

Scaling Agile: A Tale of Two Transformations

STEVEN A MARTIN, PRINCIPAL AGILE CONSULTANT, BIGVISIBLE SOLUTIONS, INC.

In two recent Agile transformation engagements at separate organizations, both companies invested heavily in introducing and scaling Agile across their respective enterprises. While both companies are in rather conservative and mature industries, they both wanted to transform so that they could prevent severely dissatisfied customers from leaving, dramatically improve quality, and have better throughput, enabling them to respond to marketplace demands quicker. One organization excelled in their transformation, and the other did not. While there are many integrated, moving parts to a transformation, in this paper, the importance of setting a strong Agile foundation to enable scaling to occur is presented first. Then, the approach and results from a successful and not-so-successful attempt at scaling is discussed. Finally, a list of questions is provided that any company should answer before embarking on an Agile transformation and again when scaling.

1. BACKGROUND

To provide a more rounded picture of the Agile transformation journeys for the two companies described in this paper, it is helpful to know about their sectors, structure and culture.

The first company is in the finance sector. They create software used by other financial institutions; all of their customers are all external to the company. The second company is in the insurance sector; the particular division involved in the transformation is related to healthcare.

Their software is used by medical staff both internal to the company as well as externally.

Table 1 below provides a brief comparison of the two companies.

TABLE 1: COMPARISON OF COMPANIES

	FINANCE COMPANY	INSURANCE COMPANY
# Agile Consultants	Up to 4	Up to 8
Roles	Agile Coaches	Agile Coaches, ScrumMasters (SM), Product Owners (PO), Quality (Automation), Developer
Length of engagement	15 months	12 months
Product & Customers	Software used by financial institutions	Software used by medical professionals to review patients' healthcare information
Internal or External Customers?	External customers only	Both internal and external customers
Technology	Mainframe based, heavy back-end	Primarily web based
# Agile Teams	7 Scrum, 1 Kanban	3 Scrum, 1 Kanban

There are several similarities between these two companies. From a product perspective, they both create software used by experts within their own industries – one might think of it as “Business to Business (B2B)” software. So while the software is not a commercially available product used by a wide public audience, each has a significant presence within their own segment.

Both companies were experiencing the same types of issues, such as:

- Severely dissatisfied customers; they were ready to leave
- Solutions not addressing customers’ needs
- Long lists of enhancement requests, from a few months to several years old
- Low quality (in the form of high defects)
- Long development and testing cycles

There were similar cultural issues as well. For example, both are publically traded and are under various forms of regulatory compliance mandates, which leads them to be more conservative. Management was also typically promoted from within (for example, this person is a great technologist, therefore they will be a fantastic manager, too), as opposed to having experience and/or training in management concepts. In the case of the finance company, a vast number of employees had significant tenure at the company, generally 10+ years. In the case of the insurance company, an overwhelming percentage of employees were contract workers. Combine these concepts together and there was a great deal of inertia to overcome for *any* change initiative.

Overall, both companies have been around for decades with mature products; there was little desire for significant innovation. While people were quite busy working, it was to service direct, pointed customer requests, fix bugs, and address technical debt versus being leaders in innovation and introducing the next generation of products, and, correspondingly, the next round of profits to keep them in business.

2. APPROACH USED

To overcome their issues, both companies invested quite heavily in what was termed as an Agile transformation over a 12 to 15 month time frame.

We employed a parallel path, three-tier model, using Agile to roll out Agile across the organization. So, this meant that we had three focused “teams”, one for Executives, one for Management and then the Agile Teams themselves, each working on their own set of prioritized respective objectives from backlogs, with all the backlogs aligned towards common organizational goals. See Table 2 at the top of the next page for a summary.

For example, the Agile Teams would focus on delivering high value, high quality products using primarily Scrum. Management would then focus on organizational-level actions to enable Agile Teams to obtain their objectives. Executives would work on things to help the company transform, such as reestablishing relationships with dissatisfied customers while also being the model for the cultural change required.

TABLE 2: THREE-TIER MODEL OF ROLES NEEDED FOR TRANSFORMATION

ROLE	TYPICAL TITLES	FOCUS AREAS
Executives	Leaders of the organization, such as: <ul style="list-style-type: none"> • C-levels • Executive Vice Presidents • Managing Directors 	<ul style="list-style-type: none"> • Set vision, guideposts, and culture for Agile transformation <ul style="list-style-type: none"> ◦ Is consistent and transparent on why they are doing what they are doing ◦ Empower Management and Agile Teams, then get out of the way • Renegotiate contracts and relationships with their customers' Executives
Management:	Typically directly manages staff on teams <ul style="list-style-type: none"> • Associate Vice Presidents • Directors • Senior Managers 	<ul style="list-style-type: none"> • Paves road for smoother Team execution by removing organization-wide blockers <ul style="list-style-type: none"> ◦ Empower Teams, then get out of the way ◦ Regularly shows up to reviews and are "present" to give valued feedback
Agile Teams	Individual contributors	<ul style="list-style-type: none"> • Deliver high valued, high quality working product • Alter direction as needed based upon feedback from stakeholders

2.1 AGILE TEAMS

Much has been written on the subject of Agile Teams and transformation, so only a brief overview will be covered. The following provides some context of how we set the foundation for Agile Teams to be successful, especially during the first few months of each engagement.

