IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

"Agile Project Management: Adapting To Emerging Technologies In Agile Project Management For The IT Industry"

Chethan V K

(Assistant Professor) MBA Department,

CITY COLLEGE, JAYANAGAR,

BANGALORE, INDIA – 560082

Abstract

Software engineering, a relatively new discipline in the traditional engineering domain, has evolved by adopting processes from established engineering fields. However, the dynamic nature of business demands and the increasing complexity of software applications have driven software engineers to continually refine development processes to address emerging challenges. Agile project management has emerged as a groundbreaking methodology in the IT industry, enabling the rapid delivery of high-quality software products far outpacing previous methods. The core principle across various agile frameworks is the ability to adapt to change, achieved through robust quality practices, skilled practitioners, and a collaborative work environment that emphasizes customer involvement more than ever before.

The IT industry has widely embraced agile, recognizing its significant success and making it a buzzword for over a decade. This paper presents a comparative study of agile project management in both IT and non-IT industries, exploring how agile's adaptability could benefit non-IT sectors in managing deliverable quality and maintaining high delivery speed. The paper also delves into the challenges non-IT industries face in adopting agile, analyzing why certain success factors that enabled agile's success in IT may not be directly applicable to non-IT sectors.

Keywords

Agile Project Management, Emerging Technologies, IT Industry, Agile Methodology, Agile Transformation, Agile Practices, Technology Adaptation, Software Development, Agile Processes, IT Project Management, Agile Tools, Innovation in Agile, Continuous Improvement, Agile Frameworks, IT Trends, Agile Development, Digital Transformation, Technology Integration, Agile Best Practices, IT Project Success.

I. INTRODUCTION

In August 1996, a conference aimed at documenting the history of software engineering and computer science recognized software engineering as being in a "pre-engineering" phase, much like the early stages of traditional engineering disciplines. While software engineering has matured over the past two decades, it still remains a relatively recent addition to the traditional engineering fields. For over fifty years, software engineering has played a pivotal role in the engineering domain, but software development has long been a complex and often chaotic process. Initially, development relied heavily on a "code and fix" approach, which led to disorganized and unpredictable results. The absence of detailed planning, combined with the subtle and varied requirements for each software project, often resulted in unstable system designs that evolved as developers gained a clearer understanding of the product they were creating, diverging from the original vision.

While this approach was manageable for smaller applications, it quickly became problematic as systems grew in size. The process of adding new features and fixing bugs became increasingly difficult, with changes often disrupting existing functionalities. This cycle persisted until the introduction of "Methodology" as a more disciplined approach to software development. Methodologies were designed to provide structure and control, making the software development process more predictable and efficient. Software processes were soon defined to set baselines for the methods and technologies used to evaluate, support, execute, and improve software development activities.

Royce's seminal 1970 paper is widely credited with defining the "waterfall" model, a formal stage-by-stage process for managing software projects. Over time, software engineering has seen a dramatic evolution of various process models, each aimed at improving how software projects are managed. Each new model sought to address the limitations of its predecessors. In recent years, the focus has shifted toward agile processes. Agile methodologies, with their adaptable frameworks, have been widely adopted in the IT industry, resulting in significant improvements in both the quality and efficiency of software development. This shift has been especially beneficial in the context of enterprise application development, which constitutes the majority of global software development efforts, supporting the complex needs of modern businesses.

II. WHAT AGILE HAS DELIVERED TO IT PROJECTS?

Agility in project management is primarily driven by three key dimensions: aggressiveness, innovation, and flexibility. These attributes are essential when working with a product that has an unclear initial vision and operates in an unstable business environment. A frequently cited report from the Standish Group, referenced by many IT process engineering professionals, presents the performance of software projects using the traditional waterfall development approach. According to the report, 24% of projects fail, 16% succeed, and 58% are challenged. In comparison, the same survey shows that Agile development results in 14% failure, 41% success, and 48% challenged projects. This shift in outcomes highlights the growing preference for Agile methodologies, as they provide significant benefits such as low defect rates, continuous valuable development, improved visibility, and better alignment with time-to-market demands.

