Enabling Agile in a Large Organization
Our Journey Down the Yellow Brick Road

Thomas R. Seffernick
KeyCorp
Key Technology Services
2025 Ontario St., Mailcode OH-01-00-0801, Cleveland, OH 44115 USA

Abstract — KeyCorp, a financial institution with $90 billion in assets, has been working hard to transition from a waterfall approach to an Agile approach for software development. This paper shares the story of KeyCorp's journey down the Yellow Brick Road to the Land of Agile.

INTRODUCTION

KeyCorp, a financial institution with $90 billion in assets, has a development organization with 1,500 associates across 2 major domestic development centers and 3 offshore centers supporting 800+ applications. Twenty-four months ago, the development organization was deeply embedded with the waterfall development practices and mindset. Today agile is on the minds of a majority of our associates, and more importantly, our business partners consistently recognize increases in value for their technology development spend.

It is hard to believe that this transformation has progressed without a single methodology or tool rollout initiative. One of the common principles we applied from the very beginning was, "you can't make people develop in an agile way." What you can do is set a clear vision, show people different ways to achieve the vision, remove every reason for not trying, enable early adopters to be successful and praise every possible success no matter how small. This seems simple in theory but very difficult in practice. Here is our story.

PICTURE THIS

As you are sitting at your desk one day, you close your eyes and begin day dreaming of a better place, a place where you are justifiably proud of the software you are developing, your boss is not telling you every little thing you need to do, you are dedicated to a single project, you are rewarded for successes rather than punished for failures, your skills are more valued than following a process and your project sponsor is not asking for an end date and cost before you begin developing.

You wake up in day dream to find yourself in the Land of Agile. To your surprise you find many of your friends there, desiring the same happy place as you. They inform you that no compliance function was watching you and you can go back to the place you were if you click your heels 3 times, and 2 or 3 demonstrate the process to prove they know what they are talking about. They talk about urban legends, others who stopped in the Land of Agile and left to follow some other path. Then they quickly follow up with the advice to stay where you are, the agile path may be too risky, no one has ever come back and there are rumors that some were gobbled up by wicked witches and flying monkeys. You decide not to listen to them and to give it a try, remembering you can always click your heels and return to the place you once were.

That is exactly where developers at KeyCorp found themselves. Before we start off, we should look back to describe what the development organization was like. Twenty-four months ago, development occurred under a command-and-control culture, and failures were punished while successes were ignored. We had a strong centralized project management office (PMO) operating in a compliance-oriented environment. Our resources were highly matrixed, often working on 4 to 5 projects at the same time. Specialized skills caused a "ball over the wall" mentality, and finally, we had a mandate to "buy" over building functionality. Methodologies were mandated and rolled out. Teams used them as a façade only. They were still developing the same old way but slower because of the additional process.

Our business partners were also not happy and did not trust development. They often looked and decided on package solutions without engaging the technology group. They told development teams, "the only thing you care about is your process," and "you make 92% of your commitments, but you are not helping the business, because you are failing on the big high value projects and too slow and too expensive across the board.” Wow, now you know why our developers and business partners often dreamed of a better place.
OUR EXPERIENCE

Set the vision for a new culture.

Our journey begins 24 months ago with the introduction of Steve Yates as KeyCorp’s CIO. He focused his attention toward breaking down the barriers between technology and the business of banking. His message was clear: projects returned more value with incremental delivery and engaged business partners. The best way to improve a project was to increase both.

The first visible evidence that our business partners were on board was a transition from a desire to purchase and integrate a package to a desire to build just enough of a solution to begin to get value and improve it over time. Organizations measure development productivity in different ways. Our conversations with our business partners told us that the only measure they cared about was speed to value. Given that value is often judgmental, the best way to assess improvement was to ask. They told us by real actions, such as trusting us to develop viable solutions to their problems. This is the stage where our developers began dreaming of a better place.

That mindset was taken to the next level 6 months later, with a change in Technology Development leadership. The very first day that Alan Buffington became our Director of Development, you knew things were different. It all started with his vision of what was important: “Our people are justifiably proud of, and recognized for, their contribution to the business.” From the very first Town Hall introduction 18 months ago to an offsite discussion with a group of developers last week, this vision has been consistent and woven into the culture of the Development Group. You can imagine the surprise at the response to Alan’s first question at his first Town Hall. A brave developer asked, what do you think of RUP? His response: “Well, I’ve developed waterfall methodologies in a past life and they just don’t work. We fool ourselves in thinking we are smart enough to put together a cookbook of steps to follow to create good quality software. I would like us to consider a new way of development, one that enables the continuous flow of business value. I have experience using Scrum as an enabler.” We don’t have any statistics, but we believe the Scrum Alliance website may have set a record for hits that day.