Agile Teams (herein referred to as "Teams"), by and large, had roughly five to nine people, were self-contained (i.e. had all the roles needed to produce working software), and initially assigned more than 80% of their time to the Team (the remaining 20% for "other" activities, such as paid time off, department and company meetings, etc.)

Scrum was chosen in most cases since the "real" requirements/business drivers were unknown. It was desired to increase customer satisfaction by being visible with what was being built and why. Demos of working software to their customers were essential to (a) show progress and (b) get feedback, all on a regular cadence to build confidence in the teams' abilities to produce quality product. This significantly lowers risk of delivering a low value, low quality product, as was the status quo. Similarly, teams also generated feedback during sprint retrospectives, so that they could make adjustments to their own processes and ways of working.

Sprints were two weeks long. This was long enough for meaningful work to get done by the Team, yet short enough to provide focus and force limiting work in progress.

Each Team was assigned a coach for the first six to eight sprints to help them get as far and as quickly through the "form-storm-norm-perform" cycle as possible. The first sprint was typically "run" by the coach with the clients' SM and PO observing. The second sprint was run by the clients' SMs and POs. The first few sprints, coaches placed emphasis on ceremonies, artifacts and roles. After the core Scrum mechanics were understood (maybe not mastered, but at least understood better), emphasis in the fourth sprint and beyond were on principles and culture. Over time, the coach tapered down to mere check-ins and was on-call for issues as pulled in by the Teams.

2.2 MANAGEMENT AND EXECUTIVES

To help the management and executives transform, an “Enablement Team” was created at both companies near the beginning of each transformation engagement. Members consisted of managers acting as team members with executives in the role of stakeholders.

With so many unknowns and an abundance of work to do with the transformations, it made sense to use Agile to roll out Agile across the organization at both companies. And, since Scrum was used for the Teams, we also wanted management and the executives to experience the same transformational journey they had expected from their Teams. So, this meant that the Enablement Team would be set up to operate using similar Scrum roles, ceremonies and artifacts, using feedback to adjust. With management and executives getting hands on experience themselves with Scrum, it helps set a great foundation and understanding of Agile from the outset of any transformation.

The composition and structure of the Enablement Team at both companies were also quite similar. There were about 12-15 managers and roughly 3-5 executives involved. We started their journey by holding a two-hour kickoff with up to 20 managers and executives present. This was followed by an all-day workshop where specific issues/problems were identified and prioritized. Epics and stories related to transformation were then written, which helped establish the initial transformation backlog. (Section 2.3 covers the transformation backlog in more detail.)

Recognizing that up to 20 people was too large of an Enablement Team, we had the managers self-organize into sub-teams of approximately five people, each with their own PO and SM. Executives would not be on sub-teams, and serve as stakeholders to the overall Enablement Team. One sub-team would focus on cross-team technical deployment issues, another on identifying/setting up/supporting Agile teams, and another related to their products/portfolio/roadmap. With smaller, focused sub-teams, this eventually helped the Enablement Team make some sort of more meaningful progress rather than everyone working on everything at once.

So, with the sub-teams established, the Enablement Team POs would look at the transformation backlog, decide which epics to bring into their sub-team, and then decompose into stories related to organizational transformation and Team enablement efforts. These transformational stories would be worked on and demonstrated every two weeks by the Enablement sub-teams.

2.3 ENABLEMENT TEAM BACKLOG TRANSITIONS OVER TIME

Epics and stories related to transformation in the Enablement Team backlog towards the beginning of each engagement were more focused on getting a pilot Team or two off and running. They were fairly tactical epics and stories, such as choosing the “right” product/program to start, selecting team members, ensuring proper physical environment setup to enable collaboration and coordination of teams, technical environment availability, communications and getting buy-in with customers on upcoming changes, communications within the company of upcoming changes, training for leaders (how to be a Leader in an Agile world), etc.

As the engagements progressed over time, the epics and stories in the backlog transitioned to addressing enterprise scaling issues. Some examples of epics/stories include prioritizing a portfolio of programs/projects based on value, limiting organizational work in progress (WIP), aligning work cadence in hybrid environment consisting of Agile and waterfall teams, adjusting release cadences when multiple teams contribute to product (more frequent, iterative, incremental releases), ongoing communications within company, reorganization of staff/teams to enable faster work through organization (i.e. creation of SWAT teams to focus on urgent issues while product teams focused on new products and enhancements), metrics, and so forth.

2.4 SET YOUR FOUNDATION BEFORE YOU SCALE

Many companies try to immediately scale Agile across an enterprise before a foundation for operating in an Agile environment is reasonably set. They typically first focus on setting company-wide standards, processes and metrics out of the gate. As we'll see from the successful company, they took the time to first learn what worked for them and then scaled from there.