Further surveys conducted by Scott Ambler in 2008, 2010, and 2011 confirm the increasing adaptability of the IT industry to Agile, as success rates have consistently improved. This raises an important question: *Can Agile deliver the same benefits to non-IT projects?* To answer this, the discussion will be structured around the following topics:

- Understanding the success factors of Agile in IT projects
- Analyzing the suitability of these success factors for non-IT projects
- Identifying gaps and providing justifications for these differences

III. SUCCESS-FACTORS FOR IT PROJECTS IN AGILE IMPLEMENTATION

The success of Agile in IT projects is attributed to several favorable conditions specific to the IT industry. These factors have made Agile a widely recognized and successful project management methodology. To determine the relevance of Agile for non-IT projects, it is important to explore three core sets of success factors that have contributed to its success in IT environments. These factors are interconnected and form the foundation of Agile's effectiveness, complementing each other in driving success. The three sets of success factors are:

1. **Principles behind Agile Methodology**

   The core principles of Agile emphasize flexibility, collaboration, and iterative progress, which allow teams to quickly adapt to changing requirements and deliver high-quality products incrementally.

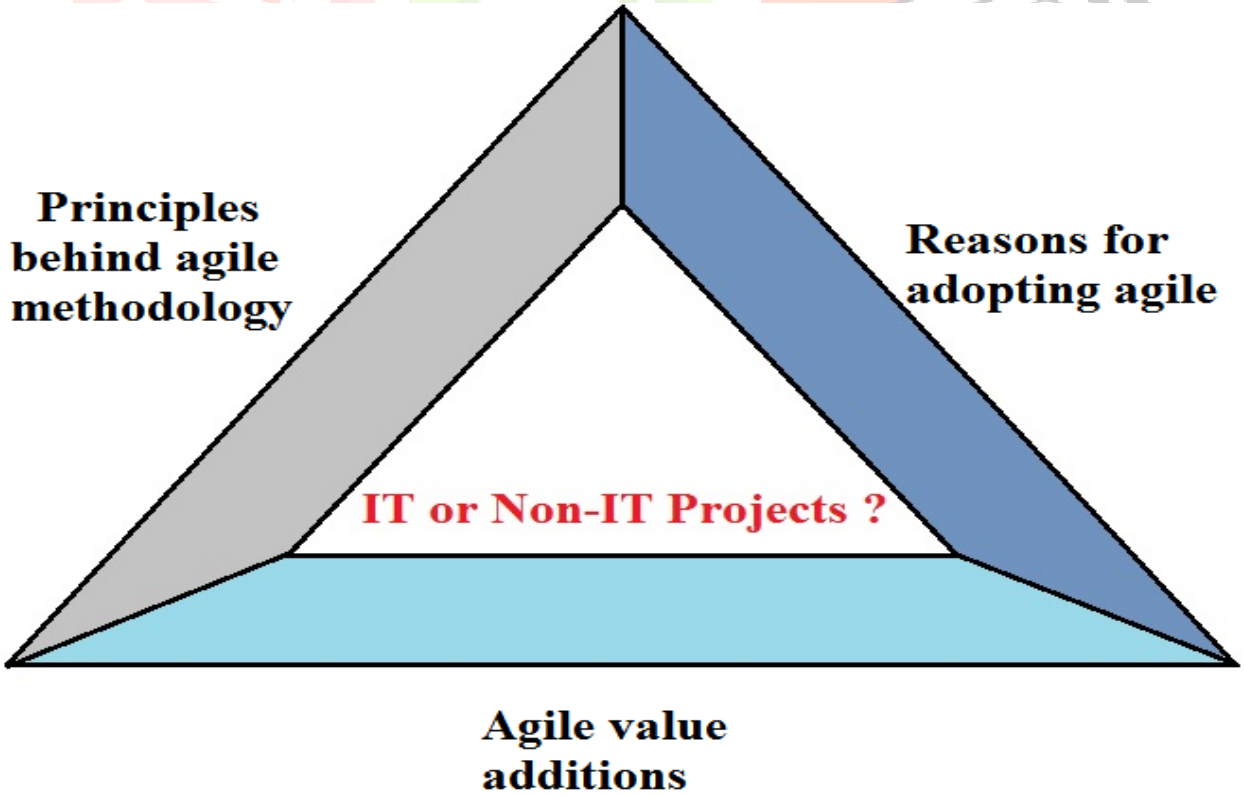
2. **Agile Value Additions**

   Agile adds significant value by improving product quality, reducing waste, boosting team morale, and ensuring that customer needs are met. This approach focuses on continuous improvement and efficient delivery of value.

