Telecom Business Review
13 (1) 2020, 27-35
http://publishingindia.com/tbr/

Value Stream Mapping (VSM) Led Approach for Waste and Time to Market Reduction in Software Product Development Process

Shekhar Tankhiwale*, Sachin Saraf**

**Head Communications, Media, Entertainment (CME) Consulting, Tech Mahindra Ltd, Pune, Maharashtra, India. Email: dr_sat08@yahoo.co.uk*

***Senior Vice President & Vertical Business Unit Head, Communications, Media, Entertainment (CME), Tech Mahindra Ltd, Pune, Maharashtra, India. Email: sachins@techmahindra.com*

Abstract

Digital transformation involving business model, products and business process innovation is the hottest buzzword in the industry today. Digital transformation in Telecom Service Providers (Telco) needing quicker software product and services development and delivery is heavily leveraging the Agile DevOps techniques. So far, the attempts at expediting product rollout have been only partly successful in realizing their stated objectives of faster time to market, and maximizing return on investments (ROI) for product and services innovation investments. Among the challenges, lack of end to end value stream visualization, siloed tools with different data sources to lacking right measures, governance resulting are the key ones resulting in longer & unpredictable time to value. The challenges can be remediated by leveraging Value Stream Management (VSM) approach being widely used in the manufacturing industry. In this paper we present value stream management (VSM) led approach to remediate agile DevOps implementations as well as to better digital transformation ROI. This approach hinges on the key improvement strategies vis-à-vis traditional in respect of a) creating end to end value chain visibility facilitating value chain monitoring and tracking b) developing a normalized common data model across disparate value chain components c) integrating product engineering tools across the value chain d) eliminating non value adds (NVAs) from the value chain. The approach has been illustrated by taking an example of VSM led DevOps implementation in a telco covering the methodology in detail, leading to the improvements in key KPIs.

Keywords: Digital Telco, OTT, Data Monetization, Evolution of Telecom in India, Data Monetization Strategies, Digital Business, Mobile Network Operator

1. Introduction

1.1. Need for VSM Approach for Software Product Development

Telcos aggressively engaged in digital transformation and new products/services rollout are faced with the challenge of selecting the right approach and the methodology for a cost effective and efficient way of product development. The various approaches that have been tried range from traditional waterfall to later v-model and more recently Agile approach (Balaji & Murugaiyan, 2020). The biggest advantage the agile approach offers over the previous ones is that of being able to accommodate the changing requirements (Martin, 2003). Agile DevOps is increasingly being adopted as a methodology of choice by organisations to deliver software products faster and with higher quality (Wettinger et al., 2015). However the issues pertaining to the lack of end to end visibility of agile DevOps value stream, siloed implementation etc seem to be impacting agility, quality and risk mitigation (Condo & Lo Giudice, 2020) in software product development. In particular, there are three major issues the teams are grappling with which are dragging down the agile product development process:

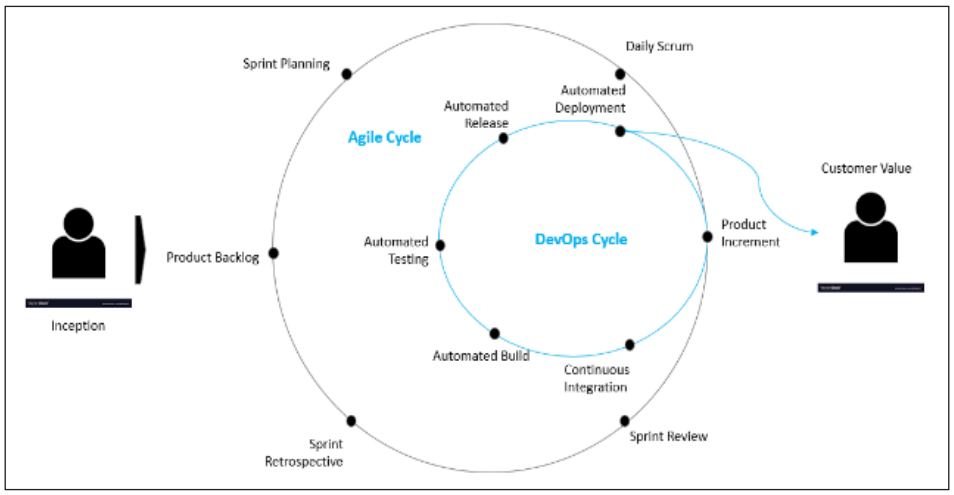
1.1.1. Non Value Adds (Nvas) in the Process Not Being Addressed Fully before Value Chain Automation

It is imperative for organisations to know beforehand what is adding value as also what is clogging up the flow and throughput in their agile DevOps enabled product development process. To maximise the benefits

of automation the non-value adds must be identified and eliminated before taking up an automation program in code build, test and deployment areas (Ung, 2020).

