Anatomy and Physiology of an Agile Transition

Christopher P. O’Connor
University of Michigan
Institute for Social Research
Ann Arbor, MI 48109
Email: cpoconnor@umich.edu

Abstract—A macro view of an enterprise transition from the point of view of a coach in the trenches. What brought about the transition that this group undertook, previously described in “Letters from the Edge of an Agile Transition?” What was the ecology the team metamorphosed into? Follow me as I uncover the answers to these questions and more. Read how a high level sponsor set out to alleviate the pain of an organization. Hear the answers to these questions and more. Read how a high level sponsorship continues to learn and grow now that the beginning of the transition is over.

I. FOREWORD

Many of the questions I receive about my original experience report, “Letters from the Edge of an Agile Transition,” are about the environment and the context in which the agile transformation took place.

In the original piece I wrote about the road I traveled as the coach of one team in the midst of the transition. I talk about both the good and the bad things that happened as we endeavor to transform a team from traditional waterfall development to a version of agile. On the good side I found that the team was looking for answers to their pain and testing provided a way to alleviate that pain and build some trust. On the bad side we struggled consistently with the pressure of ongoing releases and not falling back into bad habits.

I want to extend that original report to address some of the common curiosities as well as shed some light on what has happened since the original experience report. I will also delve into what brought about the transition and some of the driving forces behind it.

II. THE FOUNDATION OF THE TRANSFORMATION

The market was moving faster than the development group was able to deliver. How many times have you heard this or a similar phrase? Situations like this strike fear into many a company’s leadership whether or not they deal in software, but doubly so in a company that depends on software to deliver their products.

This situation set the stage for Gale’s transformation. The transformation itself started with the company’s choice to bring on a new CTO, one who was known for turning around troubled organizations. After his assessment of the difficulties the organization was facing getting products to market he recommended they move to an agile approach to product development. He went on to facilitate one of the few, if not the only, whole organization transformations leveraging agile principles in the region to date.

During the initial assessment period historical data showed time to market for a product was averaging around 18 months. Some of the underlying issues uncovered were: amount of time spent in meetings, project managers who were focused on a particular project and not on the number of products actually being produced for the customers, communication difficulties across and within projects, and contracts with offshore vendors that inadvertently encouraged bad behaviors in the offshore teams.

Upon attending the weekly status meetings the CTO noticed the content was not significantly changing from week to week. The meeting attendees often knew this would be the case. One cause was the developers working on multiple project simultaneously. Since the project managers were not forced to communicate and prioritize across these projects it was not clear what, if any, progress to expect on a given project. It was up to the developer to prioritize to the best of his/her ability. The overabundance of low value meetings and lack of communication were only part of the equation.

Another piece of the equation was the contracts, an example of which stated 80% of critical and high categories of bugs would be fixed. Either intentionally or unintentionally this encouraged the easiest 80% to be fixed leaving the harder bugs to the company’s developers or the next phase. Another negative effect of this was that many visual and usability issues fell into the lower categories and would not be addressed at all. Not exactly a recipe for quality.

The CTO had success in the past addressing similar ailments by applying agile principles at places such as Arthur Anderson and other companies through AlixPartners. But it wasn’t as easy as just declaring the development group would be agile. For starters, the group had previously tried to apply some elements of agile. This effort by the development group had not been successful, so there was a reluctance when it was proposed as the solution to their current problems.

III. THE FIELD TRIP

To address their concerns, the CTO brought the business and development people to see a high functioning agile team in action at Menlo Innovations. Menlo runs a small software development company that uses its own unique flavor of agile. Here they received a tour of Menlo’s software factory and the content was not significantly changing from week to week.
clients, and this project was being developed at Menlo when the Gale development group came to visit.

My involvement in the transformation began when the group from Gale saw the team I was working with and exclaimed “Oh, this is what you meant?” It was not hard to imagine that, like many groups trying out agile, they struggled to find the benefits while simultaneously trying to figure out agile. Thus their previous agile foray had not achieved the level of productivity and collaboration the team I was working with had.

I answered some questions about how we worked at Menlo and helped address some of the first concerns people have upon seeing an agile team. From: “How do you work in such a noisy environment?” to: “Isn’t it hard to work with someone all the time?” and: “Wouldn’t it be more productive to work individually?” Due to this I like to think I played a small part in helping them decide to give agile another shot.