From my experience, it takes several months (3 to 6 months) at the start of any transformation to set the stage for operating in an Agile mindset. This means that there should be an emphasis on launching a small set of Teams, piloting various approaches and techniques, using empirical evidence and real results to identify what works and what doesn't, and then adjust. Furthermore, the managers and executives should be resolving pointed, specific conflicts for the pilot Teams (not the entire organization) with somewhat of a tactical filter turned on. Simply put, the organization needs to learn how to work under an Agile mindset with pilot Teams before putting in grand scaled solutions for an entire organization. It becomes too overwhelming to try to scale from the outset.

In addition to the approaches shared in sections 2.1 to 2.3 above, two additional topics must be mentioned to ensure a more solid foundation. These concepts, which are covered in the appendix, relate to team selection and getting timely, valuable feedback.

Appendix A describes what I call the "Willingness-Capability" matrix. I have a strong preference for people involved in any transformation (which includes scaling efforts) to be willing to change *and* have the technical expertise and/or domain knowledge to make the change happen. Recognizing that this is an ideal (and it is not very likely to have an entire team consist of these folks), my second choice would be for someone to be willing to change – in general, technical skills can be learned; the ability to change is much harder to "coach." This goes for Teams, management and executives alike.

Appendix B describes the concept of a User Advisory Group (UAG) for gaining feedback to ensure that the right product is being built and to increase trust by showing progress on a regular cadence. It is not an unusual concept for companies to use focus groups to get feedback. What is different from our approach is to get the Team members as close to the customer as possible, to have multiple customers in the same UAG at the same time, and to tap the UAG multiple times during the build (not just at the end once the "final product" has been built).

3. INITIAL TRANSFORMATION RESULTS

As mentioned above, Teams, by and large, follow a typical team growth path. Teams at both companies generally produced potentially shippable code along with reasonable levels of documentation for training and support. They generally were beginning to collaborate nicely within their Team and with their UAGs. This seemed to be consistent for Teams for a few reasons:

- Teams are working on tangible outputs – results (or failures) can be seen quickly
- Teams tend to be tightly focused in a specific area or feature
- Teams typically contain individual contributors

By contrast, management and executives at both companies had a much more difficult time getting started and showing results. Initially, they tended to fall back and focus more heavily on control of processes (for example, heavy emphasis on metrics and governance) and the traditional tactical aspects of running projects (i.e. holding teams to original estimates established by them, not the teams, typically six to nine months earlier, etc.). It was desired to have a detailed roadmap/project plan of exactly how the rollout and scaling would happen. These habits are hard to change. Table 3 summarizes typical attitudes/behaviors observed or espoused by managers and executives at both companies.

TABLE 3: TYPICAL ATTITUDES SEEN BY MANAGERS AND EXECUTIVES AT BOTH COMPANIES

TYPICAL ATTITUDES/BELIEFS BY MANAGERS	TYPICAL ATTITUDES/BELIEFS BY EXECUTIVES
<ul style="list-style-type: none"> • There is less degree of certainty of what I am to do as manager in an Agile world. What is my role? My responsibilities? So, let me fall back on what I <i>know to be true</i> as a manager. • These are my people, my staff. Regardless if they are on an Agile team, I will pull them out when I need them for as long as necessary to solve my problem. • Let’s wait until next quarter to start Agile when things are going to be quieter (but, it never is...). • What you don’t know is that I have been promoted from within the company. I’ve tried it all, and I know what’s worked and what hasn’t. There is no reason to try it again. • My performance rating is based on my group’s performance. Working under Agile principles contradicts with my compensation. 	<ul style="list-style-type: none"> • Sounds great in theory, but prove to me this works. • I’m unwilling to let go of stuff in-flight. • I frankly don’t care how it gets done, just as long as everything I originally requested is done by the date I need. • I understand with Agile I’ll get more output faster with less people. • What do you mean you can’t tell me exactly next year when it will be done? Under the old way, you gave me a firm end date (which subsequently moved...). • How can I possibly renegotiate anything already promised and/or in contracts already? Need to wait for all contracted work to finish before we can do Agile. • My performance rating is based on my group’s performance. Working under Agile principles contradicts with my compensation.

So, while the teams typically launched fine (yes, there was conflict, but most situations were short lived and addressable), it wasn’t until four to six months into the transformation efforts that the management and executives completely fell apart, especially when we were transitioning from pilots and small numbers of teams to impacting (and disrupting) the wider organization.

The difference in what made a successful leap in transformation from pilot Teams to scaling across an organization happened next.

4. SCALING BEGINS AT THE FINANCE COMPANY

About six months into the finance company engagement, there were four Scrum teams (two at one site, two at another site) and one (unofficial) Kanban team launched. Since the Teams were delivering and the customers seemed to be happier, management turned their attention away from Agile to other more pressing matters.

