Nanda M P et .. al Journal of Management Research and Analysis (JMRA)
Available online at http://jmraonline.com
ISSN: 2394-2770, Impact Factor: 4.878, Volume 05 Issue 04, December 2018, Pages:185-188

Productivity Improvement through Agile Transformation

Nanda M P,

Research Scholar, School of Management, CMR University, Bangalore

Abstract- ***In the last few years, organizations are encountering challenges with respect to faster turnaround time with sustained on improved quality. The objective of this article is to methodical approach towards improving Turnaround time and improve business value to customer through Agile Transformation. Agile transformation focusses on organization level strategies attempting to change the traditional mindset of software development and maintenance activities and adapt to the Agile way of working. Additionally, Agile Transformation strategies tries to deliver potentially shippable products on time with enhanced business value with acceptable quality & quantity. Adoption of Agile has been gaining more and more prominence in IT organizations since they have started realizing the benefits. A case study from a software organization is considered with an emphasis on improving the productivity through Agile transformation.***

I. INTRODUCTION

In the recent years, more and more IT organizations have started embarking the journey of Agile Transformation. Due to cut throat competition, organizations have realized the need to quickly adopt the Agile way of working Agile transformation is process of transforming the nature and culture of the organization into becoming Agile. It is all about the fundamental change in the mindset of people. Iterative and Incremental development methods originated as early as 1957 with evolutionary project management. The adaptive software development emerged during the early 1970s. In 2001, seventeen software developers met to discuss the light weight development methods and together they published the Manifesto for Agile Software Development. The Guide to Agile Practices (renamed to Agile Glossary in 2016) was created by Agile Alliance in 2011. This is an open source reference of the definitions of various Agile related practices, roles, ceremonies. It also talks about the various elements, inferences, interpretations and experience guidelines from practitioners across Agile communities.

II. Basics of Agile Transformation

"Agile Transformation" is the art of refactoring the organization so that it can unlearn the traditional way of working and culturally adopt to the Agile way of building software. Fundamentally, it is all about working as a team, building backlogs and regularly producing potentially shippable products with business value to clients. It measures throughput rather than productivity. The strategies for Agile Transformation gets initiated from the point of knowing where we stand currently to where we want to head. While designing the transformation strategies, we need to consider two key aspects. First, we need a clear understanding on the company's core values. We need to be clear on our ability to accept changes. The more we design a system to for becoming more predictable, harder it becomes to change. Meanwhile, higher the design for adaptability, the system becomes less

predictable. The second aspect that needs to be considered is the business value to the customer. While designing the transformation strategy, organizations should be clear on priorities. Clear understanding is needed on whether organizations need to focus on creating business value to customer or meet commitment. During February 2001, The Manifesto for Agile Software Development was created by various delegates. This consists of 4 Agile Values and 12 Agile principles.

1) *The Agile Values*
   a. Face to Face communication, transparency, collaboration over detailed documented processes, templates, tools.
   b. Working software over detailed documentation and reports.
   c. Interactions with customer rather than effort spent on negotiating contracts
   d. Responding to requests rather than following a detailed plan
2) *The Agile Principles*
   a. The topmost priority is to gain customer satisfaction and delight through early and frequent deliveries of potentially shippable product resulting in business value to customers
   b. Exhibit flexibility in welcoming change even at a later stage of project execution
   c. Constantly delivery potentially shippable products with a focus on having shorter iterations
   d. All project stakeholders should work together with enhanced interactions and collaborations
   e. Build teams using motivated members. Have trust in the team's ability to move forward
   f. Set up a system of interacting and collaborating through face to face communication
   g. Working software is the preliminary key performance indicator to understand the progress of the project
   h. All project stakeholders must adapt to the Agile way of working and ensure a steady pace of working and delivering business value thus meeting expectations
   i. In order to bring him more agility, constant focus on upskilling on technical/ non-technical areas and good design is a must
   j. The more that we maximize the bucket of work not done, the nearer we get to closing in on the few highly valuable features which deliver highest business value to customers. Infact, this practice is truly an art and it must be mastered by all stakeholders
   k. The best in class technicalities & ideas emerge from team which are self-organized
   l. At pre-defined frequencies, teams introspect on the areas that can enable them to become more effective. It helps them to adjust, optimize and fine tune themselves to delivery more business value