3. **Reasons for Adopting Agile**

   Agile's adoption is primarily driven by the need for faster delivery, greater customer alignment, and the ability to adapt to changing business and market conditions. These factors make Agile particularly appealing in environments where responsiveness and adaptability are crucial.

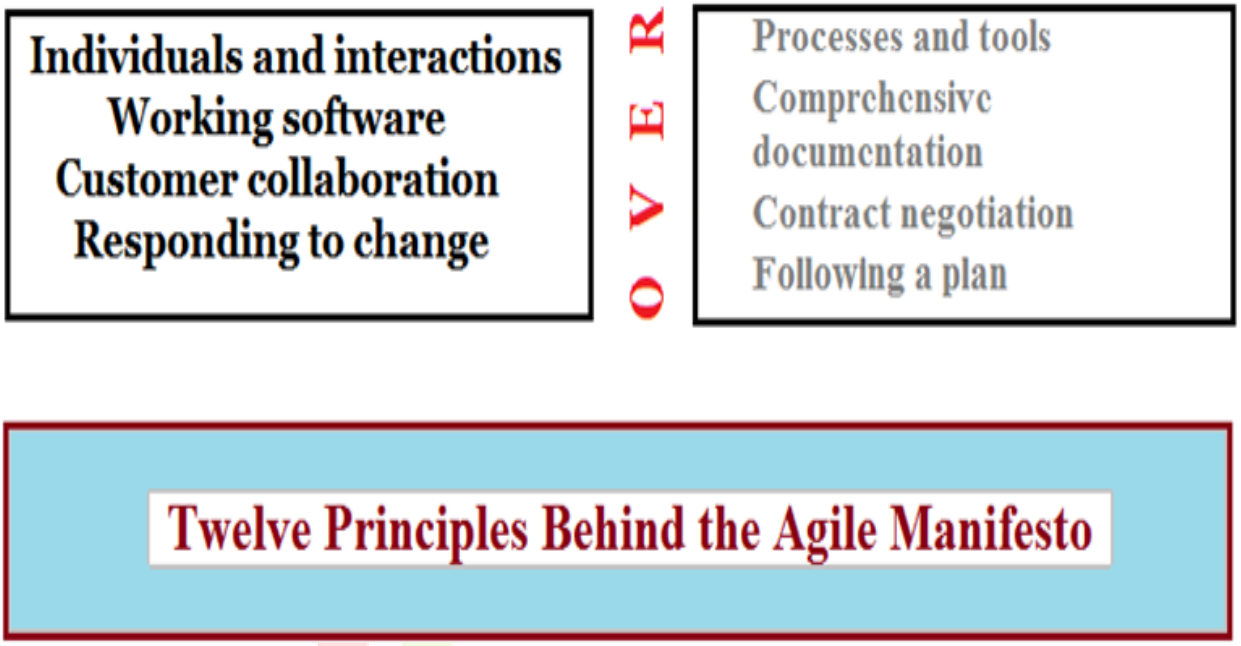
These success factors are essential for the successful application of Agile in IT projects. In the following sections, we will explore how these factors apply to non-IT industries and assess the challenges and opportunities for adopting Agile outside the IT sector.



A. Principles behind Agile Methodology

As traditional software development methodologies struggled to simplify the development process and achieve high success rates in IT projects, a group of seventeen experienced software practitioners came together in February 2001 to create the Agile Manifesto. The intent behind the Agile Manifesto was not to oppose methodologies, but rather to restore credibility to the concept of a structured development approach. Although the group had differing views on various issues, they found common ground in their shared commitment to improving the software development process.

The four values, derived from the twelve principles of the Agile Manifesto, are shown in the top-left box of Figure 2. These values represent the core tenets of Agile adoption, while the phrases on the right side reflect the underlying success factors of other existing methodologies. These four values quickly became fundamental success factors of Agile project management, as they distinguished themselves by simplifying the process and establishing a solid foundation for Agile practices. As a result, they became essential tools in the Agile toolkit.

