which may lead, at least, to inconsistencies in dealing with requirements engineering, inadequate quality assurance of working software and low credibility and friendly atmosphere between customer and distributed team.

3 **Solution for applying AGSD**

Despite the success of agile software development, its well achieved application in GSD is hard to accomplish. The differences between agile and global development rise serious challenges in combining them. All these issues might suggest that the application of agile methods in GSD settings is doomed to fail, especially when team size is large [8].

Beyond the benefits of applying AGSD, a more critical subject is how these benefits can be reached. That is, what are the practical solutions that mitigate the challenges of AGSD and will make possible to obtain the supposed benefits?

One solution is the use of information technology and communication tools in order to compensate the lack of co-location and face-to-face interaction. With the aim of improving both formal and informal communication among distributed workers, communication technologies such as video, audio, chat, instant messaging play an important role in creating a collaborative virtual environment. Large number of vendors offers ALM (Application Lifecycle Management) solutions which incorporate facilities for distributed work and virtual teams. Also, Web 2.0 tools such as social networks, blogs, wikis, folksonomy and mashup services have a great potential which it has to be exploited.

However, there are issues that cannot be solved through communication tools and technology. For example, it is widely believed that trust needs touch. Additionally, we need not to forget that the use of technology and tools may alter the agile spirit.

Another category of solution to AGSD challenges encounters the need to tailor or expand agile method in order to overcome the geographical, temporal and socio-cultural barriers of globally distributed software development. For example, some authors recommend Distributed eXtreme Programming by implementing partially or to varying degrees the values, principles and practices of traditional XP [10]. They suggest that if some practice could not be applied it does not prevent the application of others. Some experiences report that peer programming works satisfactorily in distributed teams, even across different time zones [7].

Other papers provide solution for customizing the Scrum methodology in distributed projects. For example, Smiths suggests using nested Scrum with multilevel reporting and multiple, daily Scrum of Scrums, meetings led by a senior ScrumMaster [14]. This practice makes Scrum methodology to work even within different time zones environment.

In our opinion, the tailoring agile practices solutions have to be treated with some prudence for two reasons. Firstly, some agilists advocate that agile methods have to be used as a whole due to the synergetic relationship among core practices which guarantee the maximum benefits. It is true that such an attitude tend to change in the last years. Secondly, the most experiences reported in the literature do not present the characteristics and condition of projects and these papers will not help
too much the practitioners who intent to apply AGSD.

4 Strategic approach of AGSD adoption

To be successful in coping with challenges of AGSD implementation, we recommend to take into account three categories of potential measures: preparatory measures, which aim to reduce geographical, temporal and socio-cultural distance specific to GSD; technological measures, with the scope of reducing the three kinds of distance by using information technology and communication tools; tailoring measures, which intent to adapt agile practices in order to apply them with success in GSD settings. If the first two categories aim to reduce the three kinds of distance and don’t alter agile practices, the last one has as a scope to reduce the consequences of the existence of these distances or to help to cope with them.

Thus, the main objective of implementing agile methods in GSD has to be the reducing of consequences of the geographical, temporal and socio-cultural distance. A good premise of such an undertaking is to firstly focus on the measures that lead to reduce the three distances. When adopting AGSD, we recommend a strategy comprising of three steps:

1) Define and implement preparatory measures, in order to reduce the distances in GSD projects. Any of these measures will not affect the agile values and principles.

2) Use information technology and communication tools to improve communication, collaboration and coordination. It is recommended that this step to be carried out after the first step ending because, as Herbsleb suggests, it has to determine for one thing the level of coordination to be required among sites and after that to plan the technology to support this level of coordination [5]. The level of coordination can be accurately determined only after the preparatory measures were established.

3) Once this environment is created and only after that, identify and employ the practical solution of tailoring agile practices which will provide the optimal compromise between ASD and GSD. For example, increase the time for iterations or replace the collective code ownership with the class code ownership in the spirit of Feature-Driven Development methodology.

We have already briefly discussed about the use of communication tools and the subject of tailoring agile methods. Next, we will summarize the practical solution which may be included in preliminary measure category. These solutions are:

- Assign a representative for each remote team who will have communication as an explicitly duty to compensate the missed hallway communication.
- Bring project teams together in collocated meeting for the first few iterations. This will help not only to create strong relationships between team members across sites and build trusts, but it is also important because many key decisions are taken at the beginning of the project.
- Share the core agile practices employed across distributed team. It is important to have one person on the team who plays the coach role charged with explaining the underlying agile principle and guiding the distributed team members how to adapt and to apply their agile practices.
- Carry out high-level design and architectural design on-site as upfront pre-iterative activities as Scrum and Feature-Driven Development claim.
- Arrange the participation of domain experts from remote team at the customer on-site requirements definition and they will act as customer interface for remote team.
- Distribute the work across sites according to system features not based on team specialization, so that an iteration could be completed at one site.
- Create a global collaborative culture by transferring a manager to a remote site in order to educate the local team on the global processes and to act as a liaison with the home office.
- Give distributed team members some kind of cultural and language training according to the requirements of the project.
- Scale the large project by using a team of teams or other strategy. This measure must be combined with an upfront architectural design with the aim to break down the development work into smaller components such that the project will be organized as a collection of coordinated small teams. The agile value may be difficult to apply when dependencies between work items are not well suited with division of work in distributed team.
- Encourage professional accreditation of distributed team members which might increase
trust among peoples. They will know that their remote colleagues have the qualifications, certifications, ethical framework and disciplinary mechanisms to perform their duties on project.

5 Conclusion
Merging GSD and ASD provides some benefits but also rise many challenges. In order to tackle these challenges it must to understand them by reflecting upon conflicting nature of the two approaches. In this respect, our paper points out the contradictory characteristics of ASD and GSD by taking into account three categories of characteristics to compare them: the development team, project management and customer. To be successful with AGSD, we believe that it is necessary to group the various solutions in three categories: preparatory, technological and agile practice tailoring. We assume the companies which intent to adopt agile in global projects has to start with implementation of preparatory measure, followed by technological ones, and only after that applying the tailoring measures.

References: