




Mindtree

Welcome to possible

WHITE PAPER



Distributed agile and
offshoring – antagonism
or symbiosis?

Summary

Agile software development and the breed of agile methodologies (XP, SCRUM, DSDM, etc.) have gained popularity since 2001. Agile methodologies were primarily founded for software projects executed at a single location. Today, with many adopters and practitioners across the globe, agile methodologies are showing promising results in multi-site projects too. Offshore delivery models have been successful in application maintenance and enhancement projects for more than two decades. In the case of development projects, iterative lifecycle approaches are more widespread and acceptable than the classical waterfall approach in delivering results and ensuring customer satisfaction. Distributed agile software development involves software projects done by agile teams located across geographies. This paper presents Mindtree's point of view on distributed agile and offshoring.

Contents

What is distributed agile?.....	03
Global Software Engineering (GSE), distributed agile and offshoring	03
Challenges in distributed agile	03
Distributed agile and offshoring: antagonism or symbiosis?	04
Our capability	05
What next?	06

What is distributed agile?

Agile is all about delivering business value in short iterations at a sustainable pace, adapting to changing business needs. Agile software development focuses on early delivery of working software and considers working software as the primary measure of progress. It creates an environment that responds to change by being flexible and nimble. It discourages creation of extensive documents that do not add any value. Distributed agile software development and testing is simply applying agile principles and practices to software projects executed by distributed teams or teams located at different sites. These could be at two or more floors of the same building, different buildings, cities or countries across geographies and time zones.

Global Software Engineering (GSE), distributed agile and offshoring

Global Software Engineering (GSE) involves software engineering projects executed with virtual teams from different time zones and diverse cultures. In the past decade GSE has become popular and widespread due to factors such as optimal costs, availability of a skilled pool of resources and globalization trends like mergers and acquisitions. Distribution of teams becomes a complex issue when an increasing number of teams from different organizations participate in a project from different locations or sites as shown in fig. 01. Additional factors such as project complexity, lifecycle activities, pricing models and culture impact distributed teams.

Distributed agile is the application of agile methodologies such as XP, Scrum, DSDM or any other home grown methodology based on Agile Manifesto and Agile Principles, in GSE projects.

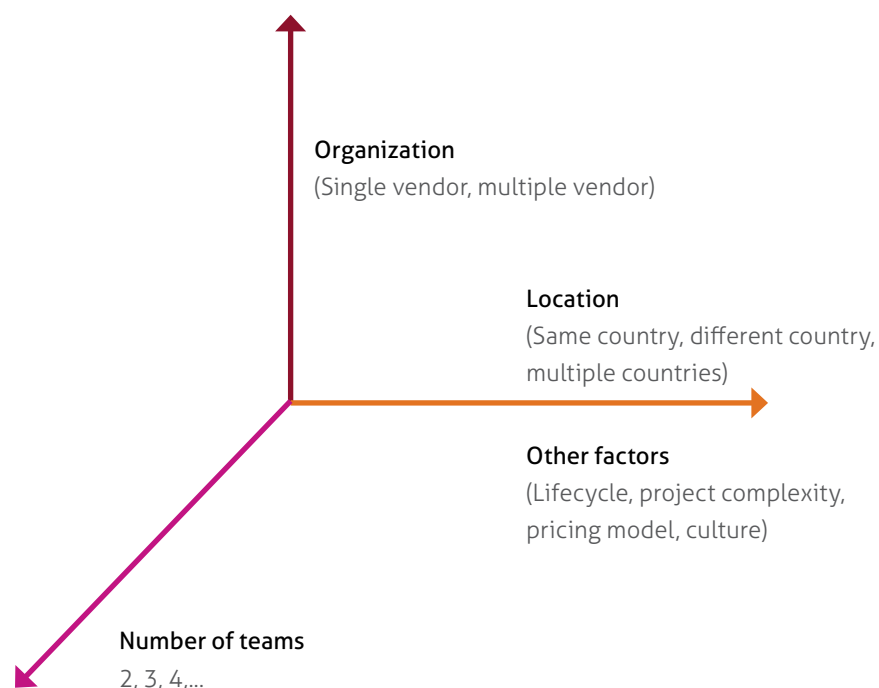
There are two variations of distributed agile in the offshoring context: Distributed agile in offshore insourcing is when two or more teams of a single organization from different countries participate in an agile project. A good example of this is when distributed agile teams involve teams from a large US based ISV and its captive center in another country.