IV. IF YOU BUILD IT, THEY WILL COME

Overcoming the “we already tried that” objections and other concerns with the field trip was only one of the issues to overcome. Some of the other challenges included the physical environment, budgeting, management structure, and business integration.

As far as the physical space concerns, it was not only about tearing down the walls. It was also about who pays for the walls to be torn down and what will be done with the walls afterward, not to mention overcoming the argument about the cost of the existing infrastructure and cost to refit the space beyond just tearing down the walls. Whether it was done by developers or facilities it would take time and money.

In order to address some of these concerns the sale of the cubes was proposed as a solution. In the short term the cubes were taken down and put into unused storage in the building. A significant amount of “horse trading” occurred in order to work with the facilities people for the actual cube removal and to overlook the short term code violations while the changes were made and brought in line with team, facilities, and code requirements.

What the CTO and the business needed next was a project to be a shining example of agility: a win that could be used to show what would be gained when the entire group was a high functioning team. The CTO brought in an agile consulting firm and together Gale and the vendor set about building a mixed team of a small number of Gale developers and a number of experienced agile developers. Next came project selection. There was a next evolution of an existing product coming up. This gave the team a leg up in terms of requirements. Both Gale and the agile vendor felt this product was a good candidate. It was a small to medium sized product that was well known and the offshore vendor had quoted a timeline of 6mo-9mo.

The project was completed in 3-4mo and was very well received by the customers. The business got more than they expected for around half the time and cost expected from the “old way” of developing the products. This success was great for the transformation but did cause some trouble down the road (more on that later). The success was in large part to what I’ve heard described as “the right people doing the right things.” An “all star” team of highly skilled agile developers, many with past experience working with each other, working toward a well defined goal.

V. RIDING THE WAVE

Leveraging the new enthusiasm for agile, the time had come to build additional teams. In order to better understand the challenges involved in this step of the transformation some context is required. In the years leading up to the transformation, the Gale development group had been reduced to a small number of Gale developers and some onshore-offshore consultants. The mainstay of the development group was offshore teams managed and coordinated by the remaining Gale development staff and the onshore-offshore consultant representatives from the offshore consultant groups. Since the transition was required to be budget neutral, bringing on new developers required reducing the number of offshore developers.

This meant some tough decisions had to be made. There was a ratio of how many onshore developers could be hired as the contracts of the onshore and offshore development team members expired or were ended. The majority of the cuts were contract expirations, and a few contracts were ended early.

As with any change that involves people, this was a difficult time. The development team had worked with many of the offshore people for some time. What helped with this situation was that a number of the onshore-offshore contractors were retained and integrated with the teams. The workload called for 5-6 teams of anywhere from 6-12 developers plus supporting roles. Thus retention not only aided Gale in building teams faster but helped maintain the team’s knowledge base. It also gave the development group flexibility in staffing. Much of the ongoing flexing is handled via swapping members between teams. People roll onto new projects and off as projects finish.

The new teams were assembled from a mix of the remaining Gale people and the onshore-offshore developers. Most of people in these groups had little or no agile experience. To help seed the teams, a limited number of agile consultants and coaches, like myself, were brought on to help facilitate building and mentoring the teams. Some of these were contractors and some were newly hired agilists.

These teams by their nature were not initially going to be functioning at the same level as the all star team of experienced agilists. This is where the trouble I spoke of earlier comes in. The previous success of the all star team made for some unrealistic expectations. This lead to some frustration as the coaches and tech leads tried to reign in some of the wild expectations. Their teams were being compared to the all star team as opposed to one in the process of learning and growing.

As the group continues to grow development staff are often in high demand, but now the group is able to deal with the one of the initial problems. Previously project managers competed for resources, each one thinking they had the developers’ time
while the overloaded developers struggled to stay afloat. Each team’s productivity suffered because none of them knew when something would be interrupted by a fire elsewhere. Now the development group as a whole works to address resource issues amongst the teams. No longer does every project believe it has a given developer, analyst, or QA person working for them when they are really working on another project. While it is true that some roles and people with specific talents still work across teams, it is kept to a minimum and is visible across the teams.