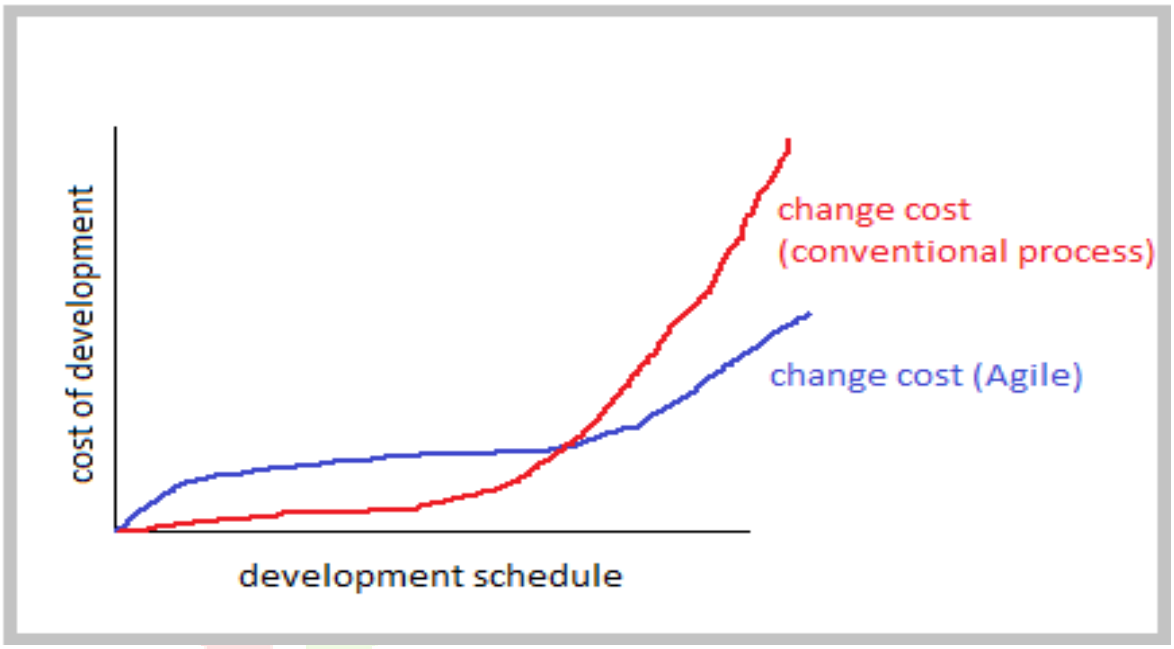


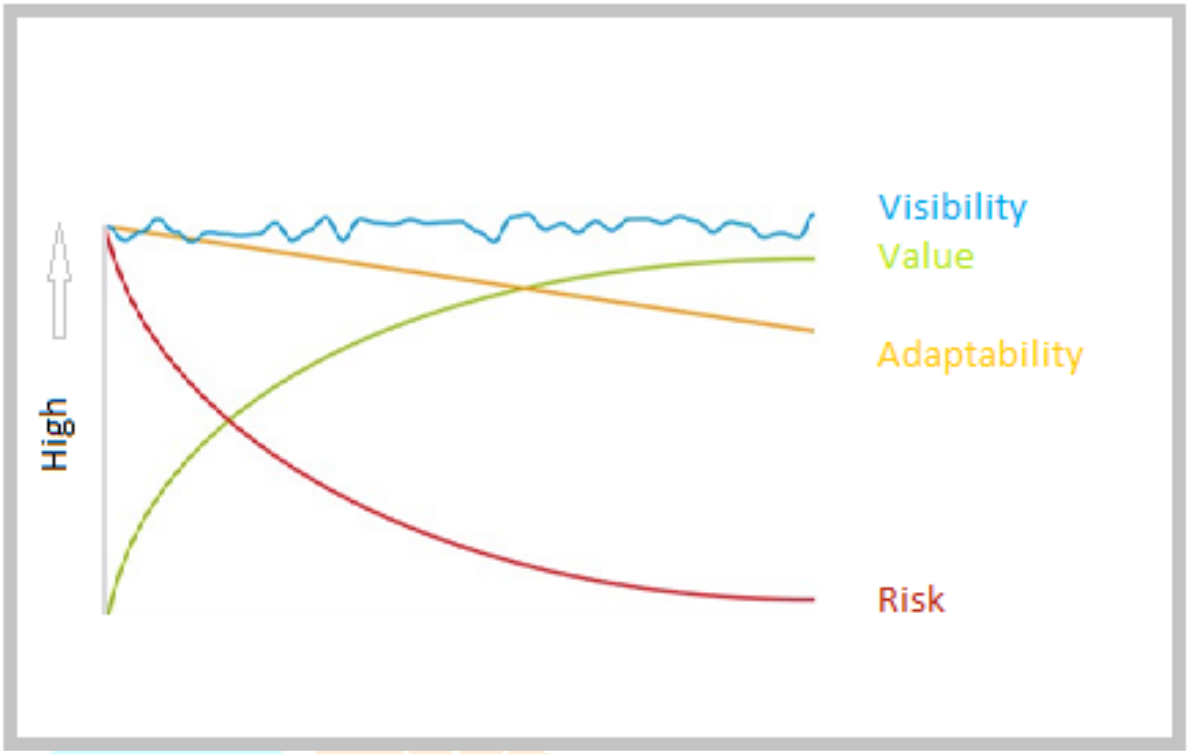
B. Agile Value Additions

As illustrated in the graph in Figure 3, the cost of change in Agile development processes tends to rise more gradually after the midpoint of development, whereas in traditional development processes, the cost of change increases sharply after the same point. This comparison of change costs highlights a key difference between the two approaches. Additionally, it was found that in traditional development, more time and cost are spent on identifying defects, fixing them, testing, and deploying to production. In contrast, the Agile process benefits from continuous customer involvement, which allows for early identification and resolution of defects. As a result, defect fixing and deployment costs in Agile remain more predictable and lower as the project progresses.

From these comparisons, several key inferences can be drawn about the value added by Agile processes:

1. **Consistent Visibility of Project Health**: Agile provides ongoing transparency about the status and health of the project, helping teams stay on track.
2. **Improved Software Usability**: Agile development places high value on software usability, which increases steadily, even in the early stages of development.
3. **Reduced Risk**: The risk associated with the project significantly decreases as development progresses.
4. **Stability in Change Management**: Agile ensures more stability and less drastic increases in costs related to change management.
5. **Predictable, Low Defect Fixing and Deployment Costs**: With Agile, the costs for fixing defects and deploying are more predictable and remain low as the project advances.
6. **Sustained Visibility, High Initial Functional Value, and Reduced Risk**: Agile provides consistent project visibility, delivers high functional value early on, and reduces risks throughout the project lifecycle.





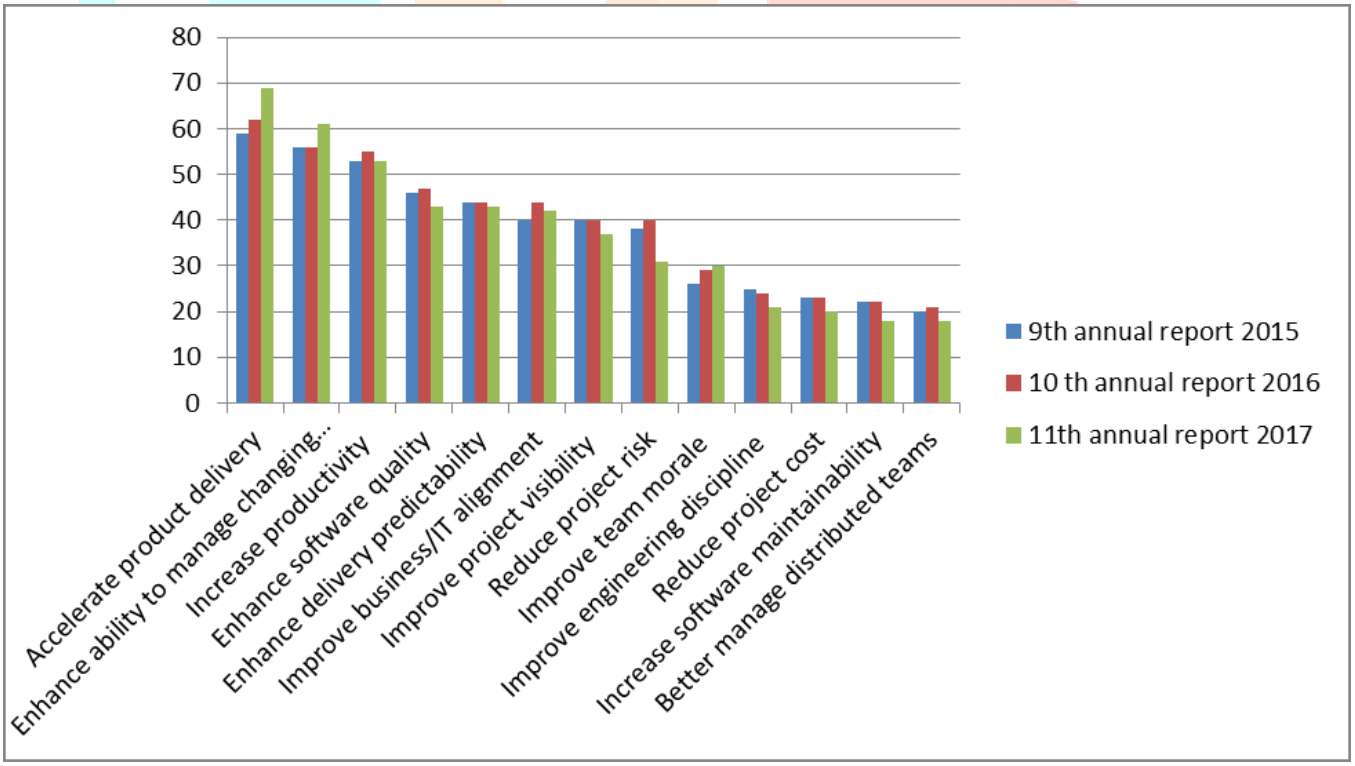
C. Reasons for Adopting the Agile Approach