The Enablement Team at this point wasn't as crisp or clearly defined as listed in section 2.2. It was simply a collection of about 12-15 managers and three executives. When they did meet, it was a status update, typically of how the Scrum teams were performing. The meeting agenda was to read through and update a typical risk/issue log top to bottom. Every meeting. Executives and most managers stopped attending. Something had to change.

Realizing that efforts had “stalled” (little incremental ROI for amount of effort being put into the Enablement Team) and that there was a desire to add up to five more Scrum teams, the executives unanimously and vocally made the call to re-launch the Enablement Team.

This time, the Enablement Team would have a prioritized backlog of items that would help increase *organizational* adoption. So, the Enablement Team went through similar workshops and training as Scrum teams, to establish a product backlog of transformation stories. The Enablement Team, realizing there were three “groupings” (or types) of work, self-organized into three sub-teams each with their own backlog, ScrumMaster and Product Owner. The sub-teams focused on (1) technical capabilities to deploy faster (e.g. continuous integration, continuous deployment, automated testing), (2) setting up/helping Agile teams in flight, and (3) product planning, product roadmaps, etc.

With this structure in place, the Enablement Team began to solve real problems that helped the Teams perform even better. It was found that while five additional Scrum teams were desired, just three more Scrum teams were eventually added (there were challenges for transitioning other two teams). A ScrumMaster Community of Practice (CoP) was established to share and “standardize” practices across teams. For example, a common structure/template used for all Teams' sprint reviews was created by the CoP, so that the presentation of working software by Teams followed the same path from demo to demo, making it easier for managers and executives attending multiple reviews to follow them more easily. A Scrum of Scrums was also created to address cross-Team impediments (such as data issues, system uptime, etc.). This was less successful, since items addressed in Scrum of Scrums seemed redundant to the Enablement Team backlog and activities in the ScrumMaster CoP. So, the Scrum of Scrums eventually fizzled out.

To facilitate greater coordination of product portfolios, a Product Owner forum was created. This consisted of a “Chief Product Owner” that worked with several POs from Teams, coordinating product backlogs and helping them prioritize stories across Teams. They typically met on the “off-weeks” between sprint reviews. Over time, the POs directly themselves began to collaborate and identify cross-Team dependencies, sequencing stories roughly three or four sprints in advance across all Teams. (Note: while stories were sequenced several sprints out, there was understanding that this was highly subject to change based upon feedback; it was not seen as a firm multi-sprint “project plan.”) This visibility helped to reduce risk by lowering potential for cross-Team code overwrites and addressing dependencies sooner than later. Sometimes, in the PO forum, stories between Teams were swapped, since one Team may already

be working on that part of the system, and it made more sense to trade stories instead of another Team learning it at that point in time.

Furthermore, the Enablement Team promoted/communicated activities related to Agile (and further learnings about Agile) via monthly company newsletters, a SharePoint site with a repository of resources, and included updates as a recurring agenda item in the weekly company-wide management meetings. Roles within the company were being funded (ScrumMasters, Product Owners), versus relying on supplemental external contract labor. Common tools (such as Jira) and common metrics for all teams (i.e. velocity, burn-down charts) were being rolled out.

The executives renegotiated contracts with two key clients, acknowledging shortcomings and genuinely approaching and engaging their customers in establishing partnerships. The executives also sponsored (at no charge to their clients) workshops led by us as consultants to their clients at their site to help them become familiar with working in Agile ways. They also provided shield to the management and teams when external clients attempted to shift back to old contacts and “waterfall” behavior. One executive even gave up an approved new hire requisition so another executive could hire a key role.

All of these activities helped the finance company scale from a few pilot Teams to something more organization-wide. The scaling, and hence greater organizational transformation, was done in pieces, iteratively and incrementally, only after a solid base of tactical performance had been achieved by the initial pilot Teams. By aligning efforts from Teams to management to executives, the Teams delivered above and beyond the expectations of the clients while also repairing delicate relationships. This could be done since the managers and executives (1) made time for the transformation, (2) became invested as servant leaders for their Teams, (3) were focused on business outcomes as an organization. In essence, the folks at the finance company were *being* Agile, not *doing* Agile.

It wasn't all rainbows and puppy dogs, though, for all managers. Some continued to resist, both vocally as well as subversively, focusing on old metrics and data that had little to no relevance any more, trying to prove that Scrum was more expensive and took longer. Over time, the role of these managers was lessened on the Enablement Team, so they could focus on other issues outside the scope of the Agile transformation.

5. THE (NON) TURNING POINT ARRIVES (PERSISTS) AT THE INSURANCE COMPANY

With the success at the finance company, it was decided to try to introduce the concept of the Enablement Team (with sub-teams) at the outset of the insurance company engagement.

So, an Enablement Team of managers was also created and implemented in a similar fashion to the finance company. This time, we did it from nearly Day 1 (not 6 months in...). We guided the Enablement Team to create a backlog of work with prioritized stories related to enabling an organizational transformation and sub-teams to pull in work. We also had an SM and PO for the Enablement Team identified.

