

Applying Agile Lean to Global Software Development

Student ID

Date

Major Code:

Supervisor Name:

*A dissertation submitted in partial fulfillment
of the requirements for the degree of*

()

University Information



University Logo

Faculty Name

JKEssay

VX: ProWriter-1

Abstract

Although challenges of Global Software Development (GSD) are well known in the industry, practitioners and the organizations look for ways to improve results and overcome challenges. Companies have tried to implement many workable solutions possible to solve issues like poor communication, lack of trust, low morale and many other such issues prevalent in the distributed setting. With the success of agile, the methodology gained interest in leveraging its benefits to alleviate some of these challenges. Similarly, lean was also implemented in distributed software development to resolve issues. While each methodology provided some improvement in the results achieved in global software development, many issues persisted and the desired growth/results were not received. Recent years have seen an increased interest of applying a combination of agile and lean software development paradigms to resolve current industry issues in the area and meet the needs of rapid changing environment. This study aims to study the current practices of the combination of agile lean existing in the industry and how it can be utilized in the global software development. The study focuses on challenges faced to implement agile lean and successful sustainable implementation of agile lean in an environment of global software development.

Keywords:

Global software development, distributed software development, lean principles, lean software development, distributed lean software development, agile-lean, distributed Scrumban, distributed agile

Table of Contents

Abstract 2

 Keywords: 2

Preface 4

Introduction 5

Problem Statement and Justification 8

Literature Review – Analysis of Related Work 10

 Global Software Development 10

 Agile 12

 Lean 14

 Combining Agile and Lean 16

 Agile and Lean in Global Software Development 18

Method 21

Results or Findings 25

 Implementation of Agile Lean in Global Software Development 25

 Applying Agile Lean Principles in Global Software Development 28

 Challenges in Application 30

 Strategies for Adoption/Application 32

Discussion 35

Conclusion 37

References 38

Preface

This research was carried out at Harrisburg University mainly as a part of Master's in Project Management during the year of 2017. I am very grateful for Harrisburg University that gave me this opportunity to work on such an interesting topic to complete my research. It has been a great experience learning about the agile lean practices in global development context which I intend to use in my career.

Many people have helped me to conduct this research. I would like to mention my professor Dr. Thomas Sieves for his guidance and support throughout the process. I am thankful to him for helping me define the scope of the research, encouragement and constant review of the research to ensure I head on the right path. Dr. Sieves helped me frame the context of my thesis, evolve my topic into Agile Lean practices in global software development and has provided many valuable input throughout the writing that resulted into this document. I deeply appreciate his time and effort in this.

Finally, I would also like to thank all interviewees that helped me discuss my observations and reach on conclusion. Their valuable input has been documented and provided great insights to conclude the research. Without them the research would not have been validated. Thank you all. I am grateful to all of you.

Introduction

Lean thinking became popular after its huge success at manufacturing process at Toyota. Lean thinking brought forth principles of increasing customer value, eliminating waste, reinventing efficient value chain to deliver customer's wants and much more which will provide higher competitive advantage. Acquiring researchers' interest Lean has been successfully implemented to improve results across many disciplines. Within the IT industry itself, lean principles have helped in improving software development process, governance, scaling agile framework and much more.

In recent years, agile methods have been widely gaining appreciation in IT and non-IT projects. Agile has come to be known as a synonym for the ability to adapt frequent changes easily. Manifesto for Agile Software Development has introduced practices and values such as individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation and responding to change over following a plan which has helped in increased visibility in a project, increased team efficiency, enhanced collaboration, increased change management, improved performance, improved predictability, reduced risk and quick ability to adapt to frequent changes. Thus, improved results have been reported by several IT and non-IT projects where agile practices are implemented carefully. Although the experts have been discussing the impact that agile can have on a project, at the same time they also recommend not to implement agile methodology on large and distributed efforts.

Whereas on the other hand, global software development (GSD) has also become increasingly popular in the software development industry due to reduced costs, extending development hours and quicker time to market advantages. Global software development allows project teams to be

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

distributed across different remote sites, thus bringing in different geographical locations, various cultures, different time zones, varied languages, cultures, values and norms off behavior (Holmström, Ó Conchúir, Ågerfalk & Fitzgerald, 2006). With all the increased advantages and growing significance of distributed software development in IT industry, the challenges faced in this environment and the number of failures reported cannot be overlooked. Many significant challenges accompany distributed software development that need to be addressed to ensure that the project meets its intended objectives.

For the purposes of this thesis, development is broadly referenced to a software development lifecycle. The research will extensively study all phases of lifecycle and will not be restricted to a new software or product development.

With the increasing trend of lean and agile practices being implemented in the software development industry, researches have started looking into possibilities of adopting agile software development and lean principles in large distributed software development organizations. There has been an increased interest of utilizing a combined approach of agile and lean in software development and many practices currently exist in the organizations as they implement a combination of the two within their processes. Such practices have proved to provide successful results in the small collocated teams thus the practitioners now look toward agile lean as a possible solution to achieve better results in global software development. The project leaders and managers have been seeking methods where the easy management practices of that of agile and lean development can be adopted in a global software development environment to deliver increased performance, project visibility and desired quality. While the demand to implement agile lean in distributed development has been increasing to avail benefits of both in a project, there is no documentation or extensive research conducted on how to apply agile lean in large organizations

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

or distributed teams. Neither are there records on the successful implementation of agile lean in a global software development. In addition to all this, currently there is lack of knowledge among practitioners on how to combine lean and agile together to achieve better results in an organization specific context.

Combination of agile development and global software development have come to known as Distributed Agile Software Development in the IT industry (Alqahtani, Moore, Harrison & Wood, 2013). Many researchers have documented successes achieved by agile development methods in a distributed context, but there is a lack of research on where does lean fit into this picture of Distributed Agile Software Development. Therefore, the objective of the paper is to identify how agile practices and lean development paradigm can be combined together to benefit global software development. The study will focus on sustainable successful implementation of Agile Lean in a geographically distributed team.