While the overall trends in adaptability show consistent positive growth over the past decade, as highlighted by numerous studies, it is useful to examine the specific reasons for Agile's adoption in recent years. A survey conducted by VersionOne over three consecutive years—2014, 2015, and 2016—provides valuable insights into the evolving reasons behind Agile's increasing preference. Although the data from these three years shows only slight variations, it offers enough information to identify key factors driving the growing inclination towards Agile in today's mature process adaptability landscape.

The incremental but consistent changes in adoption reasons over these years further reflect the shifting preferences of practitioners.

Table 1: VersionOne report (Aggregated) for last three years on reasons for adopting Agile methodology.

Reasons for Adopting Agile	9th annual report 2015	10th annual report 2016	11th annual report 2017
Accelerate product delivery	59	62	69
Enhance ability to manage changing priorities	56	56	61
Increase productivity	53	55	53
Enhance software quality	46	47	43
Enhance delivery predictability	44	44	43
Improve business/IT alignment	40	44	42
Improve project visibility	40	40	37
Reduce project risk	38	40	31
Improve team morale	26	29	30
Improve engineering discipline	25	24	21
Reduce project cost	23	23	20
Increase software maintainability	22	22	18
Better manage distributed teams	20	21	18

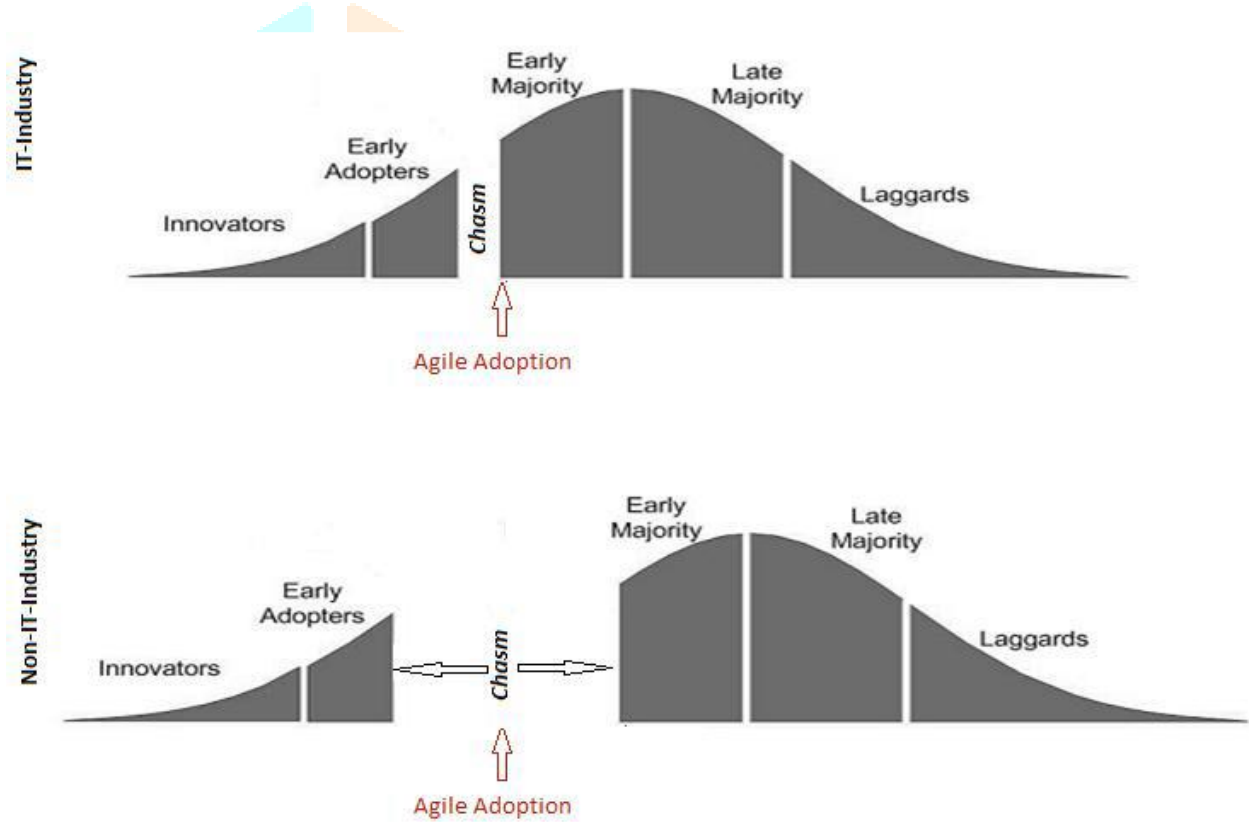


The graph in displays the reasons for Agile adoption, with the most widely adopted reasons on the left and the least adopted on the right. Analyzing the first two segments of the graph for the years 2015, 2016, and 2017 reveals a positive growth trend for "Accelerate product delivery" and "Enhance ability to manage changing priorities." These two factors show increasing popularity in 2017 compared to the previous two years, suggesting they remain key drivers for Agile adoption and continue to lead in terms of demand.

IV. Investigation of Agile Approach Adaptability in Non-IT Industries