III. Various Agile methods

The Agile Manifesto encompasses the idea of several already existing Agile methodologies such as:

a. Extreme Programming
b. Scrum
c. Lean
d. Kanban
e. DSDM
f. Crystal
g. Feature Driven Development
h. Adaptive software Development
i. Pragmatic Programming
j. Behavior Driven Development
k. Agile unified Process
l. Test Driven Development

In this case study, the Agile Transformation mainly focuses on Extreme programming, Scrum, Kanban & Lean.

a. Extreme Programming

Extreme Programming or XP is one of the software development methodologies which focuses on improving software quality against frequently changing business needs. Extreme Programming was conceptualized by Kent Beck while he was working on Chrysler Comprehensive Compensation System. Some of the fey elements of extreme programming includes practices such as pair programming, simplicity and clarity of code, building features only when they are needed etc. This concept has taken the name “extreme” since the beneficial practices are taken to an extreme level during implementation or execution. Kent Beck has also elaborated this in detail in Extreme Programming Explained (1999, ISBN 0-201-61641-6). Extreme Programming practices are still evolving by taking into account more lessons from the knowledge accumulated from experience. In the second edition of Extreme Programming Explained (November 2004) Kent Beck has added more Agile Values and Practices that are well differentiated between primary and corollary practices. Extreme programming emphasizes on team work. The key stakeholders such as Clients, Program Managers and Cross Functional teams work together in a highly collaborative environment. One of the main assumption while implementing Extreme Programming practice is that the cost of implementing change is mostly constant. There are many ways to achieve this. Following are some of the ways:

- Smaller Sprints
- Optimize design
- Focus on constant feedback from client
- Early bug fix, thus reducing cost of rework
- Delivering potentially shippable products to client

Some of the key steps in Extreme Programming involves the following:

- Have a very simplified design just enough to code the feature. It can be redesigned when required
- Even before coding activities, Unit Test cases needs to be derived and it needs to keep running constantly. They are automated and helps in eliminating defects early.
- Another concept of Pair Programming helps in Programming and reviewing in turns.
- Integrating and Testing happens multiple times in a day during the process of Continuous Integration
- Integrate a basic feature to a production system. This can be optimized or enhanced as and when required rather than opting for a big bang approach.

b. Scrum

The word "Scrum" was originated from an article in Harvard Business Review (1986). The authors Hirotaka Takeuchi and Ikujiro Nonaka compared the cross functional highly performing teams to Scrum formation used in Rugby game. Since the early 1990s, Scrum has been used extensively to manage complex work. The concept in Scrum is the split large chunks of work into smaller groups and, reviewing and integrating along the journey of project execution. As per 12th Annual State of Agile report, the close to 70% project teams across the software industry used Scrum as the way of working. Meanwhile, Scrum practices have caught attention of other support functions such as Sales, Business teams, HR & IT. They have similar challenges and they expect Scrum to help them easy the way of working and overcome the challenges. Scrum is one of the types of Agile implementation. Agile is a set of principles and rules that elaborates how teams collaborate and get work done in a timebound manner. It has been proven that Scrum provides lots of benefits to stakeholders such as increased productivity, increased employee satisfaction, faster time to market and improved team dynamics.

c. Lean

If an organization must become lean, it must understand customer value and processes to constantly be optimized. The ultimate objective is to have a high quality optimized processed which has near zero waste. A Lean Process Improvement approach can help an organization in the following ways:

- Eliminate waste
- Improve productivity
- Reduce cost
- Improve customer satisfaction
- Improve margins
- Decrease backlog

d. Kanban

The word Kanban was originally derived from Japan. It roughly translates to Visual card. Kanban is a Visual card that is primarily used to trigger a task. Toyota used the Kanban system in their relay systems to stabilize the flow of elements in their just-in-time production lines.