Empower the organization.

To make this journey a success, it was clear that the transformation had to start with our strong PMO. Here we had the hub of all things project-related – coordination, reporting, process quality assurance, compliance, code performance quality assurance and the 100+ project managers themselves – in a central place. Nothing happened without the PMO’s intimate involvement. It was a bottleneck that just could not exist for the organization to be agile. First and foremost, all the project managers were returned to their positions in the development organization, reporting directly to development managers. There the project managers (PMs) became accountable for project success – no longer were they simply facilitators and enforcers of the process. While they had always been accountable for truth in reporting, now they were accountable for results – their bonus was on the line.

But why didn’t the PMs just become strict command and controllers? Because coupled with their move, we educated everyone, developers and managers, on the inability and inaccuracy of prediction in software development. This wasn’t too tough, as we had learned the hard way (by multiple failures) that we need to be flexible, constantly seek and use feedback and establish clear ownership of our software and its projects. We introduced project managers and teams to Scrum.

Now that the PMs were accountable for success (i.e., time to value), they started to complain about all the unnecessary impediments in their way of it – our process. So we began tearing it down, little by little. We started with the “musts/shoulds/coulds” in our project management methodology. Instead of encyclopedic-sized requirements, we streamlined our policy to a page and added judgment to the PM job description. We gave the project portfolios, including their reporting, back to those who were accountable for reaping value from them. We found the compliance officers something more fun to do – help others.

With most of our people and process, so went the PMO’s identity. Now what?

Remaining was a skeleton crew of project manager/business analyst mentors, process-focused managers and analysts who felt a tremendous loss – of their friends, of their community, of their roles. We came together to build our future and begin the long process of re-shaping our identity as the Software Development Support Center (Dev Support). To do so, we had to widen the picture to see the core purpose of Enterprise Technology Development (ETD), find its holes and figure out how to fill them. We found that ETD, on its journey to becoming agile, had to become great at development. Dev Support discovered its mission: help ETD build better software, faster.

So began a new identity – Dev Support made the change from a team that rolls out and enforces process to one that provides thought leadership, introduces new practices and tools and then coaches individuals and project teams on their real-life application. We became known as the Center of Excellence for new stuff and grew to incorporate technical excellence in application development, testing, middleware and
business intelligence. The organization-wide change agent was in place.

**Enable a relentless passion for removing impediments.**

To help enable this change – remember these concepts were all foreign – we needed to engage an expert to help. We were fortunate enough to have Ken Schwaber as our organizational coach. When he came in to introduce Scrum to our technology leaders, you guessed it; we fell into the “Yes, but …” trap. We stood up and described how Scrum sounds wonderful but it will not work here. Why? We are highly regulated, our billing processes will not support incremental development, our rewards and recognition practices will not align, a matrix workforce is efficient, there is no other way to deal with 350 active projects, our performance evaluation process will not allow it and on and on and on.

All of these impediments were captured and presented to the Director of Development. Surely that would end all this Scrum talk, and we could go back to analyzing, designing and testing out defects. Just the opposite happened; he thanked the group for their courage in bringing these items to him, because this gave us a starting point. You can imagine the look on those leaders’ faces. What just happened here? What are we going to tell our peers? The era of Agile Enablement Teams was born. The list of impediments was published to the leadership team asking if anyone wanted to sign up to figure out how to remove them and show their progress at monthly review meetings. As you can imagine, that first Agile Enablement sprint review was painful as leaders stood up and described their plans to remove the impediments they volunteered to address. The message was clear – plans are good, but results count. The dynamic of those reviews changed from that point, and results became the focus. We did not recognize it at the time, but our development leadership began to feel the relentless focus on continuous flow of value.

Over the next 16 months, our developers stopped reporting time, our performance evaluation practices changed, reward and recognition became more common place, employees were coached rather than told and test driven design practices have been proven to work. After 16 months, the backlog of impediments was reduced to the low priority items that were not worth addressing.