As established by the researchers, agile practices if implemented as is or in other words directly out of the box in a distributed environment may not deliver expected results. Agile practices and methods need to be tailored to best meet the specific characteristics of the distributed software development. Various extensions need to be built into the agile procedures so that it can easily provide improved efficiency in a distributed project and help to overcome not only management challenges but also project specific challenges to an extent. Similarly, it requires special attention when it is combined with other practices such as lean to build these extensions. This paper will also identify strategies that can be used to implement agile lean practices that will help the project successfully meet its goals.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Problem Statement and Justification

Lean methodology and distributed agile software development has become a common trend in IT industry. With the growth of globalization, there has been an increasing need to adopt practices which will help increase collaboration and communication in a multi-location team and improve delivery. Lean principles can provide means to develop a process known as value chain that can lead to faster development, enhanced team collaboration, minimize waste, produce what customers' want and continuously work to improve the quality. Similarly agile processes also provide means to deliver right and fast to the customers with continuous feedback and constantly integrating their changing needs.

There has been some research on how to apply various agile methods such as Scrum and eXtreme Programming (XP) and Lean principles in distributed settings. Also, there has been researches on how distributed agile can be beneficial to large organizations and increase productivity. The results of these researches suggest that in order to achieve organization's goal of productivity and quality agile practices need to be evaluated, tailored or even combined with other approaches such as waterfall to suit a specific organizational context. Recently, combination of agile lean has gained interest to solve challenges of scaling agile in large organization or fills gaps within agile. Studies have identified need for rigorous research to identify combinations of agile lean in global software development.

While the research currently elaborates on the benefits that hybrid agile lean can provide to software development there is a lack of in depth knowledge on ways to successfully apply agile lean in GSD. Therefore, more research is required in the area how to successfully implement agile lean in a distributed team and provide efficiency. This study aims at address the research gap of combining agile lean in distributed contexts and successfully implementing it to overcome

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

challenges that prevail in distributed agile software development. The thesis will study how agile and lean are adopted separately or in a combined fashion in large and globally distributed software development organizations. The research questions that this study will address are as follows:

RQ1: What combinations of agile lean can be useful to global software development?

RQ2: Will agile lean practices be successful in global software development?

RQ3: How to scale agile lean development practices in a large distributed software development organization?

RQ4: Are there any challenges faced during implementation of agile lean in global software development?

RQ5: What strategies can be adopted to sustain and manage agile lean development practices in a global software development to avail its benefits?

Examples Provided by JK Essay

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Literature Review – Analysis of Related Work

According to the, Worldwide Offshore IT Services Forecast, 2016–2020 released in May 2016, suggests that the offshore IT services market is in its maturity phase with its gradually slowing growth rate. David Tapper, vice president for Outsourcing, Managed, and Offshore Services at IDC has advised that with the increasing pace of buyers in procuring services, offshore providers continue to invest in critical areas, particularly around supporting the transformation of firms to digital enterprises, incorporating increasing levels of automation, and forming new strategic partnerships to provide innovative solutions. In this era of increased offshore development, distributed development projects that consists of many teams working together to accomplish project goals from different geographic locations have become very common phenomenon.

Global Software Development

Following the trend of globalization, organizations of software industry have also been expanding globally to benefit from the larger pool of talent, resources and supplies, continuous development round-the-clock, cost savings etc. Herbsleb and Moitra identified the following factors that led the experiment of remotely located software development facilities to a rapidly growing trend in early 21st century:

- Successfully utilize scarce resources at lower costs wherever they may be located
- Avail advantages of being close to the local market
- Forming virtual corporations and teams rapidly to capture market opportunities
- Increase efficiency by reducing time to market by capitalizing on the time differences and support “round-the-clock” development
- Ability to exploit merger and acquisition opportunities at local market

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Today, working in a software development project that is geographically distributed has become a very common norm in software development industry (Herbsleb & Moitra, 2001). While establishing globally through partnerships, acquisitions, outsourcing or offshoring companies form distributed teams. Distributed teams with experts undertake the responsibility of software development work to be completed through coordination and collaboration while located in different geographical locations across nations. According to Richardson, “The term Global Software Development (GSD) implies teams of knowledge workers located in various parts of the globe developing commercially viable software. It is characterized by moving centralized software development from “home” to dispersed teams or/and external organizations in remote locations.” Global software development is also described by distributed software development (DSD) which also focuses on distributed development process (Richardson, 2007). Teams operating at multiple site across geographical and organizational boundaries are connected through the means of technology for communication and collaboration.

Literature on GSD identifies temporal distance, geographical distance and sociocultural distance as constraints in the field of distributed work (Holmström, Fitzgerald, Ågerfalk, & Conchúir, 2006). Their study included statements from HP officials stating how sometimes members can fall behind of the conversation, how there can a lag of few days in responses, feeling of being two different teams and language barrier can be frustrating and cause significant hindrance to the success of project. Asymmetric processes, policies and standards of various organizations and sites involved can also be a challenging roadblock in the development. Several such challenges that accompany global software development hinders communication and coordination in large teams thus resulting into delayed work, increased costs and unsatisfied customers. Drawbacks of GSD may at times outrun the advantages thus making it less attractive for organizations.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Agile

During the 1990s, agile methods have been gaining popularity in the software development field and have become highly prevalent after the formation of Agile Manifesto in 2001. Agile Manifesto states four values and twelve principles of agile software development. The core values are:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

- Agile Manifesto, 2001

Based on iterative and incremental development model as a response to changing needs and early expected deliveries, agile is characterized by orientation towards people, frequent customer collaboration, fast development cycles that are short and light, time-bound delivery, and inspect and adapt. There are many methods with each having its own practices and vocabulary under agile that share the same principles and philosophy. Twelve principles that were brought forth by the manifesto based on the core values described earlier are as follows:

1. *Customer Satisfaction*: Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. *Welcome Change*: Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. *Frequent Delivery*: Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. *Work Together*: Business people and developers must work together daily throughout the project.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

5. *Motivated Individual*: Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. *Face-to-face conversation*: The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. *Working Software*: Working software is the primary measure of progress.
8. *Sustainable pace*: Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. *Technical Excellence*: Continuous attention to technical excellence and good design enhances agility.
10. *Simplicity*: Simplicity is the art of maximizing the amount of work not done and is essential.
11. *Self-organizing team*: The best architectures, requirements, and designs emerge from self-organizing teams.
12. *Continuous inspect and adapt*: At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Despite of sharing same principles, agile does not share a universal definition. Agile methods are stated by researchers as a set of practices created by experienced software developers which constitutes multiple methods under its umbrella. Extreme Programming(XP), Scrum, Dynamic Systems Development Method (DSDM), Crystal Methods, Feature-Driven Development (FDD) and Adaptive Software Development. Each of the method loosely provide a framework to work with and are mostly open for implementation per organization needs. There was a time when Lean Software Development was considered a method under agile and many still do not distinguish among the two. Most literature however accept the difference between agile and Lean and motivate recent idea of industry to implement Lean practices in addition to XP or Scrum.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Lean

Initially originated in manufacturing unit of Toyota, Lean provided an efficient system of continuous flow for production. Working on the principle of increasing value to customer and eliminating waste, Lean is the result of applying them to the entire delivery process in any industry.

Womack and Jones described core principles of Lean as follows:

Value: The critical starting point of lean thinking is to specify value. Value can only be defined by a customer and is what customer pays for. Everything else is waste.

Value Stream: Next most important step in lean thinking is identifying the complete value delivery process also called value stream. All steps taken to deliver a product to customer from concept through design, production, launch and to delivery. The complete process should be inspected to eliminate waste and increase efficiency.

Flow: Once the value stream is identified along with the value to be delivered, next step is to create an optimized flow of work activities that create high value in the system. This step is taken to ensure that all wasteful activities that do not provide value to customer has been eliminated.

Pull: Production should occur when required to eliminate any unwanted inventory and its associated wastes. Pull principle is used to ensure that the value product is built by providing customer what they want.

Perfection: Most important principle of lean is to continuously improve. Waste tends to creep in the most optimized value delivery process in many forms, which should be avoided by continuously working on the previously improved processes. Organization should keep improving to find better ways to minimize waste and deliver more and more value to customer efficiently.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Microsoft reported lean mistake proofing of a software process eradicated whole classes of errors (Tierney, 1993). Gaining huge success at what and how Lean provides value in product and process development, it started attracting researchers' interests. Mary and Tom Poppendieck illustrated the application of lean principles and practices in Software development context. According to Poppendieck, lean principles that guide Lean Software development are as follows:

1. *Eliminate Waste*: Understanding what constitutes value and identifying wastes in the process that should be removed.
2. *Build Quality In*: Providing a product with high integrity and minimizing number of defects by testing as soon as possible.
3. *Amplify Learning*: Create processes and practices that support learning through rapid feedback or communication to share customer needs, architecture solutions, better strategies that can be deployed in the process and so forth.
4. *Defer Commitment*: Delay decision as late as possible till most of the information is available, especially for irreversible decisions and maintain many options.
5. *Deliver Fast*: Set up short release cycles and limit the number of features to be delivered.
6. *Respect People*: Facilitate leadership by trusting and respecting them, provide necessary means and expertise required for the tasks and aim for realistic goals.
7. *Optimize the whole*: Improve all processes in the identified value stream by implementing lean.

Using the lean manufacturing principles into a software development model, an organization or team can eliminate the waste in a system and provide greater value to the customer. Poppendieck et al suggests that “use of lean principles and concepts at product development and engineering can lead to process and quality improvements”. Good understanding of lean and its adoption can

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

optimize a process and provide efficient structure to it. As lean principles guide and not dictate specific practices, it can be implemented in multiple context.

Combining Agile and Lean

As stated earlier, for a long time Agile and Lean were considered as different terms for the same thing (Wang, Conboy & Cawley, 2011). Literature studies on software development practices categorized Lean Software development another methodology under the agile umbrella as their results shown that the industry did not distinguish between the two. However, many researchers recognize the difference between lean and agile and bring forth the approaches in which hybrid agile/lean is being implemented in the industry. Wang summarized his findings on the type of combination of agile and lean depending upon the perceived difference between the two in the following table:

Perception of the Difference between Agile and Lean	Combination Type
No perceived difference between agile and lean	The combination of both is non-purposeful
Agile and lean are at the different levels. Lean is thinking tool, agile is prescriptive practice	Use lean principles to guide the development and adaptation of agile practices
	Use lean principles to facilitate agile adoption
Agile and lean are at the similar levels, but have different scopes and focus	Top-down implementation of lean to create environment where bottom-up agile can thrive
	Use lean to help agile scale up
	Both can be combined at the principle level to develop and adapt practices
	Use lean techniques to improve agile software development processes
	Use agile practices to support lean software development processes

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Non-purposeful combination: Non-purposeful combination of lean was identified by Wang in literature where the researches use the word agile and lean parallelly without distinguishing much between the two. Reporter was expected to use a combination of two but did not treat them separately.

Lean facilitating agile development or adoption: Other studies in Wang research showed that the organization were using lean principles to guide implementation of agile practices. Organizations in the study were reported to use the lean principles to analyze either the practices being implanted or ways to improve efficiency while adopting agile practices for software development. This combination was observed to utilize lean principles mostly before or during agile initiatives of transition. Another combination that was identified by Wang was that organization used lean principles to build an environment and stream line their processes where they could adapt an agile methodology. For example, Capital One utilized multiple lean tools to evaluate their delivery process and streamline business values before transitioning to Scrum.