VI. SHAPING THE FUTURE

Sadly not all of the people and roles were able to adapt to the changing environment. The most notable was the project management group. The group had been very traditionally oriented. They focused on meetings and big up front project plans. They managed to the plan, often competing with each other for resources. When much of the responsibility shifted to the product owner and team to formulate the plan, the project managers struggled to adjust.

Six months into the transformation it was becoming apparent that the project management was continuing to have trouble adding value to the team. A tough decision had to be made and the CTO met with the rest of management and they decided to phase out the four project manager positions and create a new role on the teams called delivery leads. The new positions were open to any internal or external candidates but in the end none of the project managers applied for the position. They found other positions either externally or internally. Representatives from the team ended up being responsible for the interviewing and selection for the delivery leads.

This position would focus on facilitating the teams’ goals by: seeking to create the right incentives to help increase productivity, tracking the teams’ metrics to gain visibility of where the teams needed help, and identifying items or behaviors that were dragging on the teams’ productivity.

This role fell somewhere in between the role people commonly identify as a project manager and the coaches. The coaches were more focused on introducing memes, growing the teams’ knowledge of process, skills, and tools; and they often paired in to help lead by example. This in turn supported the teams’ ability to produce value.

One group more than doubled their productivity after the move to the delivery lead from a traditional project manager. Of course there were many factors involved, but other teams also saw increases as they switched from being managed to having a person focused on increased productivity. These individuals better understood agile strengths and goals.

VII. BUSINESS, MEET YOUR NEW TEAM.

Another challenge was teaching the rest of the business how to interact, participate, and collaborate. This is not limited to the product stakeholders and management but also HR, facilities, IT, and marketing.

I think marketing comes as a surprise to some people. Marketing has interesting and important requirements. There are lead times for printing material such as fliers, brochures, and conference booth banners. Press releases need to have design down to best show off the product’s content. When suddenly the site that is the product in many of these cases could change literally on a week to week basis, you can see some of the issues that had to be addressed.

In the agile world people often think “we can figure out how to interface with the rest of the business once we’ve gotten the team up and running” and while there are certainly teams that learn to interface with a non-agile business it is a risk and a barrier to change. In our case we were lucky that with the support of the CTO we were able to influence many elements of the business to be more agile themselves, which allowed them to work more effectively with the new agile development group.

One element the Gale transformation still struggles with is consistency. Specifically, when and where is consistency truly needed and when is it better to let the teams find what works for them in context?

From the management point of view there was risk in letting the teams drift too far apart process-wise. It was seen as increasing the cost of moving people between teams and making it hard for the business to interact with the disparate teams. It was feared the new hires who had come from various different agile processes would be trying to switch the teams over to what they done previously done. Techniques that had proven successful in their old teams might not work in a new context. The fear was that in the end these differences would not allow the teams to work as a cohesive whole.

Surprisingly it was not the tools or languages that would present difficulties; instead it was items like schedules. Most of the developers could in short order pick up another team’s culture or learn a specific tool a team had adopted. A difference in schedule was potentially more difficult. One team embraced a schedule close to a four day, ten hour per day work week, so often a good part of the team was out some or all of Friday. This was fine for the team but if they tried to add a team member who had a strict schedule due to child care or the other teams needed support on Friday difficulties could occur. Overall this did not end up being insurmountable but is an example of some of the complexities that crop up when teams diverge.

From the teams’ point of view this was stifling at best and derailing at worst. If nothing else it was one more hurdle when there were already many in front of them. There is definitely a point at which you can either give the team the freedom they need or they will find a way around “the rules” in order to get their work done. No one wanted to become the process police. It fell upon the shoulders of the head of development to be the enforcer along with some support from the team leads, but for the most part only some guiding was required so the boundaries remained untested.

In the end most of the teams try to be consistent in the metrics they provide, such as velocity, code coverage, cyclomatic complexity, burn down charts, and the macro level estimates that the business requires. Outside the team room
they shoot for consistency but inside the team room they strive to find what works best with some basic guidelines, such as don’t switch languages or decide to switch hardware platforms without consulting the group at large.

The group continues to try to balance these often conflicting goals. Absolute consistency across all the development teams is not desirable and would be too costly to attain and maintain, but the current level is manageable and they are able to maintain the upper level support and visibility that they need.

