JABIAN INTERVIEW:

HOW TO AVOID THE DARK SIDE OF AGILE

CAMERON FARAH

Cameron Farah is a Vice President of Product Management at Cox Automotive. He has been working with software teams at the enterprise level as well as startups for more than 20 years. Cox Automotive Inc. makes buying, selling, and owning cars easier for everyone. The global company's 34,000-plus team members and family of brands are passionate about helping millions of car shoppers, 40,000 auto dealer clients across five continents, and many others throughout the automotive industry thrive. Cox Automotive is a subsidiary of Cox Enterprises Inc., a privately-owned, Atlanta-based company with revenues exceeding $20 billion.

Many companies have begun to embrace Agile methodologies within some or all of their IT development teams. As this transition gains traction and yields positive results, companies begin to consider rolling out Agile further across their enterprises, hoping to replicate and capture this success on a broader scale. These decisions may be partly based on herd mentality (keeping up with competition or the latest industry trends) or belief in the pure theory of why Agile is such a great methodology. What many enterprises fail to see, is that if shortcuts are taken and the full organization isn't fully invested in effectively transitioning and transforming to Agile, there are several pitfalls they could encounter.

The intent of this interview is to expose, not only the positives about Agile, but also the "Dark Side," to help organizations that are either considering adopting enterprise-wide Agile or are currently struggling to do so.

Jabian: As a product person, how do you get the business side of product thinking and performing in a similar way?

Cameron: I've been at places in the past where product people were a part of the process, but they were loosely coupled. In some organizations, product actually reports to engineering (IT), but for us, it's important to have the tie-back to the business. The product manager being embedded in the engineering team supplies them with their features and breaks them down to user stories, and then works with the teams to execute. But that product manager ties back to a more senior level product leader. The product leader really is driving the strategic roadmap of the product that's being worked on.

At other places, you would see gaps in the deliverable. You could see aspects of what was coming out of the scrum teams that just didn't quite line up to where we wanted to go. And when I say "quite line up," I've seen some pretty big misses. This is very different from what is happening at Cox. The product manager is tied to the product leader who is a part of the product organization. From a leadership standpoint, we're constantly trying to communicate our direction, not just within our organization, but across the company so that other teams can hear and understand what we're doing and why. At least in the Cox Automotive way, I think we've been more successful implementing an Agile methodology than in other companies that I've been in.

Jabian: What are some of the critical lessons you have learned from your experiences with Agile?

Cameron: I can think of one company I worked at where the teams were using Agile, but the other parts of the company weren't. You are doing your planning and sprint reviews, and you have dependencies on other teams to be able to build and deliver your capabilities so that you can be successful. Yet other teams function in different ways and on different time lines which makes it agonizing to build enterprise applications.

Now, at Cox Automotive, we are following the Agile methodology as an enterprise. When we do quarterly planning where we're laying out what we are going to build for the quarter, it can get crazy. Crazy because all the interdependencies start to get passed around to other teams. So, teams such as the ones that I manage start with a focus on what we need to build to deliver on our portfolio investments and the initiatives that we're driving.

That's great. But as the planning starts to get tighter and tighter, or coming closer to an end, you start to hear about integration points or other capabilities that the other teams need you to deliver in order to help them be successful. In these discussions, you have to make trade-offs. That's part of the challenge you face as a business, but enterprise-level Agile helps give us enough structure to manage that complexity.

So in the end, the intent is to bring a lot of visibility to what has to be done, and then there are decisions as to what we're going to do and not do, and then communicate back out not just within our teams, but to all other dependent teams. At the end of the quarter, we have a clear alignment on the work that all teams will focus on.

Jabian: What are some of the biggest challenges you have seen when it comes to adopting Agile?

Cameron: Everybody has their own definition of adopting Agile. In my career, I've seen what I'll call "different levels of adoption." I've seen it where the term Agile was used, and work was done in sprints, but little else was really Agile.

IF YOU'RE NOT EFFECTIVELY COMMUNICATING, PEOPLE AREN'T ALIGNED TO WHY YOU'RE BUILDING WHAT YOU'RE BUILDING; THEN YOU'VE GOT A BIG PROBLEM.

The main challenge is whether you are really committing to execute using Agile as a Methodology. I'll give an example from another point in my career where we did "Agile." I had to take one of my project managers and throw them onto a scrum team. They were participating in ceremonies and stand-ups, but they were not solely committed to the team or solving problems in real time. This led to the team not having a really strong understanding of why they're building what they're building, and sometimes that reflects in the final output in the sprint review. The translation from a business-level epic to the user stories engineers were creating against would not align.

Jabian: How do you think the teams can best leverage their leadership in an Agile environment?

Cameron: I think you have to be committed to communicating what you're doing, and more importantly in an Agile environment, what you're not doing. This is a time investment that senior executives are committing to. They're saying it is important that we build these capabilities and deliver on the value proposition. This is built into what we do, and helps bring that visibility up to the Sr. VP level. At the end of every quarter, we're syncing back up with executives on what we are delivering. It's hard to imagine not giving that visibility back into the business at an executive level, because at some point, the question will arise as to what we are building and why.

Jabian: How do you feel that impacts transparency? Do you feel like there's an impact to what they're willing to share, or do you feel like it's a little bit more of an open environment in an Agile world?