Agile and Lean both serving separate scopes together: One other successful combination identified at many organization was that an agile methodology implemented at the project level was coupled with lean principles to achieve overall business goals. Case in the study suggested that the using lean thinking at the business level helped identify the better business values to be delivered by the team. Also, lean thinking was helpful at DTE Energy to manage portfolio comprised of agile software projects. Another combination that was identified in the study was implementation of lean within agile. Under this practice, organizations were using lean concepts and tools such as “eliminate waste” or “Kanban” or “Kaizen” to improve agile processes continuously. Study also reported cases where agile was implemented within dominant lean practices to support the process.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Most common strategies being used in the industry is utilizing lean concepts to improve agile processes. Research depicted Kanban to be a popular tool being used alongside agile in the software development industry to “eliminate waste” from the processes. Lean and agile were also reported to be combined at the principle level to develop new concepts and practices at wireless Data Services Global that are most effective when lean and agile are applied together. Hybrid agile/lean have been gaining attention to be able to achieve high service level (result of agile) in an economic efficient way (result of lean). Coplien and Bjornwig stated that fundamental differences between lean and agile may be significant, but together they complement each other by addressing different components of systems development. Overall, to be able to make a judicious selection of the type of combination required at an organization can be attained by understanding the requirements and constraints of the business environments, challenges being addressed and the goal that the teams are trying to achieve.

Agile and Lean in Global Software Development

As discussed, the foundation of Agile software development is built upon the collocation and constant interaction among team members and the customers. Also, as it is known that global software development introduces physical distance, time zone difference and cultural distance and many communication barriers amongst team members. Both approaches are popular and have been adopted separately or in combination in the software industry. And despite the odds that the agile software methodology may not be a good fit in a global software development environment, many well-known software organizations have successfully implemented agile practices in multisite projects.

As evident successful agile implementation requires effective face-to-face communication which may be difficult to achieve with the distances in a global environment. However, research suggests

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

that there are several practices that an organization can utilize to overcome various challenges. Tools like telephone, teleconferencing, videoconferencing, email and instant messaging can help facilitate communication barriers in a multiple site project. Many other web tools like social networks, blogs, wikis, folksonomy, mashup and software tools like Application Lifecycle Management (ALM) provides platform for formal and informal communication. While tools and technology provides a platform for communication, in order to maintain the spirit of agile methodology team members are required to trust and understand each other. Behm reports that team gathering, exchange visits, informal meetings with off-site team members and gradual team distribution are a few ways to help team understand each member and remove some of the cultural or other barriers. Another research by Lane in 2013 suggested that impact of socio-cultural distances in a global setting can be removed using scrum techniques. Daily scrum interaction helps in building knowledge-based trust, planning and prioritization helps in identifying value in each other and the burndown chart with reviews help in building performance based trust amongst team members.

Kirscher et al. around 2001 brought forward the concept of Distributed eXtreme Programming (DXP) concept in their research which was an extension on traditional XP – an agile methodology. Research suggested that valuable contribution of DXP helped in the integration of remote team members into the development process. In a similar research, eXtreme Programming (XP) has been useful in increasing effective communication during the outsourcing of software development projects (Ngo-The et al., 2005). The study reported decrease in overtime, high morale among team members and strong customer commitments as the perceived benefits of the application of XP in GSD. Many organizations use Kanban boards and practices in conjunction with distributed agile teams to remove wastes and excess activities and improve communication. Combination of agile

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

and lean has been judiciously used in different cases of the companies to meet the goals in a distributed team environment.

Lean practices help in reducing time to market same as the agile paradigm with their iterative cycle and fast planning phase. It also ensures providing highest customer value and introducing transparency in development. Lean helps in focusing on reducing wastes and in turn decreased number of defects in the release. Lean was also observed to extend the agile elements by ensuring that everyone inside the project team or the organization understands customer value and providing deliverables that add customer value, instead of only product owner doing so as observed in agile. Lean practices such as eliminating waste and optimizing the whole by looking at the big picture are concepts that are new that cannot be found in agile development which makes their combination powerful enough to solve many challenges in global software development. In a study of Rodríguez et al Lean and Agile combination brought in continuous improvement and learning in the process. Retrospectives and Scrum Masters use various techniques to identify root cause of the issues, ways to promote learning within the team and visual flow to realize the value. Such processes in an organization ensure that the teams change their way of working on the basis of the retrospective results.

Examples Provided by JK Essay

Method

The analysis in the thesis presents an extensive research on successful agile lean methods and practices in distributed software development. Review of literature suggests agile methodologies have been widely used in the software development industry to increase productivity, quality and decrease risks. Recently lean-agile has been gaining interests of researchers to scale distributed software development at the organizational levels and beyond.

The main step of research will be conducting a systematic review of literature with the main steps of planning the review, conducting it and finally reporting the results of the review. Systematic literature review is a structured and repeatable method for selecting data, extracting information, analysis and synthesis. The research will entail all results and findings from a methodical way of analyzing data to identify, evaluate and interpret all published research in the area of agile lean practices to manage distributed development. Basic approach will consist of identifying how to overcome challenges faced in distributed software development through agile lean practices.

Moreover, the research will summarize all strategies that will facilitate successful implementation of agile lean methods in a distributed setting, identify gaps in current research to advise further research topics. This paper will also provide all necessary information on the feasibility of applying agile lean in distributed teams and their success, which can help position future research activities.

The purpose of this research is to essentially acquire empirical evidence that can be used to examine the extent it supports or contradicts agile lean practices impact on distributed team. It will identify information on failed or challenged implementation of agile lean principles in global software development along with its successes to provide complete information in the subject.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

The research will begin with identification of the need of the research and the questions need to be addressed through the study. Once the basis of the research is baselined review procedures and protocols will be developed and evaluated with the help of academic professionals. After the review and approval of the topic, systematic literature review process was conducted which was divided into categories, each consisting of several steps. Major steps that will be used to conduct the research are as follows:

Planning

1. Formulate research questions
2. Identify all relevant search terms and search engines
3. Determine strategy to conduct search

Execution

4. Conduct search
5. Select data sources such as published literature
6. Perform assessment of sources
7. Extract Data
8. Analyze and Synthesize Data

Reporting

9. Document report of findings and conclusions

The purpose of the planning phase is to find the need for research in the global software development subject area, develop protocols and procedures that will guide the work to be done.