Distributed agile in offshore outsourcing is when two or more teams of different organizations from different countries participate in an agile project. A good example of offshore outsourcing is when one or more teams from a service provider like Mindtree participate with the teams of a manufacturing firm or ISV. In this paper the term 'offshoring' means 'offshore outsourcing'.

Challenges in distributed agile

The vision of any GSE project is to deliver work products of high quality on schedule. This is done by engaging geographically distributed teams that define the right architecture for distributed development and following the right processes and tools for coordination and communication. There are several challenges encountered in areas such as requirement engineering,

Fig. 01: Distribution of teams



change management and project management, to name a few. Practitioners continue to adopt evolutionary methodologies such as agile software development in order to ensure success in distributed projects. Delivery of working software at regular intervals at a sustainable pace and responding to the changing needs of business users are the imperatives of agile. However, one must understand the challenges involved in distributed development. These challenges fall under three broad categories:

1. Communication and coordination: Collocated teams have the advantage of interacting with each other and with the onsite customer on a daily basis. This helps them understand and refine requirements in a timely manner. Geographically distributed teams on the other hand, do not get the opportunity of meeting face-to-face or having informal hallway or water-cooler discussions. They have to depend on a set of tools and processes to communicate and coordinate with each other. Also, in contrast to collocated teams, distributed teams need to practice just enough documentation in order to create and retain knowledge across teams.

2. Time zone differences: Time zone differences impact distributed teams. This impact can be both positive and negative, depending on the work culture and relationship among team members. For example, two teams from overlapping time zones with a 5-hour overlap may feel very comfortable working with each other. However, the more they utilize their time in communication and coordination, the less time they will get for other engineering activities. At the other extreme, if there are two teams with only an hour of overlap, they will have to stretch or work extra hours for an additional overlap, unless they are smart enough to use the one hour overlap efficiently. The way distributed teams manage this challenge depends on people, culture and leadership style.

3. People, culture and leadership style: There are two main factors that influence interaction between distributed teams. The first fundamental factor is the work style and culture of the locale or country. The second key factor is the organizational culture. Organizations new to agile methodologies or in the early stages of agile transformation find it very difficult to cope with these challenges. Whereas organizations that promote agile and have an established agile culture manage these challenges confidently and execute successful projects. Geographically distributed

teams need to understand and appreciate cultural differences and work together in ensuring harmony and rapport among all teams.

Distributed agile and offshoring: antagonism or symbiosis?

The success of distributed agile projects involving offshore teams depends on several factors: project type, volatility of requirements, distributed governance, skills and competency of team members and the culture of participating organizations. Those who have succeeded in distributed agile projects in the offshoring context agree on the symbiotic relationship between distributed agile and offshoring. Those who fail however, are curious to know exactly what practices help distributed projects succeed.

The organizational culture of participating organizations plays a major role in nurturing distributed agile ecosystems. A good mindshare, open communication and collaborative participation among organizations, results in a positive relationship that supports distributed agile projects. Without these elements, organizations continue to function with differences in understanding and expectations on distributed agile.

To ensure early success, starting with a non-critical pilot project or with simple user stories is critical for positive reinforcement amongst teams. Organizations wanting to develop modules involving critical functionalities or build mission-critical products or applications using distributed agile, must ensure prior experience in distributed agile in all participating teams.

There is a widespread misunderstanding in the industry that when requirements change on a daily basis, agile is the way to go. If requirements are changing so fast and if they are to be implemented instantaneously, it is good to have all the team members or most of the team members in one location. This is because highly volatile requirements warrant intense communication among team members and timely resolution of queries related to requirements. These are the symptoms of ad-hoc development, product envisioning or concept development. Such projects can be executed in an offshore model. However, these are not 'distributed agile' projects because agile methodologies require discipline and involve delivery of working software at a sustainable pace or through time-boxes.

Governance and senior management support is important in distributed agile ecosystems. Distributed agile projects expose risks at an early stage. Continuous improvement and course correction is possible only when project sponsors understand how distributed agile projects progress over initial iterations.

We have experienced the symbiotic relationship between distributed agile and offshoring and so have our customers.