However, in retrospect, commitment for the change was severely lacking; there was little to no buy-in. There was an executive “on loan” from another business unit acting as the transformation sponsor. There was no executive within the business unit that was willing to try something new to solve their issues. I was told on many occasions from management that, “Agile

looks good on paper (in theory), but will never work here.” The Enablement Team viewed the role as an “add-on” to an already overloaded amount of work in progress. In times of crisis, any Enablement Team work went to the bottom of the prioritization stack – solving the fire du jour was status quo, and being a fire fighter was almost worn as a badge of honor. More than one manager proudly said to me, “We fight fires really well here.”

We ultimately ended up with an Enablement Team “in name only.” They did not remove impediments for their Teams; they either layered Agile on top of their existing process or they flat-out ignored the Teams themselves, thinking self-organizing teams would figure out solutions themselves. When organizational issues arose, they essentially imposed what they felt were good management solutions (i.e. implementing more checks and balances and multiple reports by different groups with same information, all to ensure all the jobs were being done) instead of figuring out how to remove obstacles for their teams. Often, they would demonstrate non-Agile like behaviors, such as introducing more work mid-sprint (this is an emergency!), removing Team members without telling the Team to work on other projects, and working towards 100% utilization of individuals and departments versus how to get out work more quickly as an organization.

There was little involvement in the management and executives in sprint reviews for the Teams. This led to lower Team morale (do they even care what I’m doing?) and lack of visibility into vision/direction of the company by the Team (are we even working on the right thing?). Enablement Team demos essentially fell off altogether, since no stories from the transformation backlog were getting to “done.”

We also attempted to incorporate a Product Owner forum in a similar fashion to the finance company. For example, several times, more than one Team unknowingly worked on the same stories without another Team’s knowledge. Clearly, more cross-Team collaboration and coordination at the product backlog level was needed. The first PO forum not successful. POs were not adequately prepared to share stories, there were too many attendees not yet trained in Agile practices that were introducing unnecessary noise during the meeting, and the session was not long enough – just a few hours – for us to walk out of the meeting with a reasonable release roadmap. Feedback was mixed to negative. So, without any opportunity to adjust from feedback, the PO forum was seen as a one-time failure, disbanded and not tried again.

Prioritization and portfolio management continued to be a challenge. It took almost 90 days to create a 90 day prioritized “release plan” (e.g. scope), with planning for the next release cycle taking nearly as long. Scrum team members began to be “poached” for emergency things more regularly, bringing the 80% team participation rate down to 50% or less in some cases. Reporting became even more arduous; typically teams were asked to do anywhere from three to five status reports a week with the same information but different reporting formats depending on the requesting parties. Executives were unwilling and/or unable to change their behaviors. Just think of the possibilities this company as a whole could have had if they actually embraced some of the principles and values.

Finally, in terms of staffing, we were asked to provide SMs and POs since they didn’t have any internal resources available (since they were busy fire-fighting...) which, in retrospect, was not a good move on several levels. First, from a coaching standpoint, coaches should be at an organization to help *them* achieve their goals, not to do the job for them. By us providing SMs and POs, it didn’t help change any behaviors, per say, of Team members or the Enablement

Team. Secondly, it was perceived by the insurance company that we as consultants were “taking over” versus trying to help them achieve their organizational goals. Combining these two things, there was a strain on the relationship such that any trust that had been built up was compromised and never quite rebuilt; we were perceived as external consultants trying to make money, not as a true partner to help them succeed.

Despite the difficulties, Teams were actually making some strides with their end customers. They were being transparent with delivery (working, tested code as well as sharing target stories for upcoming sprints), responding as best as they could to feedback from customers. There were certainly bright spots from selected individual Team members as well. For example, some wanted to learn more about continuous integration and automated testing – so, they took it upon themselves to research and implement it. Several POs collaborated across teams and swapped stories that better suited their Teams’ capabilities. But these actions were based on an individual’s predilection to working collaboratively and/or desire for more technical knowledge.

All in all, there was never any real stability from which to add more Teams. There was interest from several other parts of the organization to try either Scrum or Kanban. However, we could not align management or executives to enable this to happen. With so many challenges that could not be overcome, it didn’t make sense to scale. Eventually a new management and executive structure was put into place, and they did not want to pursue an Agile approach. So, the engagement and any transformation efforts subsequently died rather quickly.

6. COMPARISON OF OUTCOMES

For these two particular companies, Table 4 at the top of the next page summarizes some of the outcomes.

As seen from the outcomes, by and large, I have found that with proper setup and reasonable coaching (whether Scrum, Kanban or hybrid), Teams follow a typical and somewhat repeatable ramp-up cycle. Due to a variety of factors such as working collaboratively, focusing on limited items, bringing testing into development process, and so forth, Teams tend to perform better than their respective status quo.