1.1.2. Lack of Clarity Around the Scope i.e. the Start and the End Point of Process

In large number of cases where Agile DevOps has been implemented, the application development and delivery teams seem to have been focusing on the improvements in either of Agile (involving PI planning, backlog, daily scrum etc) or DevOps cycle involving continuous build, integration, testing etc (Fig. 1 below), not both i.e. the end to end from concept to product, thereby limiting the benefits delivered by the improvements to either of the cycles (West & Sklavounakis, 2019). The real end to end agile DevOps would include all the phases starting with inception to product backlog creation to product increment development (minimum viable product-MVP) and then to inner concentric circle, being referred to as DevOps cycle all the way to value delivery for the customer (Chapman, 2020). However not many organisations seem to be doing it right by focussing either on the Agile or the DevOps cycle and thus accruing limited benefits.



Source: West & Sklavounakis, 2020

Fig. 1: End to End Agile Cycle

1.1.3. The Lack of End to End Visibility into Enterprise Software Delivery at Scale, Limiting Team's Ability to Manage, Measure, Across and Improve Software Product Development Process as a Whole

These challenges/issues pertaining to tracking, monitoring and managing the process and eliminating wastes obviously lead to sub-optimal performance of agile DevOps product development process resulting in lowered throughput and velocity (Sambandam, 2020).

Hence the objective of this paper is to elaborate a solution approach for addressing the issues mentioned above leveraging strategies such as a) creating end to end value stream visibility facilitating value chain monitoring and tracking b) developing a normalised common data model across disparate value stream components c) integrating product engineering tools across the value stream d) eliminating non value adds (NVAs) from the value stream.

The remaining portion of the paper has been structured in the following way:

Section 2: Strategies to Remediate Agile DevOps Enabled Product Development Process

Section 3: How to Go About Software Product Development Process Optimisation

Section 4: A Proof of Concept (PoC) for US Telecom Services Provider

Section 5: Discussion and Conclusion

Section 6: Implications of the study

Section 7: Acknowledgements

Section 8: References

2. Strategies to Remediate Agile DevOps Enabled Product Development Process

In order to address the challenges that product development and delivery teams implementing agile DevOps are facing, organisations need to embrace a new value stream mapping (VSM) led approach for their agile implementations.

As per Condo and Lo (2020) VSM is a methodology combining process, people and technology that aims at mapping, optimizing, visualizing and measuring "as-is" and improving "to-be" and governing business value flow (including epics, stories, and work items) through heterogeneous enterprise software delivery pipelines. Various VSM software tools enable the practices of VSM.

In order to address the challenges prevailing with current agile implementations, the improvement teams need to focus on four pronged strategies as below:

- Accelerate delivery by identifying and eliminating non value added activities (eight wastes of lean) in the 'as-is' application development process. The waste in Agile DevOps pipeline could be of eight different types e.g. waiting. If the process requires waiting for an approval by senior authority and is found to be incurring loss of efficiency, then the process could be re-engineered with senior stakeholder's approval activity taken up at the start of the process. This removes non value add (waiting time) and accelerates the process.
- Effectively integrate disparate agile DevOps tools which might be implemented across 'as-is' process--into a common horizontal VSM tool layer and create an e2e process visibility. Existing DevOps implementations have many disparate automation/tools implementation for 'as-is' process. E.g. Atlassian tools for code build, Selenium for Testing, AppDynamics for monitoring performance etc. However, since these tools are not interconnected /integrated and that adds to inefficiencies in the process.
- Facilitate data capture across end to end application development and delivery process by creating a normalised common data model covering the entire process. Once the disparate tools are integrated/interconnected, the data pertaining to the underlying DevOps process can be brought up in a horizontal data store and leveraged for variety of purposes from dash boarding to efficiency improvement.
- Improve governance and control through right measures and metrics at the back of end to end visibility and deep analytical insights powered by data. Once the disparate tools are integrated the data across entire value stream could be harnessed by setting up right measures (KPIs) on the process. These measures can be tracked and monitored to initiate continuous improvement across the value stream.