Our capability

Mindtree has proven experience of over 10 years in agile methodologies. We have executed software development, maintenance and testing projects using agile methodologies with distributed teams as well as collocated offshore teams. Our experience includes:

- 50,00,000+ man hours of experience in agile projects
- 1000+ agile team members
- 100+ ongoing agile projects

We understand agile as well as other evolutionary methodologies. Our experience in executing projects in onsite-offshore models enables us to apply agile principles and best practices in distributed teams. In order to make this work at an organizational level, we have created subject matter experts and agile coaches. We also facilitate training programs on distributed agile to build agile capabilities in our organization. These are the crucial steps we adopted to make it work.

We promote agile through Mindtree Agile Council and Agile Community. Mindtree Agile Council, a team of agile experts, focuses on nurturing agile capabilities at Mindtree and facilitating competency building programs. Agile Community at Mindtree is a knowledge management community that provides a common platform for all practitioners and promotes knowledge sharing. In addition to these we contribute to external conferences and online media by sharing our success stories, presenting white papers and participating in discussions.

Case study 1

Distributed agile in product engineering

Mindtree implemented distributed agile to execute multiple projects for a leading global provider of security, storage and system management solutions. These projects involved product reengineering, maintenance and enhancement of multiple products based on technologies such as Java, C++,

Python, Perl, and C#. We supported these products on multiple platforms including Windows, Solaris, Linux and Mac.

In each project, we had multiple Scrum teams of 7 to 9 each. The Mindtree and customer teams followed Scrum for rapid development with frequent delivery and demonstration of working software in order to ensure customer satisfaction.

We selected a Sprint size of two weeks and ensured collective participation in Sprint planning. Our team participated in daily stand-up meetings and demonstrated working code at the end of Sprints. Using NUnit and CppUnit our team members practiced Test Driven Development (TDD). Also, we generated burn down / burn up charts for sprint planning and retrospectives.

With more than 8 years of partnership and 30% agile adoption in a team of 200 we have found immense benefits in implementing distributed agile practices in this engagement.

Case study 2

Agile transformation in geographically distributed teams

Mindtree implemented agile methodologies to execute multiple projects for one of the global leaders in automobile manufacturing and construction. Some of these projects involved a major refactoring or rewrite of one or more IT applications using technologies such as Java, C#, Oracle, SQL Server, BizTalk, WebSphere, FlexNet, SAP etc. The other projects involved application enhancement, change request management and support.

Mindtree set up a dedicated offshore development center at Bangalore with a team of 200 engineers for the rewrite, upgrade, enhancement and support of the existing portfolio of IT applications. We collaborated with our customer and decided to introduce agile methods in some of the key projects. To ensure success in this agile transformation, we initiated training programs on agile practices for all team members.

In one of the projects we implemented Scrum with four sub teams in order to address the major refactoring requirements in an IT application. We started with two week Sprints and practiced release planning, backlog grooming, daily stand-ups, reviews and retrospectives,

and continuous integration. After the initial months, we introduced Test Driven Development (TDD) to improve product quality.

Simultaneously, we introduced Scrum practices in a large maintenance project with Scrum of Scrums. We started with four week Sprints to match the rhythm of release cycles. In these projects we used tools such as Microsoft Visual Studio, Team Foundation Server, and NUnit.

Mindtree has been engaged with this customer for more than ten years. Considering the success and benefits of distributed agile, the number of distributed agile projects in this engagement is on the rise.

What next?

Mindtree has contributed to several success stories on distributed agile. Every year we see many new engagements kicking off in this model. We have experienced the symbiotic relationship between distributed agile and offshoring and so have our customers. We believe that distributed agile will be increasingly adopted over the coming years. For more information on Mindtree distributed agile capabilities or case studies please contact raja_bavani@mindtree.com.

References:

Mindtree distributed agile blogs:

<http://www.blogs.mindtree.com/author/raja-bavani>

Mindtree articles and white papers:

<http://www.mindtree.com/services/agile>

About the author:

Raja Bavani is Chief Architect of Mindtree's Product Engineering Services (PES) and IT Services (ITS) groups and plays the role of agile evangelist. He has more than 20 years of experience in the IT industry and has published papers at international conferences on topics related to code quality, distributed agile, customer value management and software estimation. He is a member of IEEE and IEEE computer society. He regularly interfaces with educational institutions to offer guest lectures and writes for technical conferences. He also writes for magazines such as Agile Record, Cutter IT Journal, IEEE Software and SD Times.

About Mindtree

Mindtree is a global information technology solutions company with revenues of over USD 400 million. Our team of 11,000 experts engineer meaningful technology solutions to help businesses and societies flourish. We enable our customers achieve competitive advantage through flexible and global delivery models, agile methodologies and expert frameworks.