Cameron: I don't know if it's Agile that really shapes that, or if it's the culture of the company that really shapes what gets communicated and what doesn't. One of the core tenets that Sandy Schwartz, the CEO of Cox Automotive, communicates to all of us is to be open, honest, and direct. In some companies, that may be a statement, but it really is culturally ingrained here. I think anybody who's been in any business environment understands you have limited resources on what you can execute and deliver. You have to make trade-offs. That's one of the challenges that you face as a business.

Jabian: Documentation tends to be much lighter when it comes to Agile, and there have often been misconceptions that there is no documentation needed as part of Agile. What are your thoughts on creating documentation as part of Agile delivery, and then, what documentation, if any, is still critical to success?

Cameron: We definitely do a fair amount of documentation at Cox Automotive. It isn't excessive and I don't find that it slows us down. Documentation isn't just about communication, it's really about alignment. We start with a portfolio investment. We are going to take a chunk of money and sit there and say, we need to build certain capabilities. These are the expected outcomes when we complete this investment. The investment then gets reviewed by architecture and engineering to give shape to the work required to complete it. The documentation really kicks off at this point and is reviewed by senior leadership for them to say it is useful and will deliver value to the business.

After this, that document gets pared down to the capabilities needing to be built, and then you start getting into more traditional Agile elements of epics, features, and user stories. On top of that, we also build roadmaps for the product so we have a view of where we're going and can tie back to the portfolio.

From my standpoint, the word "documentation" can get a bad rap, but it's as simple as what I said earlier. It boils down to communication. If you're not effectively communicating, people aren't aligned to why you're building what you're building; then you've got a big problem.

STANDING UP THE TEAM STRUCTURES IS GOING TO BE HARD AND PAINFUL. MOST ENTERPRISES DON'T LIKE CHANGE.

Jabian: What is your perspective on business testing in an Agile world? Does it differ from a Waterfall approach? And if so, how?

Cameron: We're building things that touch all solutions within our business. And if we are not doing a thorough and complete job of understanding, or testing our integration points, and closely working with our partners who are being affected by our code, that's going to have bad repercussions. To avoid this, we work diligently to notify affected solutions and internal customers what is going to be released, and we discuss the potential impacts of those. The misinterpretation of what might be a more traditional Agile approach to testing can be dangerous in an enterprise environment.

Jabian: In your experience, how have you seen Agile best executed? What are some of the keys to successful Agile implementation? And then, what are some of the pitfalls to avoid the quote-unquote dark side of Agile?

Cameron: The best implementation I've seen is at Cox, and what does that mean? It means you're committed—at all levels. It means you have product managers and UX teams who are embedded with engineering teams, continuously evaluating customer and competitive needs while working to build what customers and the marketplace value. You constantly monitor and track the work in process using tools like Rally. You maintain flexibility to shift priorities during the quarter due to changing business priorities.

In an enterprise environment, communication is paramount. How teams that have interdependencies share information and communicate is the basis of being able to build these cross-team capabilities.

These are some of the core tenets for really complex, hard, enterprise Agile products that I've been delivering here at Cox, and they vary from some of the other places where I've worked that had not really committed to Agile. The lack of commitment of resources and lack of communication are the two areas that I felt most impacted delivery and made it exceptionally hard to get work done.

Standing up the team structures is going to be hard and painful. Most enterprises don't like change. The culture can be very different, and change can oftentimes be viewed as bad or negative. If the culture is averse to change, you're going to have to invest time and effort on education, as well as getting the right people in place, and build a roll-out plan that implements Agile starting with one part of your business. You'll learn invaluable lessons from that first implementation that can be used to make additional implementations more efficient and effective in other parts of your business.

Understand the pitfalls and don't be afraid of them, but take them on knowing there is a need to shift to this new approach that is more iterative, that gets products into the marketplace on a more regular basis faster, has a faster time to market, and gets more feedback on your product so that you can iterate more on that offering, and evolve it to suit customer needs.

Jabian: What are some of the specific pitfalls to avoid, in order to stay away from the dark side of Agile?

Cameron: Not committing resources, like I mentioned earlier. Not funding specific roles on the team, such as scrum masters, and not committing product managers and UX to be embedded with teams are just a few. You really need to commit to hiring the right people. Otherwise, you're going to have poor alignment of your product as far as what you wanted to build and what ended up being built. Another compliant I hear is that people don't know what's going to be delivered. An effective planning process will eliminate that; but you need to fully commit to Agile and the resources in order to accomplish it.

The Agile Methodology has evolved to support enterprise product delivery in the last 15 years. Because of that, when you shortchange the process and/or resources, it has an effect which you are going to feel sooner or later. My recommendation is: spend the time, understand your resources, understand the methodology if you're going to adopt it, and then commit to implementing them. You will probably do your own shifting as we've done at Cox Automotive, but it's not shifting away from Agile, it's introducing new things that didn't necessarily hit the mark on level of engagement or level of communication that we felt needed to be done. But, that's not a criticism, we just had to evolve for our own needs as to what we were doing.

Those are the key elements that I feel will help, but how do you avoid the dark side? I think it is that commitment to resources, it's understanding why you're taking on this methodology, and being wholly committed to it.

INTERVIEWERS:

Nimesh Shah
nimesh.shah@jabian.com

Taarna Hopkins
taarna.hopkins@jabian.com