3. How to Go About Software Product Development Process Optimisation?

In order to optimise product development process, a systematic four-pronged approach as elucidated in Fig. 2 is needed. This approach was also implemented at the largest US based telecom service provider for their trouble ticket management platform development and deployment (TechM PoC, 2019).

3.1. Step 1

Map "As-Is" Software Product Development Value Stream- "as-is" value stream assessment should primarily be aimed at gaining clarity on as-is state of affairs. This includes gaining a good understanding about- (TechM Research, 2019).

- The key activities in the current process of product development and operations.
- Who are the key stakeholders/actors involved in the as-is process?
- How is the agile DevOps process performance measured today? What are the key Performance Indicators (KPIs) used to assess as-is performance?
- What is the average activity time for individual process steps such as plan, build, environment set up, testing etc.? What is the end to end lead time (which is aggregation of the activity time)? What is the release cycle time for a typical software release/MVP? What is the as-is process throughput
- What is the Value Added (VA) and non-value added (NVA) time for each step? This excludes waiting time—such as being in queue—or any other non-productive work time.

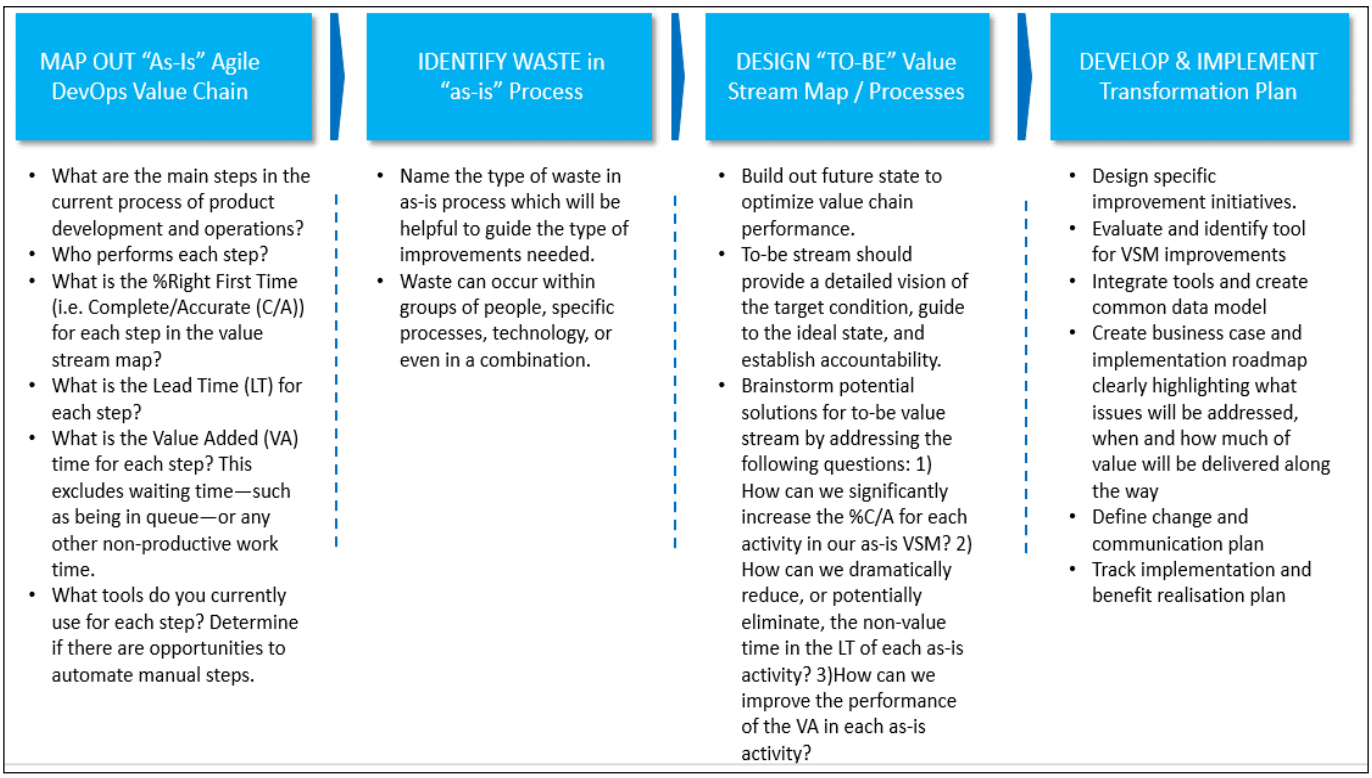


Fig. 2: Approach to Software Product Development Process Optimisation

- What process automation tools are being used for each step? E.g. Selenium for test automation, Jenkins for continuous automation. Determine if there are further opportunities to automate manual steps?

3.2. Step 2

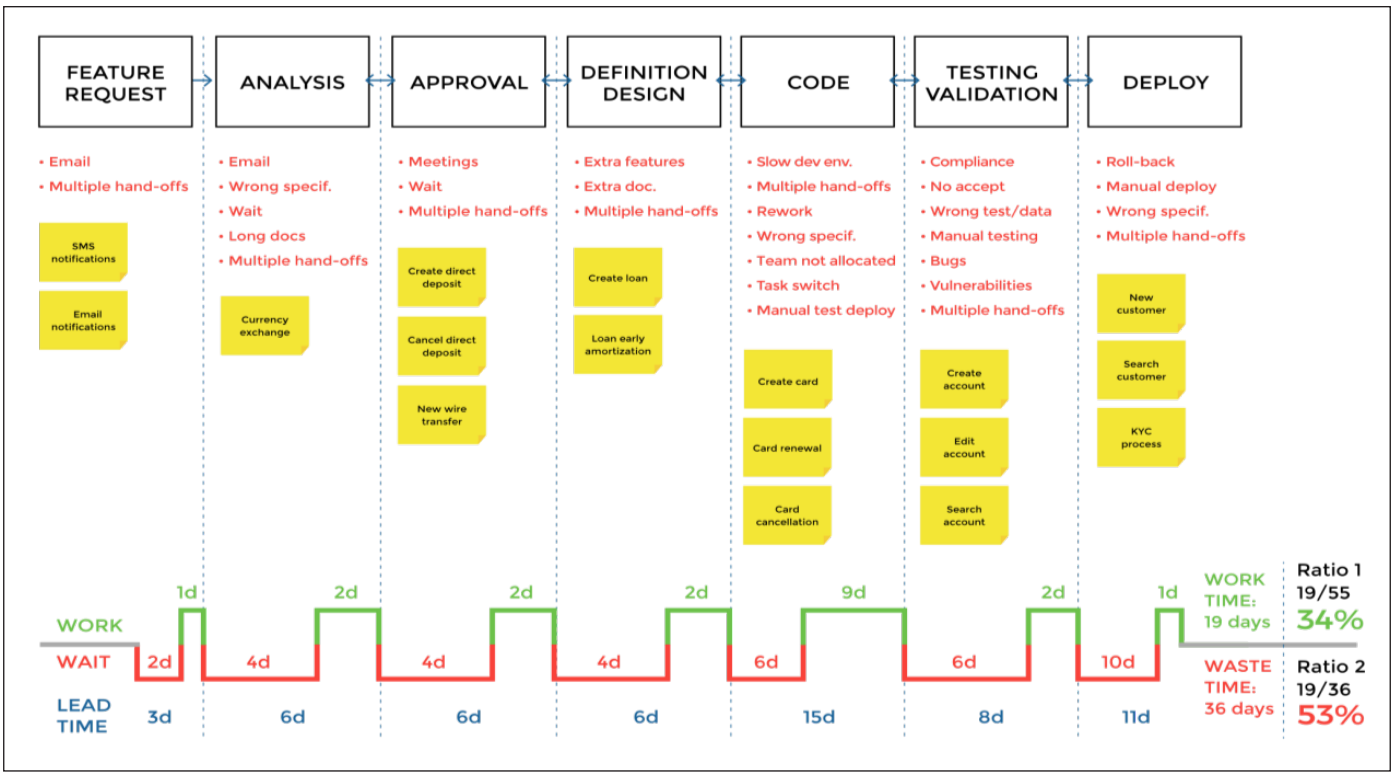
This step involves identifying waste in "as-is"- (TechM Research, 2019) As-is assessment should clearly bring out the details around the activities where the time is being wasted, whether the releases are happening on time and meeting the quality gates. The example in Fig. 3 illustrates how non value-added activities in the process such as multiple hand-offs, wait time, slow development environment etc. drag the efficiency and productivity down.