However, in my opinion, based upon observations of these two clients (and other clients as well), is that the people who call us in, the executives and managers, are the ones that tend to have a more difficult time in an organizational transformation. They understand the theory and why change should happen, but they struggle to execute. When the executives, managers *and* Teams are in greater alignment, I see better results, as is the case with these two companies.

TABLE 4: COMPARISON OF OUTCOMES

CHARACTERISTIC	FINANCE COMPANY	INSURANCE COMPANY
Scrum Team Performance	<p>Generally good to excellent –</p> <ul style="list-style-type: none"> • Dramatically higher quality software (90% fewer bugs) • Delivered double the scope (2X) than expected • Collaboration within Team greatly increased; teams functioning as teams 	<p>Generally fair to good-</p> <ul style="list-style-type: none"> • Moderately better quality software than with waterfall • Delivered on par to what was expected • Collaboration within Team based upon individual Team member’s personality and preference for doing work; some teams worked well together, some teams were just a collection of people
Management Capabilities	<p>Transitioned from managers to “Servant Leaders” – doing things to help remove blockers from the Scrum teams</p>	<p>Stayed managers – could not transition to Servant Leader mindset. Continued to “manage” and put more controls in place</p>
Executive Team Activities	<ul style="list-style-type: none"> • Had initial high engagement, then drifted. Involvement returned during re-launch of Enablement Team • Adopted mindset of Minimum Marketable Features (MMF) for customers • Excellent engagement with their Customers, provided “servant leadership” to the organization 	<ul style="list-style-type: none"> • Little change in behavior observed • Was aware of concept of MMF, but deferred to big bang rollouts (all or nothing) versus incremental builds of product • Saw little evidence that Customers were re-engaged
Customer Satisfaction	<ul style="list-style-type: none"> • Generally quite higher at the Scrum team level via continuous delivery of working software and making adjustments due to feedback • Moderate improvement at the Customer’s Executive levels (still some uneasiness, but happier than before) 	<ul style="list-style-type: none"> • Sporadic highlights from specific Scrum teams (points in time for certain features highly desired by customers) • Less overall satisfaction in the end from their customers, especially from the Executive levels
Did Scaling happen?	<ul style="list-style-type: none"> • Yes – at least eight Agile teams, with more planned upon leaving 	<ul style="list-style-type: none"> • No – stalled at four Agile teams

7. BASELINE CONSIDERATIONS FOR TRANSFORMING AND SCALING

When it comes to a transformation, organizations must remember that transformation in and of itself is not a goal –achieving your organization’s objectives *better* is the goal. Agile is not a process but a mindset that an organization must be in alignment with in order to achieve greater results. To begin a transformation, certain factors should be in place. To scale across an organization, it is essential to have a solid foundation to build upon.

While not an exhaustive list by any means, the following are several questions to ponder before engaging in an Agile transformation initiative, and more importantly, to address before significant scaling.

1. Does your company have a clear (enough) vision so that Teams can create products to map to that vision?

2. Do your Executives:
 - Believe there is a problem with the status quo?
 - Buy into the concept they will likely need to alter *their* behavior in order for the organization to change?
 - Understand they will likely need to reestablish relationships with their top customers, help their customers come along with the transformation journey as well?
 - Have the fortitude to prioritize on a limited set of key strategic initiatives and let others go?

3. Do your Managers:
 - Believe there is a problem with the status quo?
 - Have the willingness to roll up their sleeves and do something about it, including change *their* behavior?
 - Have the capabilities/business acumen needed to do an organizational change?
 - Accept that their staff may be asked to do things not in their job descriptions (e.g. Developers testing, Testers helping with story mapping, etc.)?

4. Is your organization willing to use Agile to roll out Agile, including:
 - Working from a prioritized backlog of transformation stories?
 - Limiting work in process (WIP) of transformation stories?
 - Adjusting transformation stories (pivoting) along the way (i.e. add stories, remove stories, reprioritize, etc.) based upon feedback and empirical evidence?
 - Incrementally implementing changes in small vertical slices of the organization, learning, then adjusting versus following a pre-planned (or big-bang) transformation program?

5. When it comes to setting up Agile teams:
 - Do your Teams consist of five to nine people, each available at least 80% of time?
 - Do Teams have all roles needed (front end developers, back end developers, user experience designers, quality, etc.) to produce working software?
 - Are the majority of the members of your Team internal, full-time resources (especially PO and SM roles)?
 - Can members be co-located with reasonable facilities to enable collaboration and productivity?

6. When it comes to the product selected for the Agile teams to work on/produce:
 - Is it large enough that it demands visibility but not too large that it would shut a company down?
 - Does it satisfy a business/customer need that they are willing to purchase from you?
 - Can it be released in increments?

7. When it comes to the act of releasing product incrementally:
 - Is your organization willing to do incremental releases?
 - Are your customers willing to accept incremental releases?
 - Can you identify a set of early adopters willing to provide candid and valuable feedback on incremental releases?