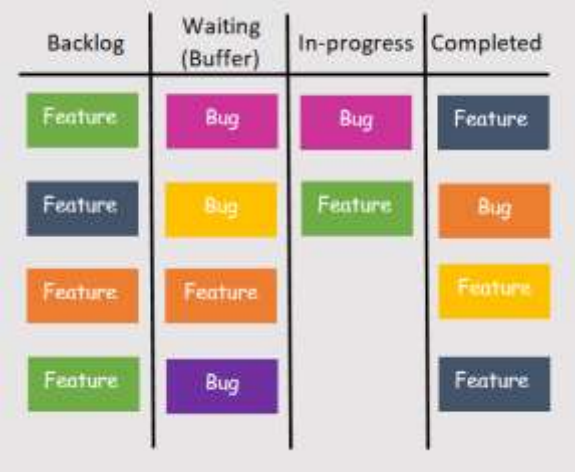

Figure 1: Example Kanban Board

There are a set of rules for effective implementation of Kanban provided by Toyota:

- Client's processes take items in a specified quantum as specified by Kanban
- Vendor generates items in the specified amounts and order as specified by Kanban
- Items cannot move without Kanban
- Each Kanban should be accompanied by an item always
- Bugs and incorrect quantum are never pushed to next stage

In case of a software maintenance lifecycle, a Kanban board just has four distinct stages:

- Backlog
- Work in Progress
- QA
- Completed

IV. OBJECTIVES OF AGILE TRANSFORMATION

The objectives of driving Agile Transformation is to provide a strong cultural shift across the organization towards embarking Agile way of working. This should ideally result in improved project performance with quantitative benefits.

- **Improved Quality** by introduction of technical practices such as CI
- **Cultural shift** through trainings and experience
- **Working as a team** by a mindset shift
- **Faster Time to Market** by improving the process cycle efficiency

Figure 2: Approach for Agile Transformation

Many organizations have embarked upon the journey of Agile Transformation and have been successfully reaping benefits from such initiatives. There are lots of studies which have mentioned about the quantitative benefits of organizations moving towards Agile. Some of the key factors which critically influence the success or failure if Agile Transformation are areas like management commitment, Employees mindset, Servant Leadership, Infrastructure, Tools and environmental setup. The key steps towards executing the transformation journey include:

- Building leadership team
- Deriving organization level vision for transformation
- Identifying transformation journey roadmap with clear milestones
- Come up with deployment plan
- Define progress assessment criteria
- Measure outcomes
- Share experience