As part of Toyota Production System (TPS), Taiichi Ohno, the Chief Engineer at Toyota developed seven wastes (Muda). The seven wastes are Transportation, Inventory, Motion, Waiting (e.g. a feature is complete but waiting for stakeholder approval), Overproduction (e.g. extra features used rarely), Over processing (e.g. creating excess capacities) and Defects (e.g. code defects /technical debt surfacing late in the lifecycle). They are often referred to by the acronym 'TIMWOOD'. The 8th waste of non-utilized talent or 'Skills' of workers (e.g. lack of full stack developer requiring frequent hand-over leads to wastage) was later introduced in the 1990s when the Toyota Production System was adopted in the Western world (Skhmot, 2017). As a result, the eight wastes are commonly referred to as 'TO WISDOM'.

3.3 Step 3

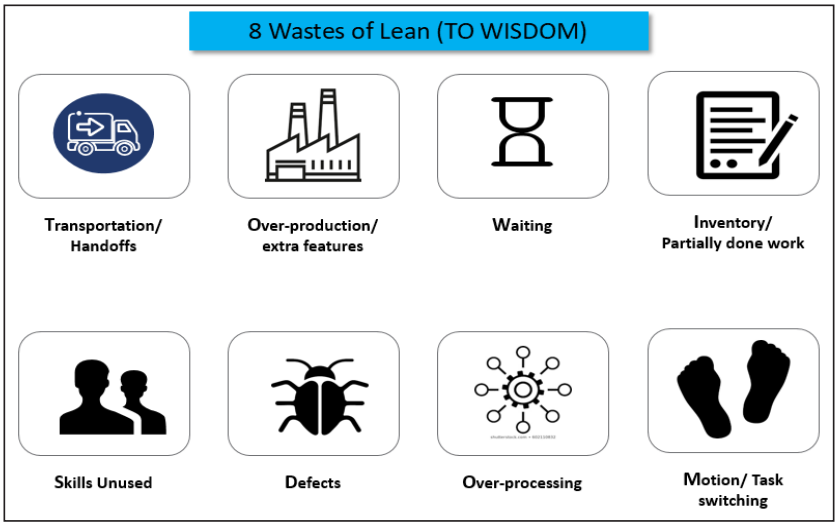
Design "To-Be" Value Stream- After having mapped "as-is" and identified wastage the next logical step is to analyse in detail the performance of the value stream and design 'to-be' value stream addressing the issues identified in as-is. The aim of the reformation should be to maximize flow and throughput of the process i.e. the process should be able to compress end to end release cycle time and deliver more number of releases in a stipulated time period (TechM Research, 2019).

The primary objective of 'To-be' value stream design should involve activities to optimize overall value stream performance. To-be value stream should bring out a detailed view of the to-be map, address issues/ wastages identified in the as-is assessment, and establish clear roles and responsibilities and accountability for the stakeholders across the value stream.