The planning phase will begin with formulating the challenges that globally distributed teams and software development faces which should be addressed in the thesis. Once a high-level idea has been developed, it is used to identify the terms and operators that will be useful to search

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

publications. Literature reviews will be identified in iterations, as first iteration of a trial search which will be useful to validate terms and operators identified and help in determining synonyms or relevant terms that will be useful in the next phase. The foundational phase will define inclusion and exclusion criteria that will be used to separate the identified literatures and provide a strong foundation for the next two phase.

Execution phase will begin with conducting further research based on the set of rules, terms and operators such as distributed agile, lean-agile in distributed development, etc. Selected articles will be reviewed in a two-phased approach. Initially the title and abstract will be reviewed during the selection process. The second phase of selection will involve reading the complete article to identify the extent to which content provides the information on the distributed development and agile lean practices that is relevant to the research. A quality assessment will be performed of selected during data extraction to identify the extent to which the information clearly explains and relates to the research questions. A review of each article will be performed during data extraction to collect all information required to determine the successful steps of implementing lean agile to solve problems arising in a global software development project.

Next step will be to evaluate and critically assess extracted on the identified results and report them to document the paper work for the thesis. After analyzing various published literature sources many agile lean practices useful to the global software development will be identified along with any open issues on those methods. Interviews and further research will help evaluate framework to be used for deploying agile lean practices in multiple large distributed team setting. Interviews will be conducted with experts utilizing agile lean methodology at their distributed project or guiding agile lean implementation. The design of interview questions validates that the interviews are currently working in distributed teams or have worked in the past in a distributed setting while

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

adopting agile lean methodology to manage their project. The focus of the interview is to use their experience as a dataset to understand the challenges they faced during the implementation, how they overcame those challenges and identify if there could standard framework that can be adopted in a global or distributed software development to exploit all advantages of agile lean practices. Researchers were asked a set of questions that explicitly reveals their experience in the field of distributed development and their approaches of working in agile lean. Each interviewee was asked questions of their agile lean approaches one by one and their real-time challenges and advantages achieved in their process. The interviews helped understood the current practices being used and analyze them against the researcher's strategies. As the review suggests that researchers have identified frameworks and the desired process flow that should be adopted to provide promising results, interviews help support identified strategies.

Examples Provided by JK Essay

Results or Findings

Development efforts in global software development are mostly planned in independent modules that could be assigned to different sites and then integrated altogether in the end with a big bang so as to minimize most common coordination and communication challenges of distributed efforts (Herbsleb & Grinter, 1999). This big bang strategy even though very useful in distributed teams brings integration issues later towards the end and requires a clear understanding of all requirements in the beginning. In today's uncertain world, iterative development with incremental and continuous integration and frequent deliveries minimizes such issues significantly (Fowler, 2003).

Implementation of Agile Lean in Global Software Development

Distributed XP has been used in projects with offshore team with onshore customers. This agile practice when implemented at all distributed sites improved customer satisfaction by improving the communication quality. Experiences of experts in the field suggest that implementing XP in a distributed setting was hard and required time consuming strategic designing but was very useful and successful (Paasivaara & Lassenius, 2006). Many organizations also adopt Kanban along with distributed XP to reduce project delays, manage change requests and alleviate the challenge of changing priorities of requirement.

Many methodologies such as DAD, SAFe etc. have been gaining attention recently as they scale agile for multiple teams. Many industrial project have known to implement scrum in their distributed project and one such project reported to tailor scrum for their need in a geographically and organizationally distributed setting to increase quality of communication. The project did face some communication problems but was reported to be helpful and so far, successful (Paasivaara & Lassenius, 2006).

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Application of agile methods to global software development brings in a major communication challenge. The fact that agile methods and distributed software development lie on the opposite ends of a communication process, where former relies on informal interactions whereas the latter relies heavily upon detailed architectural design and plans to address team impediments, poses a major challenge of striking a balance in this communication spectrum in a distributed agile development (Ramesh, Cao, Mohan & Xu, 2006). Daily stand ups and informal communication practices as required per agile can be difficult to be arranged and conducted effectively on a regular basis due to many reasons such as infrastructure, culture, technical, trust and many more. Another important requirement of agile methods is short iterations and continuous integration can bring in challenges like configuration management and version management more so in a distributed setting (Paasivaara & Lassenius, 2006).

Communication as known to be the biggest challenge is also the best tool available that can be most beneficial of all if significantly emphasized upon and regarded utmost priority. Agile methods bring in transparency of work and its progress which provides room for monitoring, flexibility and constant feedback. The delivery of high quality work early and frequently along with motivating feedback also builds mutual trust and respect. Frequent integration and testing are useful in minimizing misunderstanding or misinterpretations of requirements (Paasivaara & Lassenius, 2006).

Agile practices that have known to provide success have been experimented and evolved in distributed development to best meet its need. Such practices can be divided in various groups namely continuously adjust the process, facilitate knowledge sharing, improve communication, build trust and trust but verify (Ramesh, Cao, Mohan & Xu, 2006). In the spirit of adjusting the process, agile distributed development requires some certainty. Hence, planning iterations initially

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

to develop and finalize high level architecture and critical requirements is useful to be documented in varied level of formality. Maintaining a repository with minimal documentation has also proven to be useful in distributed agile development. If distributed agile process is new to the members involved in project, then it is also equally important for them to be familiar of the processes and tools. With this learning curve, Manco observed that delivering well understood clear requirements in the initial phases instead of critical ones was very useful to the members and their efficiency. Limiting time-boxed development and synchronized work hours were also proved to be beneficial practices in distributed agile. Such practices, as effectively concluded and represented by Ramesh, Cao, Mohan and Xu, can effectively meet many of the challenges that are faced during application of agile in distributed environment. XP and Scrum were found to be beneficial for development process and management process respectively for global software development (Holmström, Fitzgerald, Ågerfalk, & Conchúir, 2006).

