VIEW POINT

TRANSFORMATION TO AGILE– SHOULD YOU TAKE THE BIG STEP?

ELANGO S

SUJEET KUMAR

Infosys®
Navigate your next

Executive Summary

Transforming from a mindset of only developing Software for the given requirements to a state of 'Solving Problems' of the Customer helps to reduce the distance between you and your Clients.

Being agile means being responsive to change, learning while executing and switching on to 'failing fast and fixing faster mode'. Going agile is not an overnight effort; it is a journey of understanding your needs, mapping it to the demands of the clients and delivering the product. It would be wrong to state that following certain steps will be a definite way to achieve Agility as we learnt that there is no "one way" of going agile. Agile is like tailor-made suits, you get one stitched to fit your stature. Going forward we share the transformation journey from waterfall to agile and a preview of how we achieved 'Being Agile' in a leading European Financial institute.

Introduction

Transformation of any organization towards Agile is a perplexing and indispensable step forward to fast development. With the intensification of the agile methodology, many companies took the Agile Manifesto as their navigator to move in the correct direction at faster pace. Google, Spotify, Netflix and many more organization and industries made clear that Agile was not prescribed for any specific industry. Hence, Agile soon started its viral in Communication, Entertainment, Oil & Gas and Financial industry.

In order to survive, "When in Rome, do as the Romans do", makes you to be intuitive and start the very first step in the journey of Agile Transformation. As a leading institute, the huddles, pits stops, checked flags crossed in this race are vital and would aid in our success.

Challenges To Begin Agile Journey

Shifting gear and following a new process takes more than just acceptance. There are many challenges that is either not foreseen or the impact analysis is not completely accounted. Here are few of the challenges that we faced during our transformation journey.

To have an adaptive Mindset

"At its core, Agile is a mindset"

The major setback whenever a change is implemented is the resistance and the mindset of the people. Mostly the idea of transformation and its reception depend upon individual perception and can be acknowledged positively or negatively. The traditional waterfall model had marked duration for each phase of the Software Development Life Cycle. The lack of acceptance to move towards agility leads to lagging deliveries and unhappy teams. Thus, it is very important that changes such as moving from waterfall model to agile, are handled sensitively, all aspects be discussed and individual concerns recognized and clarified. Here at our organization, Dedicated Agile Coaches were hired specially, to enable the teams. Fun activities were conducted to strengthen the team bonding. Various sessions (both in person and in virtual environment) and presentations were planned to make individuals aware of the benefits of Agile and facilitated to be "Agile Ready".

Inflexible Organizational practices:

"Waterfall allows me to do Requirement analysis, Software development, Testing and User acceptance phase wise and each phase has its own acceptance criteria. How will I decide the acceptance criteria in agile mode?"

Waterfall had explicit role definition and respective functions of 'who does what'. When we bring a change in the way of working where in team is encouraged to perform 'any task' in the best possible way they can, the initial thoughts of uncertainty and non-clarity about work and role definitions become apparent. This can add to the resistance in change. Organizational practices need to be maintained along with the transformation from waterfall to agile. Agile also promotes less documentation, however, quality and project maturity should still be maintained. Without proper documentation, the traceability and back tracing is almost impossible. Minimum required documentation and regular internal audits helps in maintaining the project's maturity levels.

Another aspect can be the ease with which the organisation structure allows different departments to move to agile. In our experience, many departments could embrace change and got on the riding train very fast, while some departments were bounded by own set procedures and guidelines which made the possible change acceptance difficult. As a result, cross teams projects were directly impacted due to diverse style of working and due to different ways to priority setting.

We knew this would come, but would come so fast at such large scale was a challenge for us to fathom and face. Development teams could move swiftly with changed standards, while Maintenance team faced slower acceptance due to both, mental and procedural blocks. Working on the transition began late in Maintenance teams, as a result there was a time lag when half of the team would deliver in agile mode while the other half would still execute in waterfall to meet acceptance and approval guidelines. Finding a marriage between old best processes and keeping the spirit of agile intact in acceptance by the teams has been singular most difficult challenge, which doesn't have a 'fixed template' solution.