A comprehensive literature review, along with a survey of nineteen medium and large non-IT companies, conducted by Edivandro C. Conforto et al. in their work titled *Crucibles of Leadership*, highlights the challenges faced by non-IT companies with their current project management practices. These practices, however, significantly differ from the Agile approach. The study identifies key factors such as "success terms and conditions" and "working environments" that influence the adoption of Agile in non-IT sectors. The findings suggest that Agile has not yet gained widespread recognition or adoption in non-IT industries.

This lag in Agile adoption in non-IT sectors can be explained through Roger's Technology Adoption Lifecycle, a well-known innovation diffusion model that divides adopters into five categories, represented as a normal distribution. In his discussions, Moore argues that there exists a significant gap, or "chasm," between early adopters and the early majority. Currently, the IT industry is on the verge of crossing this "chasm," with Agile methodology rapidly gaining traction. However, non-IT industries are still struggling with this transition.



Technology adoption lifecycle and comparison of Agile process adoption in IT-Industry and Non-IT-Industry

Analysis of these three sets of success-factors as presented in the sections above, we figure out that second and third set as holds a deserved merits for any industry in general. However the first set of success-factors, that is, Principles behind Agile methodology are distinct and good with projects for which there are no pre-established standards or fast-changing standards and hence innovative in nature. Most of the IT project typically falls in this category. Contrary to this is a big subset of non-IT projects, specifically in core engineering disciplines that have well-defined standards, rigorously tested and stable procedures to follow. They also sometimes have a requirement to obtain a compliance certificate from the governing authorities and stakeholders. A relevant case study is reported in by Owen R et. which discuss the applicability of Agile project management to the construction industry . The paper concludes that there is a good possibility of better results in terms of productivity and risk mitigation for pre-design and design phases respectively for a

construction project with Agile project management approach. But the construction phase is of a different nature.

