Agile in the DoD: Twelfth Principle

featuring Mary Ann Lapham and Suzanne Miller

Suzanne Miller: Welcome to the SEI Podcast Series, a production of the Carnegie Mellon University Software Engineering Institute. The SEI is a federally funded research and development center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University. A transcript of today's podcast is posted on the SEI website at sei.cmu.edu/podcasts.

My name is Suzanne Miller. Today, I am pleased to introduce you to myself and Mary Ann Lapham. Welcome to our ongoing series exploring Agile principles and their application across the Department of Defense. In today's installment we explore the 12th and final Agile principle. "At regular intervals the team reflects on how to become more effective and then tunes and adjusts its behavior accordingly." First, a little bit about myself and Mary Ann.

Mary Ann's work focuses on supporting and improving the acquisition of software-intensive systems. For Department of Defense programs this means assisting and advising on software's use at the system or segment level. In addition, she leads research into software topics that are germane to acquisition of software-intensive systems including the SEI's research in adoption of Agile and Lean methods in regulated settings like the DoD.

My research focuses on synthesizing effective technology transition and management practices from research and industry into effective techniques for use of Agile and Lean methods in regulated settings. This is our 12th podcast exploring the real-world application of Agile principles in Department of Defense settings. If you like what you hear today, there are already 11 podcasts in the series, one for each Agile principle behind the Agile manifesto. As a reminder the four values and 12 principles of the Agile manifesto can be found at www.agilemanifesto.org.

Today, Mary Ann let's talk about reflection and becoming more effective. How is this accomplished in a DoD setting? And, how is this different from the kinds of process improvement, continuous improvement kinds of activities that many DoD settings are already accustomed to performing?

Mary Ann: Well first off, for those who are going, *What do you mean, reflect and...*It is commonly called inspect and adapt or retrospectives—think lessons learned. Now most DoD organizations are attuned to doing lessons learned at the very end of a project: *What did we learn? What worked? What didn't? What are we going to do different on the next one we are working on?* But, we don't wait in Agile until the very end. We do it to the end of each sprint, each release, each stopping point, if you will. So, the team actually gets involved in this. They sit down, and they say, *OK, we just finished two or three weeks of work. What went really well? What didn't go so well? What should we change? What should we not do again?*

Suzanne: And, *What should we keep?*

Mary Ann: And, *What should we keep?* And, *How much change do we want to try for the next iteration*? You don't necessarily want to change everything all at once.

Suzanne: Right. Incremental change just like incremental release.

Mary Ann: Right. So, if you think lessons learned you have a context for this. Then you think, *How much can I improve each time?* And, if you let the team reflect, the team will know what worked and what didn't. They will be able to give suggestions on, *You know, if we did this thing over here slightly differently, We'd save all this time. We wouldn't be redundant*, whatever it is. Maybe it is how they do their testing. Maybe it is how they do their test criteria up front, and it didn't come out quite the way they wanted.

Suzanne: It may have to do with how they deal with their tools.

Mary Ann: Right. Or, we need a tool that does…

Suzanne: XYZ.

Mary Ann: XYZ. So, the team is a good source, and you may want to ask the other stakeholders. Because, you know…

Suzanne: They are interacting with their stakeholders, and there could be improvements there too.

Mary Ann: They are interacting. There could be improvements there too. Everybody has a different view of the thing. You want to get everybody's opinions and views and then decide what really needs to change and change a little bit to improve each time.

Suzanne: Some of you listening may be as old as I am and remember things like the plan-do-check-act-cycle, the Shewhart cycle. There is a long history in the quality assurance, in the quality management movement of these kinds of incremental lessons learned, incremental

retrospectives. So, this is a case where I see this principle as being something that takes a common best practice—that is actually common in a lot of the rest of the development context—and making it explicit for Agile as well, making sure we don't forget about it. That is what I see in the more traditional settings: putting off the retrospectives until the end when, let's face it, *I can't do anything for this project if I don't reflect until the end of this project. I can do something for the next one, but I can't help this one.*

Mary Ann: Well, not even just do something with this project, but a lot of times you do lessons learned, and then they go on the shelf, nothing ever comes of it.

Suzanne: We call those lessons recorded. It is not the same as learning them.

Mary Ann: So, with this case, not only do you do them sooner, at the end of each sprint, every two to three weeks, however long these sprints are. But, then you figure out, *What of those are we going to use, employ, for the next sprint?*

Suzanne: Or, how are we going to adapt?

Mary Ann: How are we going to adapt? You don't wait and say, well, *Lessons learned. We'll do it the next program.* No. *We're going to do it on this program, and figure out what's the most important? What's the biggest bang for the buck, if you will, on the things that we can adapt to, and are they possible?*

Suzanne: One of the things you often have when you get into DoD settings is to have multiple teams running. So, one of the things that teams in DoD settings have to be aware of is improvements locally to their own work may affect others. Usually at release time is when you will do larger retrospectives where you look across the teams that are working on a release and get everybody together and say, *What do we need to change as a whole group, not just as an individual team?*

Mary Ann: That is true. Release time is when they usually do that, but you might want to consider doing it at the end of iterations or sprints. Because, for instance, say you have three or four different teams. And, team A and team B worked really well together, and they had some kind of cool technology they were using. But, team A and C didn't have that, but they were doing similar kinds of interfaces. They might want to identify that and say, *Why don't we use this for our interface too or our tool or whatever.* That way you are not waiting until the very end of the release to upgrade the whole group as opposed to just your team. It gets a little more complicated but it's like Scrum of Scrums, your team does the retrospective and then…

Suzanne This is one of the roles of a Scrum Master, to help identify these places where as the different Scrum Teams come up with things, the Scrum Masters get together and help identify

where these opportunities are. That is one of the important things about this: tuning, this idea of tuning our work, is about taking advantage of the learning in the moment. It is really about that plan-do-act-check cycle over and over and over again and getting people accustomed to looking at their work from that viewpoint.

I can tell you from all the work that we did in the process improvement arena back in the 90s, that when you're looking at things from that perspective of, *What can I do better? How can I improve my value stream?* All those kinds of things make you start thinking about your work differently; and you start thinking about the work and the process of the work that you do. You are not just doing it. When you're mindful about it, and you know that there is going to be a retrospective in a short period of time—two or three weeks, maybe two or three releases or two or three iterations at the most—you are thinking differently. I think that is one of the things that is an advantage for some of the DoD programs that have a history of process improvement kinds of activities, is they have people, not necessarily everybody, but they have got a core of people that have been thinking this way. When those people get allocated out to the teams, to the development teams, you have already got some people that are in that mode already. So, that is actually an advantage that the DoD has, that some commercial settings may not have.

Mary Ann: It is. I was thinking as you were talking, who better to improve your work than yourself? One, people are more likely to accept it willingly and say, *Well, I could do that better*. Most of these teams, they are professionals, and they want to do better. They want to do their best thing. They can say, *Well, we did this but boy if we had that tool or different process it would make it so much faster and so much more efficient*. If they bring it up themselves, they are more likely to embrace it and do it as opposed to having someone come in and say, *Now you will do this*. That just doesn't always work.

Suzanne: I want to go back to one of our other principles, which is the simplicity principle, and remind people that you want to apply simplicity to this as well. I have done lots of these kinds of retrospectives with a whiteboard and Post-It notes. You don't have to have complicated techniques.

Mary Ann: It works better.

Suzanne: You don't have to have sophisticated tools to effectively do these retrospectives.

Mary Ann: Two or three years ago, I had the opportunity to watch a retrospective of a team. Boy, they almost said nothing because outsiders, that was us, were there. So, be careful when you're doing these who you invite. If it is an outsider, the team may clam up.

Suzanne: Yes. This principle is about the team reflecting.

Mary Ann: Yes. It is about the team not somebody else coming in and saying *Well, from an outside prospective I saw blah, blah, blah*. Unless they are actually interfacing with the team.

Suzanne: That's a different thing. Now they are part of the extended team.

Mary Ann: Now, they are part of the extended team.

Suzanne: This is an interesting principle because it is one that many people may already be familiar with in their own work. It is just being applied a little bit differently and the history of the DoD and being very active in trying to improve its engineering and management processes, actually makes this one a natural fit for the DoD. Not all the principles are a natural fit. But there is a lot of ways in which this one is.

Mary Ann: It is. And, you are practicing, and you are honing your skills. DOD does that a lot with their operational crews, why not do it with your developers?

Suzanne: Sure. Absolutely. OK, Mary Ann. Thanks for joining us today.

Mary Ann: You are welcome.

Suzanne: Well, this is our last podcast in the Agile principles series. Mary Ann and I have enjoyed exploring the 12 Agile principles with you, our listeners. We would welcome your feedback on the series at any time. If you are interested in accessing our research on Agile adoption in the DoD via papers, blog, posts and podcasts please visit sei.cmu.edu/acquisition/research/index.cfm.

Also, please remember that you can access the entire podcast series exploring all 12 Agile principles and their applications across the DOD at www.sei.cmu.edu/podcasts/agile-in-the-dod/.

If you would like more information about the SEI's recent publications in all areas of our work, you can download all of our technical reports and notes at http://resources.sei.cmu.edu/library/.

This podcast is available on the SEI website at sei.cmu.edu/podcasts and on Carnegie Mellon University's iTunes U site. As always, if you have any questions, please don't hesitate to email us at info@sei.cmu.edu. Thank you.

Thank you for listening to us for whichever of these 12 principles that you have been with us for. For those of you that have been through all 12, I apologize for you having to listen to all the beginning stuff 12 times, but we never know who is listening when. So, thank you for your patience and we look forward to your comments.