Resistance in bringing Structural Change:

"Continuous Change is the essence of Agile"

When we move from waterfall to agile, the structural aspect also needs to be relooked into.

Consider the following:

- The (re)structure of the team to adapt inline with the agile mode.
- The roles of each member in an agile team, which might require cross functional capability (a role beyond their general expertise). This also makes the team self-enabled and removes individual dependencies on one hand and the proficiency on the core knowledge area holds value for the delivery, on the other. One of the transformation challenges has been-falling back of agile teams to waterfall mode of execution in absence of cross functional team members. An empowered Agile Team can accomplish more in terms of business goals and each member plays an important role as agility builds stronger bonds within the team. With each sprint, the team identifies its strengths and weaknesses and learns more. Enabling and achieving a cross functional team is one of the most important success criteria for agile adoption.

Technical complexities:

"Not all can fit in agile. Just like the building blocks can make a building but you should know how to put the pieces together."

Understanding your need and then transforming to agile makes more sense than just trying to move to agile mode. A good 'Preparation v/s Implementation' analysis can save a lot of trouble at a later point of time.

Technical complexities such as interdependency with one or more applications and services which may be developed at different timelines has to be brought together and planned in correct sprints. There is huge dependency on the services and applications provided and consumed. Misalignment between them can cause major failures. Integration of multiple systems across different departments of the firm and which are developed in different technologies such as Mainframe, Java, and Package applications should be carefully designed, integrated and tested to achieve the 'right' end result.

Best Practices

"There are no secrets to success. It is the result of preparation, hard work, and learning from failure." -

- Colin Powell

Learning from failures is constant step in achieving success. Few best practices that can be adopted during the transformation are:

Product Increment Planning:

"Let our advance worrying become advance thinking and planning."

– Winston Churchill

Who said Agile does not need a planning! In fact planning becomes more and more vital in Agile as you are in more matured and faster path to delivery. Hence making a road map of the future sprints is essential to achieve your Product goals and helps to identify dependant systems. Hence Product Increment (PI) planning has to be done before beginning of any product milestone.

Preparation(s) Work- Introduce Sprint 0:

One or more Preparation sprints are required before actually starting up new sprints. This helps the team to utilise Sprint 0 for all the groundwork and the actual work can start from Sprint 1 right away. In our experience, quite a good number of teams realized importance of Sprint 0 after experiencing lower than expected velocity or after spending more than expected time in some critical inter dependent user stories. With proper planning and discussions, such situations can be avoided. Before you start ask yourself, is the team ready? At this point, authors would like to caution against 'analysis paralysis' as well. Don't analyse the situation so much that you are not able to act on it; hence there is a fine balance between preparation and execution.

Backlog needs Continuous Refinement- keep up to it:

Convert carefully the requirements to user stories for the new Product backlog as this Backlog becomes the spine of the project. The Product Owner should have clear understanding and should be in sync with business with respect to the requirements and dependencies. Priorities keep switching hence keep track of execution alongside priorities.

Keeping Daily Huddle within bounded time frame:

Few teams spent a longer duration in huddle and extended discussions which should be ideally taken up separately. After a couple of sprints, this was corrected and the huddle was used only for updates. Any discussion between individuals was handled outside the huddle duration.

There are 2 shareable learnings here:

1. Since agile is about few thoughtful processes, the overall spirit of agile adoption gets a beating if those few processes are not followed in letter and in spirit.
2. Actions, like detailed problem discussions during a meeting of 15 minutes, negatively reinforce the team getting into the old comfort zone will not only kill discussions but also demotivate the spirited team members.

