Lean Change: Enabling Agile Transformation through Lean Startup, Kanban, and Kotter

An Experience Report

***Abstract*—Agile methods for software development and systems development has grown in popularity to the extent that larger organizations have been looking to adopt these methods at the enterprise level. Typically these adoption approaches are implemented as Agile Transformations and there has been mixed results in the successes and failures of these change initiatives. The change tactics used, and the specific methods chosen for adoption vary wildly depending on context. In a nutshell, the only certainty about large-scale agile transformation it is to expect uncertainty. This paper will provide an experience report on a new approach known as Lean Change which has adapted techniques designed to handle uncertainty from other domains to provide a framework for executing Agile Transformations more effectively.**

Keywords—Agile Transformation, Lean Change, Lean Startup, Kanban, Kotter

I. Introduction

Over the last four years I have helped a number of relatively large IT organizations of 200-400 people embark on ambitious Agile IT Transformations in pursuit of delivering better, faster and cheaper. The majority of these transformations were looking for quick wins and real sustainable change within 1-2 years. I work in a large global management consultancy and many of the clients I work with are traditionally conservative IT organizations built around decades of process focused, command and control cultures. Some of them have recognized that traditional waterfall based methods are failing them and are ready for wholesale change. Others are simply looking to optimize their existing processes with a more pragmatic adoption of lean and agile principles. The team of consultants I work with have been partnering with the IT leaders, managers and teams in four separate IT organizations to design, lead and manage Agile Transformations at both ends of this spectrum. Two of these organizations were public sector IT organizations while the other two were financial services.

Throughout our journey, we have faced many challenges as we worked with our clients to enable their goals. As a result, we have continuously adapted and adjusted our transformation approach many times in pursuit of finding a “good enough” answer that could serve as a useful recipe for safely guiding Agile Transformations. Our current thinking has developed into an approach we call Lean Change. It is based on the lessons learned from our own work and our adaptations of other approaches defined by thought leaders in other fields that have found good answers in their respective problem domains. We believe it is still early days for Lean Change but we are seeing significant evidence of its effectiveness in de-risking and accelerating Agile Transformations for our clients. We believe the problem domain of Agile Transformations is still highly immature and the agile community is in need of answers to this challenge if we are to continue to help larger and more conservative organizations seeking transformational change with lean and agile. The hope is that others in the agile community see the value of our approach. Our goal of this paper is to share the thinking that went into the development of it as well as share a set of lessons learned that motivated us to refine our transformation approach to where it is today. Much work remains to enhance this method, but we believe it is a “good enough” state that it will be useful for others embarking on their agile change efforts.

II. Agile Transformations #1 and #2: The case for Lean Change: Building the perfect plan and learning from it

Agile Transformations are hard. They require a successful orchestration of people, process and technology and it is a highly unpredictable process getting from point A to point B in the transformational process. To further add complexity, the target end state the organization is trying to achieve may not be what an organization initially envisioned. During our early transformation engagements, we worked with two separate organizations that truly believed they knew what the target state agile world was going to look like in their organization and they set to work to make it happen. Both organizations took a plan-based approach, spending a significant amount of time (4 - 12 months) defining a detailed target state consisting of a new organizational structure, new agile delivery processes, and new cultural norms for their people. It all looked great on paper. When the organizations initiated the rollout of these plans, they started realizing that something was wrong. Lack of management support became evident as teams surfaced problems through their retrospectives and management failed to respond. Teams adopting the new agile methodology were consistently struggling to collaborate effectively with their business partners resulting in impediments in requirements definition and failing to deliver shippable code frequently. As these challenges and others emerged, the organizations continued to follow the perfect plan and seek bandaid solutions such as hiring more external coaches or consultants to show the organization how to do it right. After two years, both organizations are still struggling to make the agile transformation work. However, both have learned that their perfect plans were not working and are now revisiting what went wrong in order to move forward again. “Stop the line”

978-0-7695-5076-3/13 $26.00 © 2013 IEEE
DOI 10.1109/AGILE.2013.22

mentality finally set in, and both organizations are now learning. Reflecting on these experiences, I believe there are a number of reasons for what went wrong but there is one overarching theme to be learned from these experiences. There is no such thing as a perfect plan, and trying to plan your way to success in an agile transformation is a recipe for disaster. From our experiences with these types of transformations we have seen two other organizations that took on a drastically different transformation approach see much greater success. My hypothesis on why they have had significantly better outcomes in their agile transformations is because they focused on learning their way to success instead of following a perfect plan.