Construction phase is influenced by lots of disperse factors. These factors include diversity of workforce, their professional abilities, and remunerations, education level etc. which leads to serious dis-alignment to Agile principles because of construction culture baselines, which are already well established to the roots of the process. Moreover, values of Agile process as described in the agile manifesto along with its underlying principles encourage practices which may not be completely suitable in non-IT industry, specially core engineering projects. A number of such Agile practices are listed which may not fit naturally to non-IT projects.

1. Close team collaboration may not be possible as the team includes varying skill forces of different capability and lot of hierarchical levels, leaving a big gap in process understanding and communication.
2. Roles could not be exchanged as easily as assumed by Agile methodology for non-IT projects.
3. Procurements and resource finding process may not best fit in Agile framework.
4. Whole team may not be capable to adapt to self-evolving ability. This requires high skill and motivations.
5. The team size in many non-IT projects need to be huge, which itself is contrary to Agile way of doing as it limits extensive and open communication and restricts the gelling of the team members.
6. Federated project distribution is difficult to follow up with Agile model
7. Compliance process and standard's certification dependency from external bodies is not iterative in nature.

So, the cause analysis points out the partial suitability of Agile methodology and hence needs the tailored approach that is custom made for respective disciplines and industry defects. For many disciplines, the Agile guidelines still needs to evolve or the guidelines are vague.

V. CONCLUSIONS

In this review work to answer the research question „Can Agile deliver the same impact for non-IT projects?", on one hand, we have tried to establish the importance of Agile success-factors in IT-industry and their importance for any industry in general. On the other hand we compared the needs of non-IT industry with the instance of construction industry align the discussion in the context of a typical core engineering project. We found that although the Agile success-factors hold good value to such cases, however the principles of Agile methodology sometimes is an exception in the sense that it may not naturally fit some of the phases of development in non-IT projects. These principles derive the process tools and sub-practice which further define the industry specific tools and models and their way of doing things. Hence these Agile principles and their derivatives as the industry specific tools and models partially fit the nature of non-IT projects, so a hybrid modeling may be more suited to cater such needs instead of borrowing the whole of the Agile idea. To summarize the whole discussion in one sentence, we suggest the formulation of a tailored set of success-factors suited for each discipline individually, and then defining sub-process and tool on top of it to push the adoptability of Agile process across the chasm to fully harvest the agile process benefits.

REFERENCES

A. Brennecke, R. Keil-slawik, W. Aspray, R. Keil-slawik, and D. L. Parnas, “History of Software Engineering,” 1996. M. A. Awad, “A Comparison between Agile and Traditional Software Development Methodologies,” *Univ. West. Aust.*, vol. 1, pp. 1–300, 2005. D. W. W. Royce, “Managing the Development of large Software Systems,” *Ieee Wescon*, no. August, pp. 1–9, 1970. “Genesis Consulting.” [Online]. Available: http://www.genesisconsulting.com/insights/leveraging-lean-and-agile-in-the-public-sector. [Accessed: 17-Jun-2017]. K. Beck, “Agile Manifesto,” *Agil. Manif.*, p. 2001, 2001. A. B. M. Moniruzzaman and S. A. Hossain, “Comparative Study on Agile Software Development Methodologies,” *Glob. J. Comput. Sci. Technol. Softw. Data Eng.*, vol. 13, no. 7, 2013. Version One, “State Of Agile Survey,” *Ann. Phys. (N. Y).*, vol. 54, p. 10, 2015. P. Allisy-Roberts, P. Ambrosi, D. T. Bartlett, B. M. Coursey, L. A. DeWerd, E. Fantuzzi, and J. C. McDonald, “Executive Summary,” *J. ICRU*, vol. 6, no. 2, pp. 7–8, 2006. © 2017 VersionOne. Inc, “Annual State of Agile Report,” 2017. W. G. Bennis and R. J. Thomas, “Crucibles of leadership,” *Harv. Bus. Rev.*, vol. 80, no. 9, p. 39, 2002. [11] E. M. Rogers, *Diffusion of Innovations*. 1995. [12] R. The, *Crossing the chasm*, vol. 20, no. 10. 2007. [13] R. Owen, L. Koskela, G. Henrich, and R. Codinhoto, “Is agile project management applicable to construction?,” *Proc., 14th Annu. Conf. Int. Gr. Lean Constr.*, no. June 2017, pp. 51–66, 2006.