**Establish an agile project management framework.**

So how did we get started with Scrum? In order to grow real change, we wanted to create a project success, and then ask the team to stand up and share their success with their developer peers. We started with 2 pilot projects, sending 3 people to ScrumMaster training (2 ScrumMasters and the leader of Dev Support). We gave them everything they needed to be successful, including Ken Schwaber to coach them along for the first several iterations. The first lunch-time panel discussion in which one of the pilot teams and their product owner answered other developers’ questions about their experiences was memorable. Scrum had caught fire and life in the Development Support Center got real busy.

**Add Developer practices.**

After a year of Scrum focus, we reflected on where we were and recognized our developers were ready to change the way they develop. We used the same model for Test Driven Design (TDD) practices as we did for Scrum. We engaged Brian Marick to show our organization what agile looks like from a developer’s perspective and then visit us monthly to coach teams in the practice of Test Driven Design. We then gave a project team an opportunity to show the development leadership how they develop. You can imagine the reaction when the team’s tester stood up and stated that they had developed for 3 sprints without a single defect. TDD practices are beginning to catch on with our development teams.

**Connect product owners from the start.**

After month 16 of our journey, focus shifted toward connecting the product owners, ScrumMasters and technology managers prior to the project start. Our projects were starting without good backlogs and the level of product owner involvement was inconsistent, causing projects to start slowly. After some soul searching we concluded that we were treating agile development as a paradigm shift (mapping existing roles to Scrum roles, yet not doing anything different). What we needed was to educate it as a new paradigm. The result was Agile Connections. This is a sequence of training that starts with an overview to create awareness and desire to use agile practices in delivering the business value. Then, hands-on workshops transfer knowledge, followed by jump starts and coaching to help reinforce the ability to use the practices. The overall goal is to get the product owner, ScrumMaster and technology manager together to enable projects to start better.

To date, this workshop has created raving fans among our product owners. One product owner who participated in Agile Connections commented, “This is the first experience I have had at pulling both technology and the business together for an up-front, more detailed conversation about approach and prioritization of project requirements. I thought it was very beneficial and hope that it set the communication stage for the entire project.” Another told us, “It allowed all of the participants of the project to hear the same message about what agile methodology is and how to best utilize it.” We are most proud of the fact that our technology and business leaders have become
our Agile Connections marketing tool by sharing the positive experiences with their peers.

Never stop looking for opportunities.

With the inception of agile development, it became apparent that Technology Operations was not able to keep pace with rapid deployment and dedicated development teams. Senior leaders from both development and operations came together to openly share their thoughts and provide feedback. As an outcome of those meetings, the Infrastructure Solutions Management team was formed within the Technology Operations group. The goal of the team is to provide infrastructure support for the rapid deployment of applications from inception to production. Teams are aligned to each development group, with the purpose of helping the highest priority projects within the group to ensure that the Technology Operations teams are not being an impediment to development. In addition, this group will partner with other Technology Operations groups to streamline processes and enhance the current service catalog or request for services.

Other groups within Technology Operations are also adapting their organizational structure and processes to support agile development. They will do this by having engineering and operations functions handled by single teams of subject matter experts. This reduces the number of people on the project and the need for "hand-offs" between resources.

Although in their infancy, these changes have led to several successes already. Lead times have been reduced through process improvement, and we've been able to deliver better solutions faster, allowing the development teams to focus on the task of developing high-quality software.

What is next? Mainframe development seems to be an opportunity that consistently arises. We hear our mainframe team tell us, “This agile stuff is good for web applications but does not work for my Cobol development.” or “You keep showing me the open source tools that those Java and .Net developers have for automated unit testing and continuous build, but none of that applies to what I do.” We have a mainframe development team that disagrees and is in the process of developing their own Cobol unit testing tool. This is an example of how innovation is alive within the development community. It is currently too early to predict success, but we are optimistic that this will provide a way for Cobol developers to adopt Test Driven Design practices.

Measurement

Measuring the effects of agile software development is a sensitive topic. We come from an environment that had 2 or 3 measurements for the same thing, because we did not trust people to do the right thing. Knowing that you get what you measure, we have been very cautious with measuring agile development practices.

Instead, we have looked at results and listened to feedback from our business partners. The feedback has been clear; our business partner satisfaction has greatly improved as a result of delivering business value faster. The resulting impact to projects has been clear – our product owner involvement in the day-to-day project decisions/priorities is at an all time high and internal development is now consistently winning out over package implementations. Those are two examples that lead us to believe that a true partnership has evolved.