Dumitriu, Oprea and Mesnita recommends three measures in their research that an organization can apply to resolve agile implementation challenges in large global software development settings. Three measures are preparatory measures to reduce geographical, temporal and socio-cultural distance specific to GSD, technological measures to reduce all the distances utilizing information technology and communication tools and tailoring measure to adapt agile practices with the required tailoring for the specific situation. Organizations also implement Communities of Practice (CoP) to help support a Lean, Agile or Agile Lean transformation in a distributed setting by providing ways and tips to move forward or improve. Ericsson case during research showed that Communities of Practice is also helpful in scaling agile lean to large and distributed setting by providing knowledge sharing mechanisms and principles that are not clearly defined in agile/lean.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Applying Agile Lean Principles in Global Software Development

As described, Agile and Lean are governed by their own separate set of principles. While these principles seem very different in their scope but there is a huge overlap. There are many aspects and qualities that both share. Agile Lean Centre at Harrisburg University describes few principles that are shared aspects between lean and agile, they are: Respect for people, Small batch size, Inspection and Adaption, Sustainable pace and Continuous improvement. During the research, it was found that many practitioners have tried to map agile principles to the lean principles and define how they relate or complement each other. This connection has been identified and utilized in combination to alleviate several challenges of global software development. Agile and Lean together were found to solve many issues that are faced in functioning and management of distributed software development. Some of the issues addressed by agile lean principles are described in the table below:

Issues in GSD	Agile Lean Principles
Lack of effective communication	<i>Directly Addressed by</i> Face-To-Face conversation Amplify Learning Work Together
Cultural differences	<i>Directly Addressed by</i> Self-organizing team <i>Partially Addressed by</i> Motivated Individuals Respect People
Lack of trust	<i>Directly Addressed by</i> Work Together Self-organizing team Respect People

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Issues in GSD	Agile Lean Principles
Coordination	<p><i>Directly Addressed by</i> Work Together Self-organizing team</p> <p><i>Partially Addressed by</i> Sustainable Pace Continuous inspect and adapt Amplify Learning</p>
Time zone differences	<p><i>Directly Addressed by</i> Optimize the Whole</p> <p><i>Partially Addressed by</i> Self-organizing Team</p>
Asymmetrical processes	<p><i>Directly Addressed by</i> Eliminate Waste Optimize the whole</p> <p><i>Partially Addressed by</i> Technical Excellence Build Quality In</p>
Knowledge sharing	<p><i>Directly Addressed by</i> Work Together Amplify Learning Face-to-face Conversation</p>
Language barriers	<p><i>Directly Addressed by</i> Working Software</p> <p><i>Partially Addressed by</i> Technical Excellence Simplicity Optimize the whole</p>
Unclear roles and responsibilities	<p><i>Directly Addressed by</i> Self-organizing team</p> <p><i>Partially Addressed by</i> Respect People</p>
Lack of team spirit	<p><i>Directly Addressed by</i> Respect People Self-organizing team Face-to-face Conversation</p>

Examples Provided by JK Essay

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Issues in GSD	Agile Lean Principles
Lack of transparency	<p><i>Directly Addressed by</i> Frequent Delivery Face-to-face Conversation Continuous inspect and adapt</p> <p><i>Partially Addressed by</i> Simplicity Self-organizing team Eliminate Waste Optimize the whole</p>
Configuration management	<p><i>Directly Addressed by</i> Eliminate Waste</p> <p><i>Partially Addressed by</i> Defer Commitment Technical Excellence Frequent Delivery</p>

The study identified benefits of agile lean principles and practices in a globally distributed environment. As a combination, they provide high code quality, facilitates ownership, increase efficiency by eliminating bugs in early phases, increase collaboration and understanding, increase team cohesiveness and much more.

Challenges in Application

Research identified many practices that are being used to apply agile, lean or a combination by different organization in their global environment. Many have been reported to be successful but they still face several challenges daily due to several loopholes in the structure. And, the fact that distributed team needs to put in extra effort for active synchronous communication and collaboration adds to the effort and thus decreasing the efficiency. Study suggests that striking a balance between too much communication and too little communication in an agile global team is the biggest challenge. When combined with documentation this challenge only compounds to more complexity. Agile lean teams do not give importance to documentation but constant communication is critical. This is hard to follow in the context of distributed agile lean team as

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

time differences and limited communication between the two may end up in missing details, decreasing visibility of development progress and lack of understanding about the project.

Communication related results identified in the study for a distributed agile lean team are inability to understand architecture, inability to understand design, inability to understand customer or inability to understand requirements. Other challenges related to communication may also affect project and compound collaboration issues are lack of proficiency or skills in the English language and lack of shared knowledge or information. Discussions that may take only few minutes in a co-located team may require comprehensive collaboration and management to transfer the same intent to remote teams. Different time zones involved may also effect the efficiency of the team ad a dependency or issue that impede progress of work must wait till the other day for more information.

Another major challenge that has been identified during the research is the culture differences between the geographically distributed teams may impact the progress of the project. Difference in culture may create a lack of understanding of the responsibility and awareness. Few cultures have a strong hierarchal structure and the upper level management holds strong control over the staff where the separation between the two is significant. These cultures do not have tendency to disclose full details to all employees and the staff are mostly used to focused tasks and activities without understanding the big picture. This can create misunderstanding during the meetings where team members are to understand a project in its entirety, meet with business constantly and own responsibility of the project from beginning to end. Research suggested that there is a tendency of offshore teams to stay quite in the meeting and may hide development issues. Culture may also cause a lack of transparency which is critical in an agile lean project. With time and practices these challenges may be alleviated but team operating in their separate culture daily has

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

a high chance of sliding back into the same characteristic behavior with the addition of new members or existing members falling prey to same behavior. Overall it decreases awareness and trust within the team.