A. First Actionable Insight: Accelerate the feedback loop through experiments

Taking our lessons learned from previous work with clients, we set out to take a different approach towards agile transformations and challenging how we think about change. Instead of focusing on planning as the way to reach success we turned to learning.

Reflecting back on Agile Transformation #1 and #2, we compared the two and realized that the second organization fared slightly better than the first because they responded to feedback faster in the transformation program and were able to learn slightly faster about what was working and what needed to change. At the same time, we luckily stumbled upon the Lean Startup movement happening that was being distilled by Eric Ries and others from the startup community. This helped us connect the dots on a big a-ha moment, asking ourselves what if we ran transformations like a startup?

Using this as the basis for our future learning based approach we set out to map the Lean Startup model to how we view transformations resulting in the finally thinking. If we look at how the Lean Startup method views product development, it tells us that startups operate in a world of uncertainty and risk. A popular and simple view for a successful startup is one that successfully navigates through 3 key types of risk:

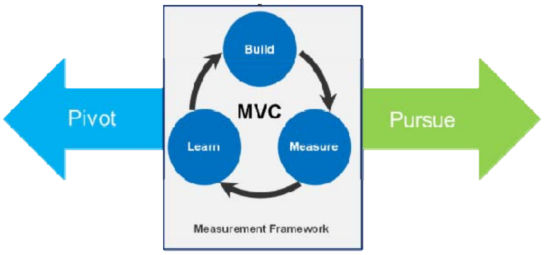
- Product Risk - will you build the right solution for a problem worth solving
- Customer Risk – will you build an effective path to customers who care
- Market Risk – can you find a market that will support a viable business

Startups fail when they spend too much time building a product that nobody wants. If we map this problem domain over to the transformation and change space, it looks like this:

- Change risk – implementing a change that will solve business problems
- Resistance Risk – implementing an approach that result in successful adoption across the organization
- Sustainability Risk – getting the right commitment necessary to achieving change benefits

Change initiatives fail when they introduce a change that does not have buy in at all levels of the organization.

Therefore the overall emphasis of the learning based approach needs to enable us to run change experiments fast to accelerate learning on validating change, resistance and sustainability risk assumptions inherent in the agile transformation plan. This model from Lean Startup would form the basis for our transformation approach. Instead of building a Minimum Viable Product to validate product assumptions, we would frame future transformation changes as Minimum Viable Changes (MVC) to validate the transformation assumptions.



III. Agile Transformations #3 and #4: Running Change Experiments: Applying a Learning Based Approach to Change

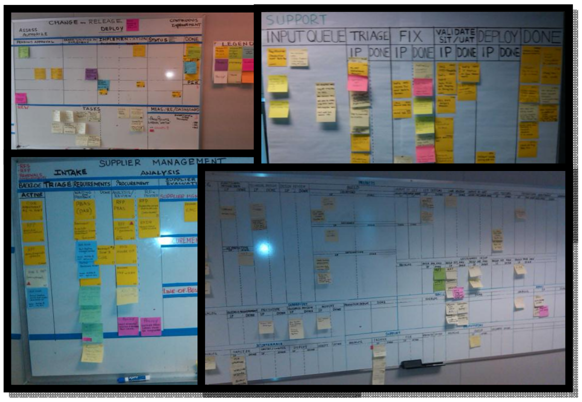
In the last two years, we have worked with two organizations to approach agile transformations using lean startup principles. The emphasis is on experimentation and embracing the assumption that we are likely to be wrong with our target state. Furthermore, this approach accepts that we are also likely working within a complex or chaotic domain. Theory aside, in order to provide some evidence based support that this approach is viable, we have collected some metrics that indicate these organizations are seeing significant benefits by applying a learning based approach.

For both of these agile transformations we worked with people at all levels of the organization to define, validate and refine each change as a Minimum Viable Change experiment. We found we were able to inspect and adapt much faster than previously. Instead of taking 6 – 12 months to pivot and adjust it would take us 1 – 3 months with our new approach. At first we defined every change action we were doing as a Minimum Viable Change and adopted a simple format to ensure consistent thinking and structure across the change agent team. The format consisted of a structured hypothesis, a fixed duration with a set of predicted behaviours from the change recipients. One of our first Minimum Viable Changes was running a two day Kanban training class for 40 people in one of the organizations. Our hypothesis was that we would identify 4 or more eager adopters that could serve as our initial agile change champions with their respective teams. The first training class was a success and we easily met our hypothesis and also learned something new. There was significant enthusiasm from the trainees to start adopting Kanban on the job right away. Based on our lesson learned from previous agile transformations we set out to respond quickly to feedback, and focused on meeting this new demand and leverage the momentum building up in the organization.