There are indirect quantitative measures as well. First, we saw a 4% increase in technology employee satisfaction survey scores between 2004 and 2006. Second, as a financial institution we do a good job investing performing assets. Over the past 6 months KeyCorp has approved capital investments of $500,000 to improve the development team’s physical work environment. These improvements include a reconfiguration of the work space to include team rooms, physically co-locating teams and a game room and coffee shop in a common area. We believe this investment demonstrates that technology development is viewed as a performing corporate asset.

We also know we need to get some basic measures at the project level. We are currently piloting a set of project-level leading indicator metrics. These indicators include portfolio partner satisfaction, anticipated value multiple, working as one team, demonstrating value each iteration, focus on business priority, and inspect and adapt. Upon successful pilot we are anticipating that these metrics will be reported in the KeyCorp Quarterly Balanced Scorecard.

Results In Their Own Words

The best way to understand the results to date is to hear the words of the people doing the work. Here are the results in the words of a product owner, ScrumMaster, developer and external regulator.

Product Owner

Thomas Malengo, product owner for multi-million dollar, multi-year teller hardware and software replacement project, says:

“Agile’s flexibility is the big win – we do the right thing in each sprint with proper focus. Instead of our developers asking us to change our product vision to overcome a design problem or system constraint, there is now opportunity to discuss and change the design to enable us to get the most product value from the development effort. This has resulted in higher confidence in delivery, flexibility in product features and an open discussion about what is best for our shareholders.”
“Agile is a common sense approach for software development. Here are a few other direct benefits we have seen:

“Learning – we have been able to learn how the system operates and adjust requirements accordingly to improve process flow, procedure and controls.

“Testing – we have been testing since the first delivery, which enables us to improve functionality and be better prepared for pilot, as we go forward.

“Feedback – we have been able to demo actual system functionality to field representatives and get input.

“Training – we have begun training modules prior to system completion and could not do this without agile.

“Confident flexibility – we gained consensus on the idea that quality of system delivery and training effectiveness is more important than meeting a date!”

ScrumMaster
Sue Godley, ScrumMaster for Card Management Project – Core CMS Team (Card Sharks), says:

“As a project manager coming from a strong centralized PMO, it was difficult at first to assimilate back into the development organizations. It was clear, however, that the processes and procedures established within the PMO framework were hindering our ability to produce tangible, profitable technology in a way that would meet market demands and business needs. As I spent more time within the development organizations and as agile concepts and workshops were rolled out, I became much more comfortable with the direction we were heading.

“I am no longer a project manager but a ScrumMaster. Although the roles are similar, the differences have spawned a change that is continuing to spread throughout our organization. Our individual contributors are much more empowered. They no longer view my role as a hindrance due to the removal of heavy documentation and process requirements that were in existence within the PMO. Command-and-control project management style no longer fits within our organization – I am there for the team to remove impediments and I rely on them as experts to drive the technology development work along with our business partners.

“Another major change I have witnessed is that our product owners are much more engaged. We work with them daily now as opposed to the pre-agile era, when status meetings were held once a week and there was little if any ‘hands on’ communication between technology and business. They are vocal in their appreciation of the incremental delivery approach as opposed to the whole enchilada at the end of a waterfall project. We see and share increases to our business partner’s bottom line numbers.

“I know we still have a ways to go…it is not an easy transition. It’s still a bit hard to ask the question, ‘How do you...’ and the answer is ‘Whatever fits.’ But it gets easier every day.”

Developer
Adam Hutcheson, developer/team member of a dedicated project team creating online banking solutions for small business owners, says:

“Development within KTS has by far taken more steps into understanding the mentality and role of a software developer in the past 12 months than it did in the previous 5 years. Management is creating organizations that encourage peer reviews of project ideas such as the ThinkTank and Mentorship programs. Innovation is a performance measurement, and developers are consistently reminded that it’s okay to think different and question existing practices. Some of these topics include what tools do we need, how can we communicate better with the client or what are the boundaries between excessive documentation and well-written tests. Developers are given the opportunity to weigh in on those decisions and, wow, have we done so.

“With the help of Brian Marick we have implemented testing frameworks such as JUnit and Fit, which have led to those zero defect sprints. These frameworks, in tandem with continuous builds, are replacing the need for excessive documentation, and our business analysts and testers will be the first to ask, ‘Can I see the test that covers X functionality?’ We have experimented with paired programming and now use it as a vital software design tool. The engagement between 2 developers truly leads to the multiple perspectives that you need in order to design good software.