The next biggest challenge is the managing, governing or leading two or more teams around the world. With an agile lean facilitator and increased sites there is a difficulty in controlling the team by one person as a leader. Due to all the discussed reasons, there is a tendency in distributed teams and distributed agile lean project to lack agility which conflicts the core values of agile and lean.

Strategies for Adoption/Application

As discussed there are many ways to implement agile lean, various combinations are being used in the various field to provide a different value to the concerned organization. Software industry was listed in as the identified case for each of the combination discussed which was identified to be implemented in distributed teams in some cases. Research suggests that amongst these most common combinations that are being used in global team structure are utilizing lean principles to facilitate agile adoption, use lean to scale agile and the combination of agile and lean to improve software development process. Capital One and Systematic used lean lenses to evaluate and stream their development process which paved way to consider and implement successful agile practices such as Scrum or distributed XP. In the research cases lean provides organization with the governing practices such as aligning team structure with the architecture, stages of delivery within the program and milestones based on risk which help in scaling agile by alleviating challenges that arise from large complex structure of organization or distributed teams.

Using both lean and agile alongside as a combination in software development practice is the most prevalent of all in a distributed/global software development. Research has shown that Scrumban is the most common combination of agile lean that is being implemented in many organizations.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Other combinations such as Kanban system paired with agile methods like distributed XP have also proven their success and value in the organizations with distributed teams. Each combination is helpful in alleviating challenges in its unique context. For example, Scrumban principles positively impacted distributed software development in Finland and Italy case by minimizing strategic issues, project management issues, communication issues, technical issues and cultural issues amongst the team.

Craig Larman and Bas Vodde has suggested multiple practices that will be useful in scaling Agile or Lean in an Organization in his book “Practices for Scaling Lean & Agile Development”. The book provides practitioners with multiple tools and techniques that will be helpful in reducing challenges of scaling methodology in distributed environment with a focus in Scrum. While the author concentrates mainly on Scrum, the techniques will be helpful to implement any agile or lean or their combination in a globally distributed setting. Research also show that if the one of the development agile or lean are working well in an organization and target elements such as people oriented development, focus on customer value, transparency and continuous improvement activities are well integrated within their process it provides significant strength to the organization to combine and scale Lean thinking and agile software development.

Another practice that was identified during research to help successful adoption, implementation and sustainability of agile lean practices in large scale software development was establishing Communities of Practice(CoP) in the organization. A CoP in specific area was proven to be effective in every stage of transformation phases beginning from transformation to scaling to continuous improvement by facilitating knowledge sharing, organizational development, coordination and process development. An Ericsson study found in this research implemented four separate CoP in their largely distributed organization. A coaching community of practice (CoP)

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

was setup which consisted of scrum masters and coaches. The structure of the Cop was many site-specific CoP under one cross-site CoP that was working towards improving the whole organization. Coaching CoP get together weekly as a role-based CoP to share knowledge and learning. Ericsson also set up a Feature CoPs to solve the dysfunctional Scrum-of-scrums in the organization. The main goal of Feature CoP was to meet weekly or on need basis to support the coordination between the product teams working on common feature. A Developer CoP was set up at organizational level to unify tools and technology. End-to-End CoP was set up by bringing together members with various roles and titles to improve the overall development flow of a product in the organization. The main focus and aim of End-to-End CoP was organization development and increased efficiency to deliver customer value. The managers on this CoP along with the practitioners and members facilitated quick decision making.

Ericsson organized value workshop and defined common values to provide a direction and align each team's structure and practices to the best meet organization goals. Value workshops were helpful in communicating vision, building contacts, refining organization structure and creating a culture of one team in large multisite organization after the implementation of agile lean. Similarly, many other practices such as building right team, building trust between the teams, establishing good infrastructure for collaboration, investing in high quality tools that can support the work effort and defining a control of process were also found to be essential in a distributed setting.

Discussion

The study was systematically conducted using the existing literature work on the agile lean and global software development. As described in the methodology, the articles were carefully chosen to study if agile lean practices are successful in global software development and how. Due to availability of limited time the systematic literature study was followed by a few interviews to identify research gaps and understand agile lean practices being used in global software development industry. This research identifies what makes agile lean successful and how to effectively scale them in a distributed environment.

Research defines several combinations of agile and lean that currently exists in the industry and suggests that with the combination of agile and lean project members receive quick and iterative feedback while implementing the required changes and feedback. Agile Lean was also found to increase team cohesion and coordination among team members. It also provided positive environment that increase team morale and produced higher customer satisfaction and better outcomes. As described by most of the authors, practitioners also believe that agile and lean combined together complements each other very well to solve challenges raised by distances in global software development by scaling agility. In most cases, agile principles were found to facilitate flexibility in the process while lean principles increased efficiency and scale of the development process.

Agile and Lean both provides separate tools that provide specific unique value for example gemba walk of lean practices provide insight into the problems and solutions by bringing the management team to the place where the actual work is being conducted while retrospective meeting of Scrum gives the working team chance to assess the problems in the previous period solve in future without any involvement of the management. While each such practice solves an issue in software

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

development, combination of these tools and implementing them together or an enhanced hybrid form of both can provide a powerful platform to the organization that is useful in alleviating many communication, collaboration and distance challenges of global software development. Root cause analysis techniques can be used in retrospectives to reach to the best solution of previously encountered issues and also ensuring that it is not caused in the future. A plan-do-check-act cycle plan can also be helpful in retrospectives to ensure if they really adopted the discussed changes.

According to several authors and practitioners in the field every organization is different and has its unique context thus require a unique solution to overcome the current challenges. Even though standard practices, methods and tools of agile, lean or their combination are available to use, their successful implementation in an organization depends upon the context of the organization and its needs and requirements. Agile Lean was found to bring in a fundamental change for all stakeholders involved in a project in every case study of this research and therefore it requires executive sponsorship and management buy in. Too many changes can be difficult to manage and sustain among large distributed teams and thus should be carefully handled to ensure successful agile lean implementation. Synchronous communication, training and coaching was found to be critical for the successful application of agile lean in distributed team as it helps in implementing standards all across the organization. Methods used to provide standards of agile lean in organization found in this research were workshops and communities of practice.