One last piece of the puzzle in terms of the development group is Gale’s focus on continuous improvement, encouraging their staff not only to increase their own skills but those of their colleagues. To this end at a macro level they support their staff’s participation in conferences, such as Agile and Beyond, Software Craftsmanship North America, and the Agile Alliance. Internally they support lunch and learns on tools being used, code kata/randooori, and craft days where an entire day is devoted to improving the staff’s skills.

VIII. WHERE ARE THEY NOW? GALE EDITION

Now that many of the initial challenges have been overcome, the teams have moved on to longer term goals such as continuous improvement, assisting management with forecasting and consistent metrics. These goals can be every bit as challenging as the ones faced while becoming an agile organization.

One constant battle is the tension between self-organizing teams who see opportunities to optimize their team by continuously improving their process. While achieving this goal, teams have ended up with different processes. This makes it difficult for the upper management to forecast costs and returns when the estimates and metrics they receive from different teams are difficult to compare with one another. On the other hand, not allowing the teams to find their own way to achieve their goals led to frustration and reduced productivity.

Another tension is trying to produce reasonable initial estimates. The estimates assist the business side with macro-level prioritization of which products to invest in. One of the elements we worked hard to develop in the teams is learning as we build instead of trying to gather all of the requirements and do big design up front.

This is often problematic because the time when the estimates are required is also when the teams know the least about the product. They have only begun to learn about the solution. Of course the teams do their best to leverage similarities with other products, especially as many of them have similar elements given the publishing domain, but this does not completely resolve the issue and it continues to be an area to improve.

The transformation has gone a long way toward developing and increasing the trust between the business and technology, thus helping the business feel more confident in when products will be delivered. For example, one team previously went through three rounds of high/macro-level estimation and refinement before being given the green light. On the next project the team was able to learn from the previous project and what the business was looking for in terms of packaging and presentation of the high level estimates. The result was the next project required only one round of estimation. Another result was that the project was broken up into more phases. The first project had been done in two large phases. In the first phase the majority of the features were completed and the second was the more or less for overflow.

In the second project, the project was broken up into four phases and the business only budgeted out the first two phases. The team also focused on a smaller set of requirements and had a clearer idea of what was to be built. The second two phases were unfunded and were less defined. This allowed the development team to do less guessing while allowing the business to risk less by investing. Then after the first release and into the second they could refine and plan the other two phases and allow them to leverage feedback from the user base to determine whether further investment was warranted.

The team also approached the two phases using different philosophies. The first was done based on time, and whatever features were able to be developed before the deadline would be the first release. The plan for the second release was feature-based and the release date would be more open ended. This gave the development team a way to adjust their sprints and planning accordingly.

IX. THE REVEAL

With the transformation heading into its second year it is amazing to see how far this group has come. They have cut the average product delivery time from 18 months to nine months. Communication both among the development group and between the development group and the rest of the business is much better and continues to improve.

The non-development groups have also embraced some agile principles and are not only better set to interact with the development group but also are functioning at a higher level themselves. The development group is able to move people between teams with little or no trouble.

For perhaps the first time the business has true visibility into the development pipeline and is better armed than ever to guide the business accordingly. Gale still has room to grow as a high functioning development group, but they are well on their way and continue to evolve as an example to others.

The most interesting observation for me was the shift in attitudes among the non-agile developers as they learned to appreciate pieces of the agile principles that resonated with them and began to see the value we hoped to bring to the teams with the application of the processes.

I have been asked a number of times if I thought we could have done something less than a full transition and I do not think it would have been successful. Along the way I saw a number of times where people clung to the old way regardless of the fact that it was broken and caused them trouble.

While there is no guarantee that they will continue to be successful, one of the best tools they now have is an organization that now knows how to learn: learn what works for them and how to improve continuously. Even though this
is not what I would have declared as the goal at the outset, it is one of the most valuable outcomes. The group has come a long way with their process and they have assembled a very good group of people. Where they go from here only time will tell.

ACKNOWLEDGMENT

The author would like to thank Gary Baker, Brian Foote, Karen Morand, and Patrick Wilson-Welsh for their help in bringing this paper together.

REFERENCES