Source: TechM Research, 2019

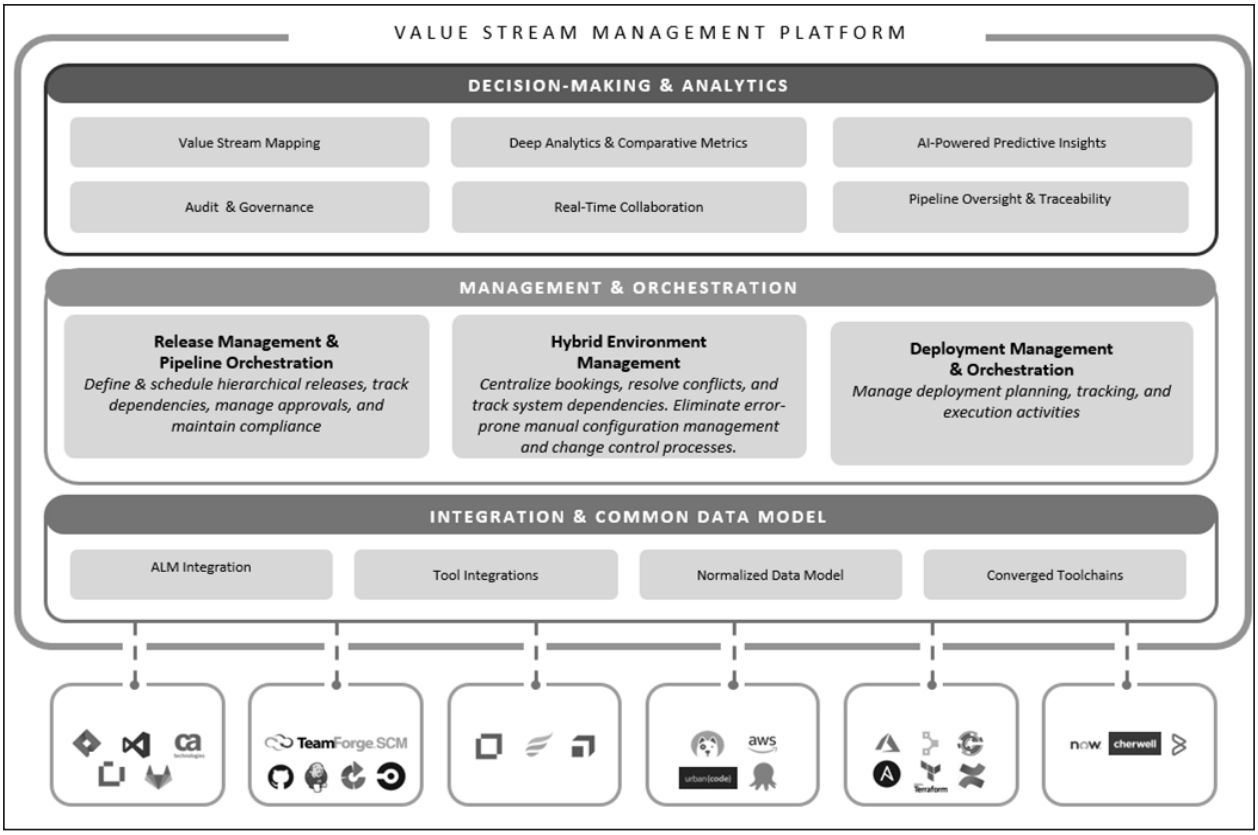
Fig. 3: NVA in the Process



Source: Skhmot, 2017

Fig. 4: Eight Wastes of Lean

The potential solutions for 'to-be' value stream design hinges on addressing the following questions:



Source: The Plutora Platform - Value Stream Management Platform - Plutora.com, 2020

Fig. 5: A Typical Value Stream Management Platform

1) How can we significantly increase the % quantum of complete/accurate work items for each activity in our as-is VSM? 2) How can we potentially eliminate, the non-value time in the lead time of each as-is activity and improve process cycle efficiency (PCE)? 3) How can we improve the performance of the value added in each as-is activity? 4) How can be leverage KANBAN to continuously ensure shift right on the Kanban dashboard?

This also involves integrating the disparate value streams into a common horizontal layer being called as Value Stream layer to let teams visualize more than a single process area. It not only allows to identify the waste but also helps to see the sources of waste by highlighting the bottlenecks.

Value Stream Management Platform: A typical value stream management platform consists of three layers as shown below in Fig. 5 (The Plutora Platform - Value Stream Management Platform - Plutora.com, 2020.

Integration and Common Data Layer: At this layer, the integration with existing software development Agile DevOps tools as well as creation of common data model should happen so as provide an end to end visibility and bi-directional data flow for the underlying processes, tools etc. The need for correlation of sourced data from existing tools - so that it can be turned into an actionable insights - is important. Product Development and delivery teams need a a common view across entire value stream of correlated data that provides insight across all stages of the product delivery lifecycle, from planning and application development to testing, deployment, and production monitoring.

Business Process Layer: BP Layer Generates views and insights on key processes such as release management, environment management and deployment management. The views so generated are important to monitor and track the processes in an efficient manner. For example, release management view should give a complete picture of the release's activity progress, number and names of applications impacted, release health, test progress, defects at any given point in time, metrics etc

Decision Making and Analytics Layer: This layer should enable design studio for value stream mapping as well

as AI powered predictive insights on the data collated through integration layer.