Comparing the research and practitioners responses several similarities were found in the usage of agile lean. Even though every organization context can be unique requiring its own solution, a standard set of practices can be evolved through future studies that can guide agile lean application in a large distributed organization. One other important takeaway from the research is that an organization require a strategy to sustain agile lean in their process along with its implementation.

Conclusion

This study examines how Agile Lean is applied in distributed team working on software development in an organization by analyzing previous research in the area and interviewing some practitioners in the field. The results show what practices of Agile, Lean or their combination are being adopted in the global software development and how agile lean principles can benefit large organization that operate in such an environment as agile lean together address several issues arising due to distances in distributed development. In addition to highly iterative development practices established through agile in a distributed team, Lean thinking adds new aspects such as minimizing wastes and optimizing the whole with a greater emphasis on people-orientation, respect, trust, collaboration and transparency. Therefore, Lean in most identified cases of distributed organizations has found to be helpful in scaling agile or facilitate agile adoption. The study also reported cases where agile lean combination used to improve development process in a global development provided fast and iterative feedback. There are many other benefits that lean agile principles can provide in distributed setting and hence can be very useful to increase efficiency and cohesiveness.

Like any other methodology adoption, distances in global software development poses several challenges in implementation of agile lean and makes it difficult for agile lean to provide reported benefits or sustain. Therefore, the issue is not hybrid agile lean or any other methodology, but the right selection and integration of appropriate aspects, methods, tools and practices of any paradigm that will help solve the issues arising in the organization. Company strategy, goals and vision also play a part while choosing the aspects of a methodology to ensure alignment of the success achieved with the business. Last but not the least having a strategy to ensure success of the chosen aspects of methodology and its sustainability by not sliding back into routine is very critical.

References

- Ågerfalk, P. J., Fitzgerald, B., Holmström, H., Lings, B., Lundell, B., & Ó Conchúir, E. (2005) A framework for considering opportunities and threats in distributed software development. International Workshop on Distributed Software Development. Austrian Computer Society (August, 2005).
- Fowler, M. (2003) Using agile software process with offshore development. <http://martinfowler.com/articles/agileOffshore.html>
- Holmstrom, H, Fitzgerald, B, Agerfalk, P., & O Conchuir, E (2006). Agile practices reduce distance in global software development. *Information Systems Management*, 23 (3), 7-18.
- Paasivaara, M., & Lassenius, C. (2006). Could global software development benefit from agile methods? *Global Software Engineering, International Conference. ICGSE* (October, 2006) DOI 10.1109/ICGSE.2006.261222.
- Prikladinicki, R., Da Silva, F. Q. B., Costa, C., & César C, A. (2010). Challenges and solutions in distributed software development project management: a systematic literature review. *International Conference on Global Software Engineering* (2010). DOI 10.1109/ICGSE.2010.18
- Ramesh, B., Cao, L., Mohan, K., & Xu, P. (2006). Can distributed software development be agile? *Communications of the ACM*, 49 (10), 41-46.
- Razzak, M. A. (2016). An empirical study on lean and agile methods in global software development. *2016 IEEE 11th International Conference on Global Software Engineering Workshops (ICGSEW)*. DOI 10.1109/icgsew.2016.22
- Rodríguez, P., Markkula, J., Oivo, M., & Garbajosa, J. (2012). Analyzing the drivers of the combination of lean and agile in software development companies. *Product-Focused*

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Software Process Improvement Lecture Notes in Computer Science, 145-159. DOI 10.1007/978-3-642-31063-8_12

Wang, X. (2010). The combination of agile and lean in software development: An experience report analysis. Proceedings of the IEEE Agile Conference, 1-9.

Beck, K. et al. Agile manifesto. Retrieved, 2017, from <http://agilemanifesto.org/>

Wang, X., Conboy, K., & Cawley, O. (2010). "Leagile" Software Development: An Experience Report Analysis of the Application of Lean Approaches in Agile Software Development. The Journal of Systems and Software. DOI 10.1016/j.jss.2012.01.061

Swaminathan, B., & Jain, K. (2012). Implementing the lean concepts of continuous improvement and flow on an agile software development project - An industrial case study. 2012 Agile India. DOI 10.1109/agileindia.2012.12

Paasivaara, M., Lassenius, C., Heikkila, V. T., Dikert, K., & Engblom, C. (2013). Integrating global sites into the lean and agile transformation at Ericsson. 2013 IEEE 8th International Conference on Global Software Engineering. DOI 10.1109/icgse.2013.25

Middleton, P., & Joyce, D. (2012). Lean software management: BBC Worldwide case study. IEEE Transactions on Engineering Management, 59(1), 20-32. DOI 10.1109/tem.2010.2081675

Paasivaara, M., & Lassenius, C. (2014). Communities of practice in a large distributed agile software development organization – Case Ericsson. Information and Software Technology, 56(12), 1556-1577. DOI 10.1016/j.infsof.2014.06.008

Behm, B. (2014). Large-scale agile and lean transformation in a globally distributed organization - Case Ericsson. Master's Thesis for Degree Programme in Computer Science and Engineering at Aalto University.

APPLYING AGILE LEAN TO GLOBAL SOFTWARE DEVELOPMENT

Dumitru, F., Oprea, D., & Mesnita, G. (2011). Issues and strategy for agile global software development adoption. Proceedings of the World Multiconference on Applied Economics, Business and Development. 37-42.

Noll, J., Beecham, S., & Richardson, I. (2014). Using agile practices to solve global software development problems - A Case Study. 2014 IEEE International Conference on Global Software Engineering Workshops (ICGSEW). DOI 10.1109/ICGSEW.2014.7

Examples Provided by JK Essay