A. *Second Actionable Insight: Kanban as an x-ray to gather information fast*

In response to this new insight, we pivoted from our initial plan that focused on initiating three focused pilot projects and getting Kanban up as fast as possible for as many teams as possible in order to leverage the adoption momentum. This pivot resulted in the second MVC which was launched as a self-starter program for these eager adopters. The self-starter program was designed for teams to learn on their own supported by a regular cadence of "open office hours" from our coaches where any team could come in, bring their work and get advice from the coaches. In order to execute on this program, we prepared a set of self-starter guides and focused the curriculum on "as-is Kanban" which encouraged teams to not change their process on day one and instead focus on visualization of work while limiting work-in-progress. We felt "as-is Kanban" would be a safe and effective learning method for these eager adopters to bring more agility into their process without significant disruption. Within 8 weeks, over 50% of the organization had started adopting Kanban and our team was able to gain vital insight into the inner workings of the organization by simply walking the floor and observing the Kanban boards and attending the daily stand-ups each team was having.

"As-is Kanban"



As a result of this MVC, the next insight become apparent from the Kanban boards due to the number of work items with blockers and the lack of flow on each board which indicated that there was significant silos across teams, lots of functional specialization and lack of consistency in their delivery approach.

B. *Focusing back on pilot projects: Hopes of building a reference kernel for the organization*

As we discovered the challenges in communication, collaboration, and discipline across the organization we realized that the organization needed to grow a set of eager adopters that could act as guiding teams to demonstrate what collaboration and disciplined agile delivery looked like for the rest of the teams. Our team worked with the IT Managers to identify three pilot projects that could serve as the launch pad for growing the guiding teams and we kicked each one of as a new MVC with the hypothesis that these teams could adopt a baseline set of agile practices, enable standardization of a kernel SDLC with focused coaching from our team. After 8 weeks we started learning that two of the three projects were having challenges in adoption due to multi-tasking on other projects/support work, unwillingness to revisit previously completed requirements prior to the pilot start, and majority of the developers significantly lacked the skills to effectively communicate and collaborate with other functional specialists (i.e. business analysts, testers, etc.). In response to this new learning we defined a new MVC focused on untangling the multi-tasking and looking for a solution to hopefully stabilize demand based on capacity to lower the level of multi-tasking.

C. *2.3 Starting the Enterprise Kanban: untangling the multi-tasking problem*

Our team focused our next effort on leveraging Kanban to visualize the entire IT project portfolio and set constraints on projects by limiting work in progress at the project level. The setup of this Enterprise Kanban was structured as a MVC focused on validating the assumption that visualization would motivate the IT executives and managers to address the overflow of projects that were in-progress and the organization would start limiting the amount of projects being taken in and also slow down or stop existing projects as required based on finishing priority ones first. After the setup of the Enterprise kanban, we encouraged executives and managers to regularly meet at the board to prioritize active projects and address impediments to these projects. The executives and managers responded positively to the Enterprise Kanban and saw it as a tool for risk, issue and blocker escalation and resolution so they could better monitor the project portfolio. In response to this insight, we defined a new MVC for the Enterprise Kanban which was a concierge service we would provide to facilitate stand-ups 3 times/week with all management in IT to discuss and resolve problems. During each of these stand-ups, risks, issues and blockers for all projects in the IT portfolio were discussed and to support this discussion we created a Risk/Issue/Blocker Kanban board next to the Enterprise Kanban. This new Kanban was used to visualize any risks, issues or blockers identified, and were pulled into a "Resolution In-Progress" column if one or more executives/managers felt they could resolve it. As a result of this MVC we learned that the organization was very keen on these stand-ups and saw tremendous value as it provided the only forum for this type of collaboration. As the executives and managers grew accustomed to the stand-ups, it become transparent to them that the organization was overcommitted in projects due to the amount of blockers being identified from the lack of people resources. This realization triggered a set of focused prioritization workshops over two weeks with the executives and managers to force rank projects based on available capacity. The projects that were considered lower in the ranking and lacking capacity to work on them were handled by sending out a formal communication to the business partners on the impact of these projects and a plan was put in-place to quickly bring in external resources to temporarily increase capacity in order to meet these project commitments.