V. BUILDING AGILE CENTER OF EXCELLENCE & AGILE GOVERNANCE

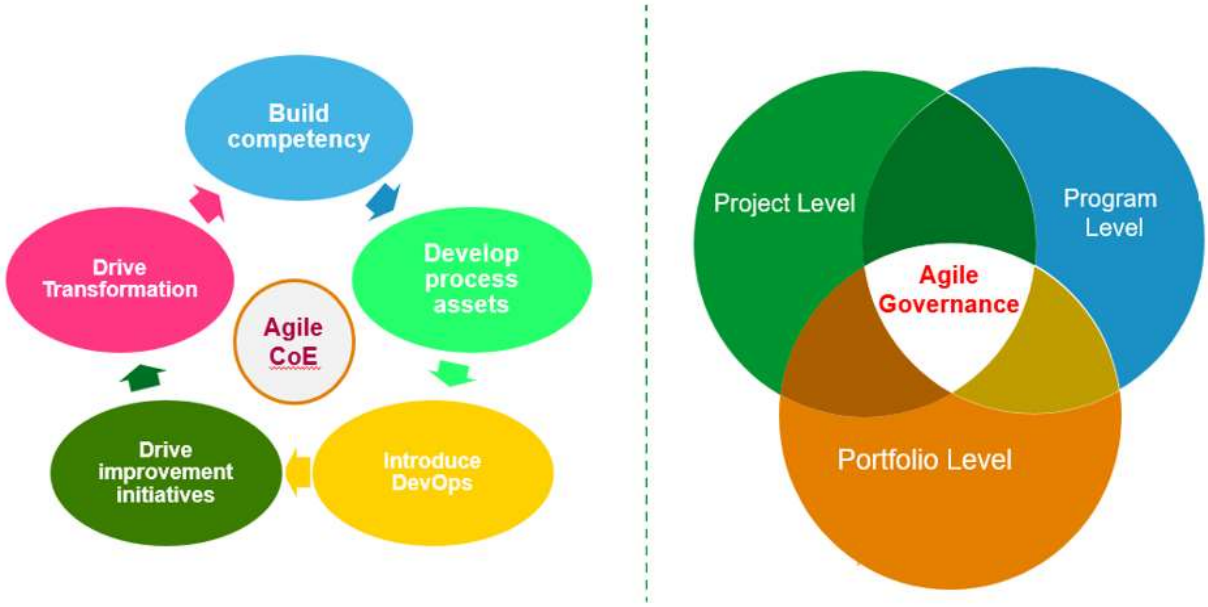

Figure 3: Building Agile Center of Excellence & Agile Governance

The Agile Center of Excellence team is a group comprising of dedicated set of motivated individuals. They possess expertise and specific skills. Their primary role is to showcase leadership role and drive the Agile Transformation journey.

Agile governance is often achieved at program level where these multiple projects interface with broader business objectives. At the program level, more comprehensive strategic goals of the organization filter down to individual projects that produce tangible results. Even at portfolio level, Agile Governance is needed to facilitate effective management to meet strategic business objectives.

V. Overall Governance

The overall governance model is needed to monitor the assess the transformation journey at defined frequency. This involves stakeholders at various levels.

Forum	Frequency	Participants	Scope
Program review meeting	Monthly	Program Managers, Quality Leads, Agile CoE	Review Agile implementation at project/ program level
Business Unit Review meeting	Monthly	BU Head, Directors, Program Managers, Quality Leads, Agile CoE	Assess the Agile transformation journey of the BU against plan
QBR	Quarterly	CEO, BU Head, Directors, Quality Head, Quality Leads	Review the progress of organization level Agile transformation journey