4. Case Study of US Telecom Services Provider

A large US based telecom service provider was facing number of issues pertaining to release, deployment management processes for their service management platform vTM development program. These issues were obviously having detrimental impacts on the release cycle times, costs and productivity of the product development process. Below table (sample) summarizes the release management related challenges, the use cases and business impacts.

Table 1: Issues in US Telecom Operator's Agile DevOps Implementations

Release Issues	*Use Cases*	*Business Impact*
Parallel release going on	A real time release dashboard with up to date status	Direct impact on the business users working as functionality may get blocked.
Each release has dependency on other	Manage dependencies between releases, applications and association scope	May hinder production functionality.
Common modules being updated for different teams for different releases	Tool to check sign off and lead architect	Chances on missing delivery artifacts impacting incorrect deployment.
Code merging after release	A tool to ensure merge activity completed	High risk of getting incorrect and irrelevant code getting deployed production
Tracking testing completion status for each release	A tool to integrate with existing test tracking tool and provide real time status	Risk of getting untested or partially tested code deployed to production
Question always there whether the modules is ready to go to be deployed or not	Need way to find percentage completion for each planned release	Risk of getting untested or partially tested code deployed to production
Providing environments and resources for parallel releases	Need a tool to provide resource availability and % utilization	Resource utilization fails which leads to delay delivery time lines/ efficiency
Making sure each release through ST, UAT, LSP(Performance environment) test cycle	Need a tool to capture testing phases. Need a tool to show development methodology and governance	Risk of getting untested or partially tested code deployed to production

Source: TechM PoC, 2019

A similar set of challenges, use cases and business impacts were formulated for environment management and deployment management areas

Solution involved mapping 'as-is' value stream, designing 'to-be' by eliminating non-value adds in the process, integrating disparate agile DevOps tools into a common horizontal VSM layer and creating common data model and governance to address the issues and use cases listed above.

On the basis of systems integration, common data model and data storage, monitoring, tracking and reporting of the process was enabled through reporting dashboards.

5. Discussions and Conclusion

Based on the literature survey as well as the practical experiences of running agile DevOps implementations, it was identified that the existing Agile DevOps methodology - although far superior to the earlier ones like waterfall and V-model - had a few shortcomings in terms of not being able to present the real-time end to end view of the value stream leading to difficulties in monitoring, tracking and improvements. Value stream mapping led agile DevOps approach was found suitable to addressing the existing challenges. VSM led Agile DevOps approach although widely used in manufacturing industry is new to the software product development industry and hence proof of concept was planned to prove the concept and the benefits.

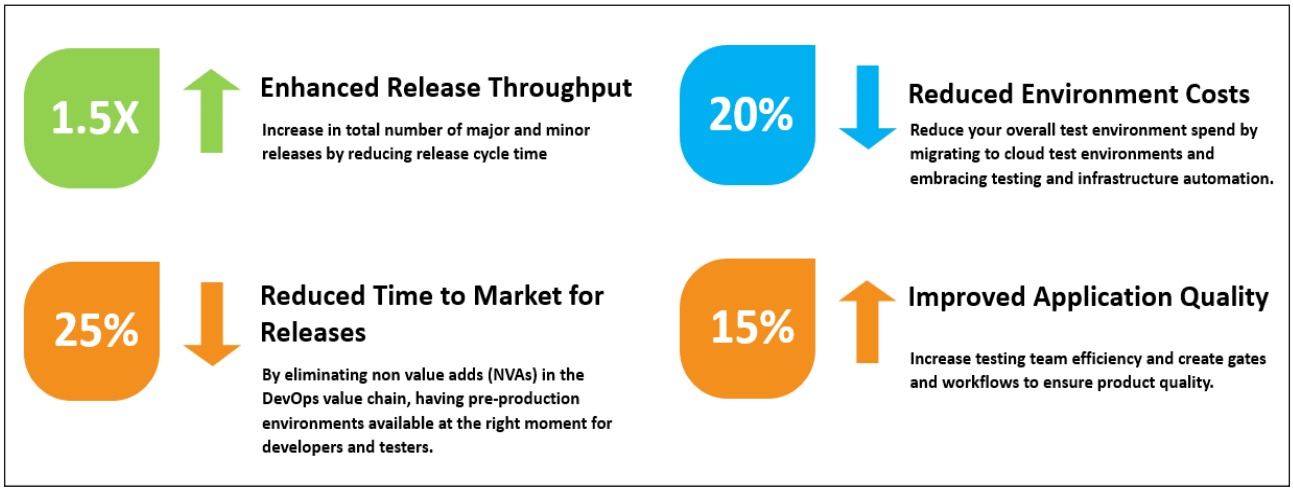
PoC was planned on a Telco service management platform. Plutora tool was used for the purpose of developing the PoC. Value stream maps were developed to identify the value adds. Re-engineered value stream maps removing non value adds (eight wastes of lean) were developed and deployed.