Feel free to ask more:

Walk – Call - Email

The more questions you ask the better will be the clarity. During refinement, all relevant clarifications should be done so that the team has a common interpretation of the Sprint goals. Assumption is a killer so no room should be left for assumptions. To avoid any scope for suppositions, demos should be planned with stakeholders and end users at the end of each Sprint, to better the alignment. The lags and mismatches can be handled at the right time, if any.

Achieving Minimum Documentation:

Although agile favours less documentation, there should be minimum artefacts that should be maintained at all costs to have a good traceability and to check requirement coverage.

In our situation, as a result there was concept introduced for test deliverables called as "One-Pager". As the name states all the required test goals, impact, likelihood and results are collated in one page document. Later it is integrated to JIRA tool using a self-made plugin.

Internal and Org-level audits should be conducted from time to time, to see that all adhere to level of quality.

Tools and New Approach: JIRA

This tool is the backbone of Scrum teams. All the logging and tracking is done in JIRA. Product backlogs have the requirements mapped to User stories which are pulled into Sprints based on priority after deep discussion within the Scrum Team and Product Owner. The progress of the User Story is tracked based on the sub tasks and any impediments or bugs found are recorded in this tool. JIRA is used as a one stop tool by several institutions including ours. (It's not an endorsement of JIRA, any similar tool can be considered for agile adoption as needed).

Test Automation:

Automation has shifted from being "good to have" to a "must have". With quick changes in code and rapid deployments, the quality has to keep pace. Automation helps in running test cases and Regression suites quickly.

Jenkins:

This tool provides swift deployments and does a quick check of the unit tests. The package to be deployed is 'assured' defect free and 'confirmed to be' standard compliant. It allows us to integrate various developments and test related tools to create a series of jobs which can automate the integration and deployment of applications.

Shift Left:

SHIFT LEFT methodology is a key enabler in Agile in identification of defects and issues at the earliest possible stage and fix them. This ensures that the focus of testing shift towards the left in SDLC and also acts as a checkpoint of the minimum viable product delivered at end of each sprint.

Service Virtualization:

Service virtualization is a tool to freedom for service development teams, as they will not relay on their consumer/service provider application/service. Through this tool you can create a mock-up of the expected outcome for your service as consumer/ service provider hence letting you work independently and more efficiently.

Conclusion

With the acceptance of agile across the globe, the competition for consumer-centric faster delivery is reaching new heights on daily basis. With further complementary addition of artificial intelligence and machine learning, authors, are of the view that the focus on delivering quality and the fastest pace ever will be getting into non-traditional environments as well.

One of the examples has been comparable time taken for adopting agile (approx. 2 years for a large organization wide implementation) with respect to moving towards Continuous Integration and Continuous Deployment or DevOps (~6 months to 1 year for similar organization).

We believe that Agile can be adopted in any project and in any industry but the teams should be aware of the preparations(which is more about getting mentally ready than procedurally) to move to agile and make a conscious call on their journey devising the team specific customized methods for faster adoption and easier implementation.

About the Authors

Elango Shanmuga Sundaram

Technical Test Lead, Infosys Validation Solutions – Financial Services, Infosys

Elango is a Test Lead in the Financial Services practise at Infosys. With 12 years of experience in IT, he provides test consultancy for both technical and domain level working for a leading European client in Financial Services industry. Along with test consultancy, Elango works on mentoring team and creating knowledge assets for spreading awareness in the industry.

He can be contacted at Elango_SS@infosys.com

Sujeet Kumar

Project Manager, Infosys Validation Solutions – Financial Services, Infosys

Sujeet is a Project Manager in the Financial Services practise at Infosys. With 12+ years of experience in IT majorly working out of Europe, he provides testing solutions for advanced packages for both technical and domain level in Financial Services industry. Along with consultancy, Sujeet is part of Enterprise Content Management team leading input management solution.

He can be contacted at Sujeet_Kumar02@infosys.com

For more information, contact askus@infosys.com

Infosys®
Navigate your next