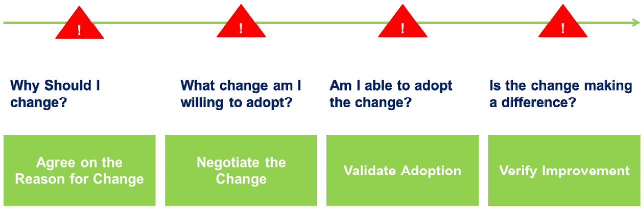
“Enterprise Kanban”

D. 3.0 Improving the Learning Approach: Blending Kotter and Minimum Viable Changes

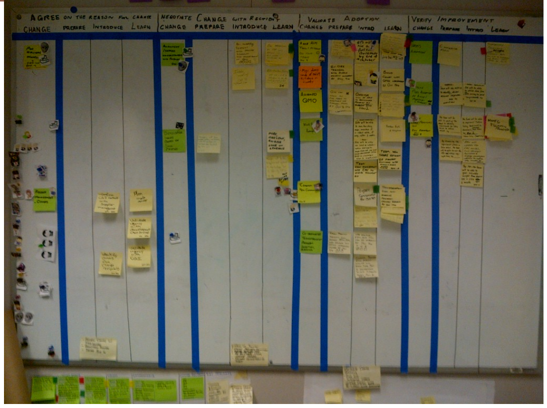
The next evolution of our learning approach came as the organization invested in building up an internal change team that could help focus on sustaining the continuous improvement culture. We felt we had done a good job applying the Lean Change approach within a change agent team of five but as the team grew to eleven with new faces and change agents with different backgrounds we wanted to provide some additional structure to encourage discipline in the process. In reflection we looked back and realized many of our MVCs got stuck at times and adoption stalled. Our analysis of why this was occurring revealed that some MVCs were started prematurely without buy-in from the people involved and others failed because the adoption stopped after a short period of time. To solve this problem, we set out to make the lifecycle of an MVC more explicit. From previous agile transformations with other organizations, we felt that the Kotter 8 step approach has served us well as a guide and looked for a way to miniaturize it so it could be applied within each of our MVCs. We also noticed that Ash Maurya did a great job distilling the Lean Startup approach further into a lifecycle process to guide startups and that inspired us to do the same for our learning method so we could bring more discipline into how MVC experiments were executed. As a result, we formulated a four step lifecycle for MVCs which were:

- agree on the reason for change
- negotiate the change
- validate adoption
- verify improvement

Each of these steps was designed to address a common failure mode we observed from our previous MVCs.

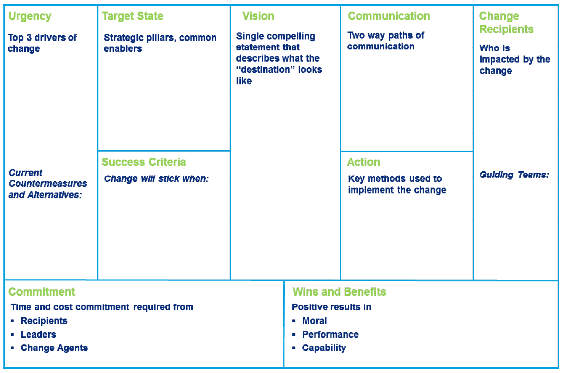

!
!
!
!
Why Should I change?
What change am I willing to adopt?
Am I able to adopt the change?
Is the change making a difference?
Agree on the Reason for Change
Negotiate the Change
Validate Adoption
Verify Improvement

To help coordinate the larger team a Kanban system was used to run the set of MVC’s through the lifecycle to help limit work-in-progress and visualize problems within our experiments. The Kanban board consists of four states, each state representing one of the steps in the four step MVC lifecycle outlined above. A set of policies were defined with the extended coaching team to ensure understanding and discipline for how each change agent conducted their changes.



E. Learning from Business Model Generation: Use a simple tool to co-create the change with your change recipients

At this point of the transformation, the organization was making strong progress towards adopting agile in a way that was effective for them and our team started shifting our focus on transitioning the ownership of sustaining the change over to the internal change team. One of the challenges we noticed in adopting the MVC lifecycle process was the lack of a structured technique in moving the MVC through the four steps. Each change agent would have their own particular technique in how to execute their MVC. As a result, we introduced another mini-innovation to the method by using a common technique that ensured each change agent executed their experiment in a similar way. In particular, we wanted to ensure that change agents had an effective technique to co-create each MVC with their change recipients. Borrowing the idea from Business Model Generation, we adapted the business model canvas to address dimensions within the change world as a Change Canvas.