In conclusion the concept was proven - on a smaller scale - and can be easily scaled up for a full scale VSM led agile DevOps implementation accruing business benefits as discussed in section 6.

6. Implications of the Study

The Proof of Concept (PoC) demonstrated the benefits for the trouble ticket management platform for a US Telecom operator, its Service Management department, TechM Platform Innovation teams. A similar benefit could be accrued to a typical product development effort.

The PoC led to a significant (1.5X) improvements in the release throughput, reduced release environment cost (20%) and significantly reduced (25%) time to market for releases.



Source: TechM PoC, 2019

Fig. 6: Business Benefits of Proof of Concept (PoC)

7. Acknowledgements

The authors wish to express their sincere gratitude to Mr. Manish Vyas (President, Communications, Media & Entertainment and CEO, Network Services at Tech Mahindra), Dr Satish Pai (Chief Delivery Officer at Tech Mahindra, Communications, Media and Entertainment, Tech Mahindra), Padma Parthasarathy (Senior Vice-President and Global Head Consulting and Digital Services, Tech Mahindra), Upendra Dharamadhikary (Vice President), Brandon Sweet (Enterprise Account Executive, Plutora, USA) & Richard Winslow (Director of Solution Architecture, Plutora, USA), TechM Team (Sadhieshbabu Dhandapani, Pramod Patil, Hari Krishna Labhe) for their help, support and encouragement during the research and proof of concept development.

References

Balaji, S., & Murugaiyan, M. S. (2012). Waterfalls v-model vs agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, *2*(1), 26-30.

Martin, R. (2003). *Agile software development.* Upper Saddle River, N.J.: Pearson Education.

Wettinger, J., Andrikopoulos, V., & Leymann, F. (2015). *Automated capturing and systematic usage of DevOps knowledge for cloud applications.* IEEE International Conference on Cloud Engineering, Tempe, AZ (pp. 60-65). doi:10.1109/IC2E.2015.23

Ung, A. (2013). Eliminate non-value added activities in your organization. Retrieved from https://flevy.com/blog/eliminate-non-value-added-activities-in-your-organization/

Condo, C., & Lo Giudice, D. (2020). Elevate Agile-Plus-DevOps with value stream management. Retrieved from https://www.forrester.com/report/Elevate+Agile PlusDevOps+With+Value+Stream+Management/-/E-RES142463

Sklavounakis, A., & West, M. (2019, September 30). Maximize the success of enterprise agile: Utilize DevOps as the accelerator. Retrieved from https://www.gartner.com/en/documents/3969871/maximize-the-success-of-enterprise-agile-utilize-devops-

Chapman, Z. (2020, January 20). Scaling agile in large organizations. Retrieved from https://www.clearlyagileinc.com/agile-blog/scaling-agile

Sambandam, S. (2020). Visibility in DevOps - It matters more than anything else. Retrieved from https://www.information-age.com/visibility-devops -matters-123482861/

TechM Research. (2019). Approach to software product development process optimization. Retrieved from https://www.plutora.com/blog/release-management-best-practices

Tech M Proof of Concept. (2019). Approach to TTS product development process optimization. Retrieved from https://www.plutora.com/blog/complete-guide-product-management

Plutora. (2020). The plutora platform - Value stream management platform. Retrieved from https://www.plutora.com/platform

Skhmot, N. (2017). The lean way blog: The 8 wastes of lean. Retrieved from https://theleanway.net/The-8-Wastes-of-Lean