Table 12: Governance model

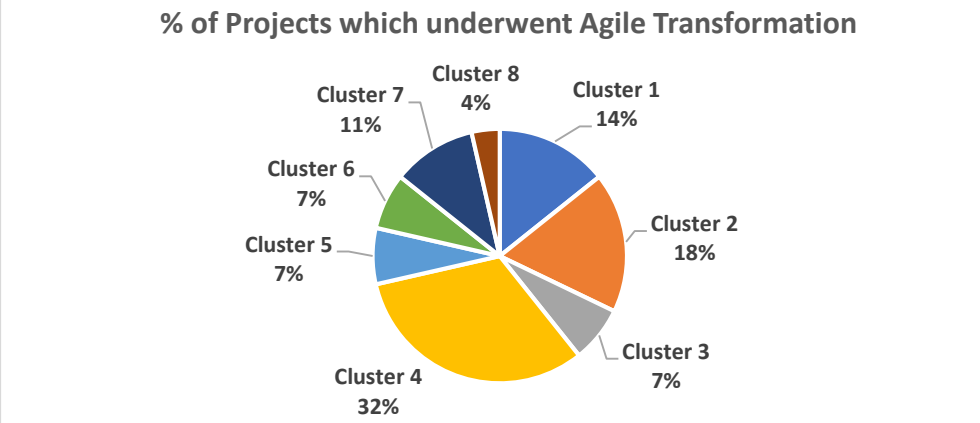

Figure 4: Cluster wise distribution of projects

Program Name	Program Type	Agile Methodology	Reason for moving to Agile way of working
Program 1	New Development	Scrum	Frequently changing requirements
Program 2	New Development	Scrum	Frequently changing requirements
Program 3	New Development	Scrum	Faster release cycle
Program 4	New Development	Scrum	Frequently changing requirements
Program 5	Support & Maintenance	Kanban	Productivity improvement
Program 6	New Development	Scrum	Frequently changing requirements
Program 7	New Development	Scrum	Frequently changing requirements
Program 8	Support & Maintenance	Kanban	Productivity improvement
Program 9	Support & Maintenance	Kanban	Productivity improvement
Program 10	Support & Maintenance	Kanban	Productivity improvement
Program 11	New Development	Scrum	Frequently changing requirements
Program 12	New Development	Scrum	Faster release cycle
Program 13	New Development	Scrum	Frequently changing requirements
Program 14	Support & Maintenance	Kanban	Productivity improvement
Program 15	New Development	Scrum	Frequently changing requirements
Program 16	Support & Maintenance	Kanban	Productivity improvement
Program 17	Support & Maintenance	Kanban	Productivity improvement
Program 18	New Development	Scrum	Faster release cycle
Program 19	Support & Maintenance	Kanban	Productivity improvement
Program 20	Support & Maintenance	Kanban	Productivity improvement
Program 21	New Development	Scrum	Faster release cycle
Program 22	New Development	Scrum	Faster release cycle
Program 23	New Development	Scrum	Faster release cycle
Program 24	New Development	Scrum	Frequently changing requirements
Program 25	Support & Maintenance	Kanban	Productivity improvement
Program 26	Support & Maintenance	Kanban	Productivity improvement
Program 27	New Development	Scrum	Faster release cycle
Program 28	New Development	Scrum	Frequently changing requirements

Table 1: List of projects which underwent Agile Transformation

The Agile Transformation journey resulted in significant quantitative improvements across programs and clusters. Following are some of the quantitative benefits observed:

- 32% improvement in productivity in terms of delivering bug fixes faster when compared to the point before starting the Agile transformation
- Team's ability to respond to frequent changes in Sprint backlog has improved by 89%
- Turnaround time of user story completion has improved by 28%. This has in turn resulted in faster time to market

Apart from these outcomes, there are number of qualitative benefits such as:

- Cultural shift observed in teams. There is a better sense of ownership across team members
- Organization has been able to build a repository of case studies based on Agile Transformation success stories
- Teams are more self-organized than ever before

By understanding and implementing the Agile way of working, teams have been able to deliver more business value to the customer. By minimizing waste, it naturally allowed it allowed more time to spent of meaningful activities which add value to the project lifecycle. This resulted in increased ability to rapidly release working software with sustained or better-quality software. Hence, we consider Agile Transformation as healthy tie-up between Agile & Lean principles and value.

REFERENCES

I. Williams, L. (2010) Agile Software Development Methodologies and Practices. Advances in Computers, 80, 1-44. https://doi.org/10.1016/S0065-2458(10)80001-4

II. Tuffs, D., Stapleton, J., West, D. and Eason, Z. (1999) Inter-Operability of DSDM with the Rational Unified Process. DSDM Consortium, 1, 1-29.

III. Highsmith, J. and Cockburn, A. (2001) Agile Software Development: The Business of Innovation. Computer, 34, 120-127. https://doi.org/10.1109/2.947100

IV. Poppendieck, M. and Poppendieck, T. (2003) Lean Software Development: An Agile Toolkit. Addison-Wesley, Boston.

V. Baruah, N. (2015) Requirement Management in Agile Software Environment. Procedia Computer Science, 62, 81-83. https://doi.org/10.1016/j.procs.2015.08.414

VI. Hamed, A.M.M. and Abushama, H. (2013) Popular Agile Approaches in Software Development: Review and Analysis. 2013 International Conference on Computing, Electrical and Electronics Engineering, Khartoum, 26-28 August 2013, 160-166. https://doi.org/10.1109/ICCEEE.2013.6633925