Urgency
Top 3 drivers of change
Current Countermeasures and Alternatives:

Target State
Strategic pillars, common enablers

Success Criteria
Change will stick when:

Vision
Single compelling statement that describes what the "destination" looks like

Communication
Two way paths of communication

Action
Key methods used to implement the change

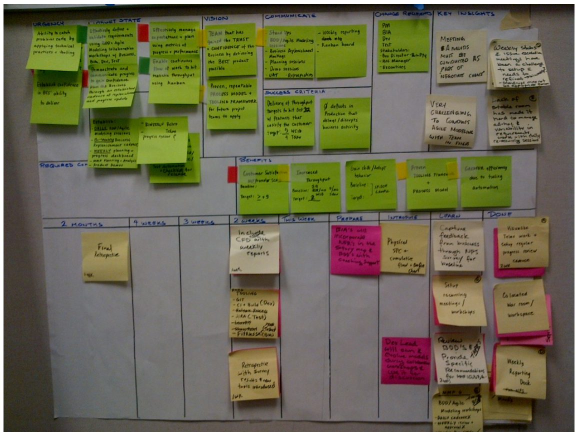
Change Recipients
Who is impacted by the change
Guiding Teams:

Commitment
Time and cost commitment required from
- Recipients
- Leaders
- Change Agents

Wins and Benefits
Positive results in
- Moral
- Performance
- Capability

Each dimension of the Change Canvas is designed to validate a key concern for the MVC. By combining the canvas with the MVC lifecycle based on Kotter we were able to scale up the change team effectively by enabling a smooth transition process on the change approach being used in the transformation.

"Sample Change Canvas"



F. *Lean Change as a framework for change recipients to own their changes: Choose your own adventure*

In our latest agile transformation we have started experimenting with a new way to execute change that increases the overall organizational ownership of the agile transformation. As we now have a stable transformation approach consisting of MVC's, the lifecycle and Change Canvas, these changes were still driven by change agents rather than the teams themselves. We are now turning the change process over to the organization itself as a way to engage managers and eager adopters to self-own components of the transformation plan. To make this work, we identified a set of change areas that were better owned by the organization than by external change agents and identified change owners within teams which we ran workshops with to show them how the change lifecycle worked, how to populate the canvas and engage other people in the organization for co-creation, and finally to execute the change themselves. From our early experiments with this, we are seeing strong adoption and the managers of the organization have built a number of change canvas' that has helped them define a target state for themselves and aligning other teams within the organization on a common vision.

IV. What's Next: The next frontier

From our experiences in evolving and applying the Lean Change method across four separate agile transformations we have seen it help bring structure, discipline and feedback into our change efforts. The formalization of changes as experiments structured as Minimum Viable Changes guided by a MVC lifecycle process has helped our team take a learning based approach to change while also helping to provide a consistent change approach when working with a larger change team. We believe it's still early days for this method, and there are a number of improvement opportunities to make it better. As part of our work, we have evolved a set of qualitative metrics we use to validate adoption for specific Kanban and Agile practices and we see this as an open area for improvement. The question still remains on what are the best measures change agents can use to empirically validate. Another area of improvement is the definition of the MVC's. Defining a "good" MVC is challenging and we have started to see patterns emerge for common MVC's that we tend to gravitate towards across our transformations. We believe there is a potential pattern catalogue for MVC's that could be formalized to guide other change agents. As well, the Change Canvas has been a very effective technique to structure the changes as they move across the change lifecycle. However, we have seen other change agents apply different techniques as a replacement for the Change Canvas such as A3's which is another change technique borrowed from Toyota. The Change Canvas we advocate for Lean Change is likely to be one of many options that can be substituted depending on the context. Finally, the last area of interest we see with the Lean Change method is that there is likely to be applicability for any type of organizational change effort and not just agile transformations. While this remains untested, we see potential for this method to work in these contexts as well.

Acknowledgment

I would like to acknowledge Jeff Anderson who is the initial creator of the concept behind Lean Change and has been the primary driver for many of the concepts behind this method. Together we have been working to apply it and evolve

it along with other members of the team which include Taimur Mohammad and Andrew Larosa.