“Interaction with the product owners and clients has changed drastically. Our product owners have been much more engaged and we have recently moved to weekly sprint reviews that don't take half a day's time. Owners make decisions and move on. As for client knowledge, we have recently created a schedule for the team to make actual on-site client visits, which has not only been eye-opening but we are building great relationships.”

External Regulators
In a recent audit of our development and acquisition practices we shared our project delivery policy and development standards, the fact that we don’t mandate or enforce a set of required documents, and that we expect every project to have their own methodology based upon risk and business vision. They then interviewed 20 ScrumMasters and took a deep dive into the project documentation. The results: zero findings, zero actions and only several observations. One of their observations was that the project documentation that was reviewed was
consistent and matched the needs of the project, which was surprising to them, because no one told teams what they had to do and or checked that it was done.

**CONCLUSION**

We still have a long way to go. Things on the horizon that we are actively pursuing include helping product support teams use Scrum, user story planning and estimating workshops, and enabling peer-to-peer communities.

Knowing what we know today, would we follow the exact same path again? Probably not. Do we consider ourselves the experts? By all means, no. We are always happy to learn better ways. Should you follow our lead? Well, that is for you to decide. Here is what we can confidently say, based upon our business partner feedback: We are justifiably proud of the progress made to date.

In the end we learned that we had the courage, heart and brains all along to make this transition. We just need to continuously feed and positively reinforce all 3 while removing every reason to click our heels 3 times to return to a place we once were.

**ORGANIZATION AGILE ADOPTION TIME TABLE**

<table>
<thead>
<tr>
<th>Month</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov 2004</td>
<td>CIO Steve Yates joins Key – Breaking barriers between technology and business.</td>
</tr>
<tr>
<td>May 2005</td>
<td>Development Director Alan Buffington joins Key – Our people are justifiably proud of, and recognized for, their contribution to the business.</td>
</tr>
<tr>
<td>June 2005</td>
<td>Two pilot Scrum project teams begin</td>
</tr>
<tr>
<td></td>
<td>Ken Schwaber introduces Scrum to development management team.</td>
</tr>
<tr>
<td></td>
<td>Agile Enablement Teams form to remove organizational impediments to agile development.</td>
</tr>
<tr>
<td>July 2005</td>
<td>PMO disbands.</td>
</tr>
<tr>
<td>Aug 2005</td>
<td>Software Development Support Center communicates identity – Help development build better software, faster.</td>
</tr>
<tr>
<td>Sept 2005</td>
<td>Internal ScrumMaster training begins.</td>
</tr>
<tr>
<td>Oct 2005</td>
<td>Initial Scrum pilots communicate results.</td>
</tr>
<tr>
<td>Dec 2005</td>
<td>iRise introduced as prototyping tool.</td>
</tr>
<tr>
<td>Jan 2006</td>
<td>Performance review process for development changed to focus on stretch and results.</td>
</tr>
<tr>
<td>June 2006</td>
<td>Brian Marick introduces TDD and begins coaching first 2 development teams.</td>
</tr>
<tr>
<td>Sept 2006</td>
<td>All Development Managers attended coaching workshop.</td>
</tr>
<tr>
<td>Oct 2006</td>
<td>Pilot team demonstrates TDD success and informs management they have several zero defect sprints.</td>
</tr>
<tr>
<td></td>
<td>Initial Product Owner Workshop conducted (later evolved into Agile Connections).</td>
</tr>
<tr>
<td></td>
<td>Team rooms available for dedicated project teams to move in.</td>
</tr>
<tr>
<td>Nov 2006</td>
<td>Final Agile Enablement sprint review conducted.</td>
</tr>
<tr>
<td></td>
<td>Infrastructure Solutions Management team formed to change the way Technology Operations supports projects.</td>
</tr>
<tr>
<td>Dec 2006</td>
<td>Agile Connections workshop begins.</td>
</tr>
<tr>
<td>Feb 2007</td>
<td>Regulatory audit of development and acquisition practices completed.</td>
</tr>
<tr>
<td>March 2007</td>
<td>Game room/coffee shop grand opening.</td>
</tr>
<tr>
<td>April 2007</td>
<td>Agile mainframe unit test tool development begins.</td>
</tr>
</tbody>
</table>

**Product owners are connected from the start.**