8. From a technical/physical environment standpoint, does your organization...
 - Have technology and processes in place to quickly and incrementally release product with little overhead?
 - Have enough technical environments to support Agile development?
 - Use “permission based” coding to enable simultaneous releases of software to different sub-sets of clients?

Again, this is by far not an all-inclusive list. There are many more questions out there. But, this can be a place to start for organizations to consider (and adjust) before starting out on their journey. If there are more “Nos” than “Yeses” above, then there may be some foundational work to do first before initiating an organizational transformation effort.

If you find yourself ready to scale (or are already scaling) beyond pilot teams and likewise have more “Nos” than “Yeses”, in this case, then you will likely need to temporarily pause further expansion efforts. Create a force-ranked list of issues to address before continuing, and work through them until reasonably resolved. Positive responses to these questions above have helped me establish that foundation and have enabled greater success in overall scaling efforts.

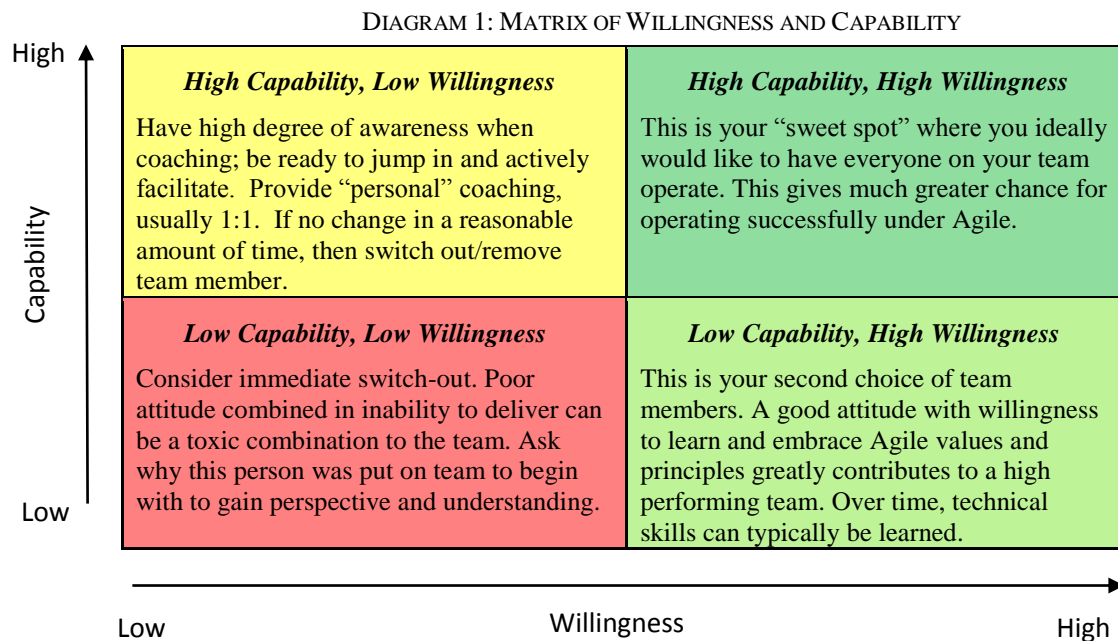
8. NEXT STEPS

In my subsequent transformation engagements, I have continued to use this approach (three-tier model, coaching Teams, setting up Enablement teams, Team selection (see appendix A), User Advisory Groups (see appendix B), etc.) still with mixed results. I keep iterating the questions presented in Section 7, adding/removing based upon my clients’ goals and needs. As with the above two companies, the Agile teams eventually do perform. Scaling is where it gets tricky. Time will tell when there will be a more clear, repetitious path for the managers and executives as well. Until then, I will keep theorizing with the best/known information at the time, trying something and adapting.

APPENDIX A: SELECTING TEAM MEMBERS

As most already know, when it comes to executing a transformation, the people involved are the key to success. It is essential to have the right mix of folks involved, especially during scaling, having both “technical” skills as well as “people” skills, along with the capacity to adjust and learn along the way. Whether at the Team level (e.g. Scrum teams that produce potentially shippable product) or with an Enablement Team, there are certain characteristics that I look for in participants.

The following Team member matrix contains two dimensions, “Willingness” along the X-axis and “Capability” along the Y-axis. Willingness refers to a person’s ability to be open for change, willing to try something, see if it works, and then adjust based on empirical data. The “Capability” here refers to a person’s “technical” skills (or that set of domain-specific skills required for someone to perform their job). One could also add a third dimension related to “people” skills, such as collaboration and communication. For simplicity, let’s assume that the collaboration and communications skills are embedded within “Willingness”.



I prefer to have any team made up of folks in the upper right quadrant: High Capability / High Willingness. This is not only someone who can technically perform their job well, but “plays well” with others and is willing to try something new. Failure is OK for these people, as long as valuable lessons can be applied. However, having an entire team of these members is pure fantasy. Organizations needing coaching typically do not have many folks that fall into this category. So, I find it helpful to have a handful of these folks sprinkled around each of the teams.

My next choice is High Willingness, but Low Capability. Over time, technical skills can be learned. But, it can be difficult to “teach” someone collaboration, communication, and willingness to try. It takes great effort to raise Willingness skills. It is better for teams to have members that are willing to at least try something new and learn from failure. I have observed many times over that positive energy carries a team further than pure technical skills alone.

Be cautious of folks that fall into the High Capability, Low Willingness category. This is typically someone that's been around for a while, is seen as expert – and this is not inherently bad. As a coach, this person needs to be observed and will likely need coaching (even if they don't think they need it), especially if their attitude and actions tend to have detrimental impact on other team members.

In the last group, Low Capability, Low Willingness, you need to evaluate this person and their role in the company, to see if there is a more appropriate fit somewhere else in the organization. It doesn't necessarily mean let this person go – you need to see where this person could potentially contribute better. While this is a rare occurrence, I had a few hard conversations with both individuals (on Scrum teams) and managers (on Enablement Teams) to reassign. Don't be afraid to move people around if it's just not working.

WHEN CHARACTERISTICS APPEAR

There are two places that I tend to use the matrix: for Agile teams (that deliver product) and for Enablement Teams (which consist of primarily managers). Members of each group begin to display which quadrant they lay in at different times. I find that for Agile teams, members typically show their primary characteristics rather quickly – as soon as a Sprint 0, definitely within Sprint 1 when we are trying to get to some sort of working software/product. It's better to do any reshuffling earlier on, especially when trying to guide teams through the “storm” phase as quickly as possible.

For Enablement Teams, it tends to take longer for their characteristics to appear. When launching pilot teams/programs, they are really focused on tactical things. But, when it comes to scaling, conflict is more prominent. Management must now address somewhat serious and perhaps “legacy” organizational impediments, which can conflict with their own motivations (i.e. financial, protecting their silo, metrics that make them look good, etc.). This is when their placement in the Willingness-Capabilities matrix becomes much clearer.

When push comes to shove, my experience has been the majority of managers tend to fall on the lower end of the Willingness axis. They typically begin to revert to what they perceive to be tried and true approaches, even if the data shows the status quo is failing. The managers also tend to become protective of their groups/people, which is not necessarily unreasonable. It's when they reinforce or build silos where it becomes an impediment. If a manager is unwilling or unable to change, it's OK to transition those managers involved in the transformation to more of a “keep them informed” role. It's not uncommon to reshuffle the Enablement Team several months into a transformation effort, typically once the scaling efforts have really started.

APPENDIX B: THE USER ADVISORY GROUP AS “EXTENDED” TEAM MEMBERS

When undergoing a transformation, it’s not just the company itself that needs to change, but their customers as well. For example, the intent with Scrum is to get rapid feedback from ideally the end users of the product. Incorporating the feedback and adjusting is key to ensuring that the highest value is created and delivered. Teams need feedback; customers need to learn how to give feedback.

So, in both companies, the concept of User Advisory Groups (UAG) was introduced. The intent was two-fold: (1) to gain feedback on whether or not the *right* product has been built, and (2) gain input into the *future* direction of the product. While the concept of getting feedback early and often via the UAG is directly related to the *product* itself, it also helps indirectly with an overall organization’s and customer’s transformation efforts as well.

At each company, for each product, we created six to eight member UAGs. There were three UAGs at the finance company and two UAGs at the insurance company. This number of UAGs seemed to work from a scaling perspective, since there was little overlap in products and customer sets between the UAGs.

Here’s the tricky thing: the UAGs contained participants from multiple customers, which was quite controversial. (Why would you put multiple customers together in a room together at the same time? Inconceivable!). The UAG members would attend at minimum the sprint review every two weeks. In addition, on the off-weeks from the sprint review, the UAG members also had an option for a mid-sprint check in for two purposes (1) to answer “last minute” questions about work in the current sprint while the product was still being built, and (2) to give visibility into a “roadmap” of what the next three to five sprints are likely to include (subject to change based on feedback...).

In most cases, just like teams, UAGs need to go through the form-storm-norm-perform cycle. The first several reviews and meetings were guarded and a bit tense. But, after three or four sprint reviews, members in the UAG became more familiar with the format and comfortable enough to begin to share opinions and gave feedback. When the Team reacted to their feedback and adjusted, trust began to be built, and comments started to flow more easily. Also, members in the UAG from different companies actually started sharing their practices with each other, giving each other tips on how they solved certain problems. There was now collaboration between each of the customer’s companies – which was fantastic to see.

One of my favorite parts of coaching teams is when they actually start to joke with their customers. For example, in the insurance company where we were working with healthcare information, we had mock data for fake patients such as Anita Doctor and Imma Patient. Once the Team members became comfortable, began to listen and understand their customers’ needs and how it impacted their job, the light bulb went off, and the teams started to really design and implement software to solve real business problems, instead of guessing solutions that they thought were reasonable. Overall, this improved relations with some especially cranky customers; they actually became allies, and supported/defended Scrum teams even to their senior management.