



## Applying Agile Principles in the DoD: Fifth 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 at Carnegie Mellon University in Pittsburgh, Pennsylvania. A transcript of today's podcast is posted on the SEI website at [sei.cmu.edu/podcasts](http://sei.cmu.edu/podcasts).

My name is [Suzanne Miller](#). I'm a principal researcher here at the SEI. Today, I am pleased to introduce to you to myself and to 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 fifth agile principle which says, *Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done.*

But first, a little bit about me and Mary Ann.

[Mary Ann Lapham'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 issues at the system and/or segment level. In addition, she leads our research into software topics that are germane to acquisition of software-intensive systems.

[My research](#) focuses on synthesizing effective technology transition and management practices from research and industry into effective techniques for use of particularly Agile and Lean methods in regulated settings like the DoD.

This is our fifth podcast exploring the real-world application of agile principles in Department of Defense settings. If you like what you hear today, there are already four podcasts in [this series](#), and we will eventually have 12, 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](http://www.agilemanifesto.org). We have lots to talk about Mary Ann, don't we?



---

**Mary Ann Lapham:** We do, Suz.

**Suzanne:** Should we start talking about trust, maybe?

**Mary Ann:** Trust. Trust is one of the key components and underlying tenets of doing anything in Agile. You have to trust your team. You give them a job, and you trust them to get it done.

**Suzanne:** The team trusts the product owner and the other stakeholders to tell them what they need, so that they can develop something correctly.

**Mary Ann:** Right. Trust them not to interfere—quote unquote—with their getting the job done. Not overly constrain them with overhead management things. Trust them to work through the process. The team, on the other hand, provides transparency and openness. Every day they have what we call an “[information radiator](#),” something up on the board or even in a tool so that anybody, any stakeholder can look at it and see where they’re at. So trust on both sides.

**Suzanne:** So how is that different from our traditional software development in DoD environments?

**Mary Ann:** Well, over the years, as you’re probably aware, trust has become...

**Suzanne:** Broken.

**Mary Ann:** Broken, yes. Back in Reagan’s day, *trust, but verify*. That’s very much the moniker of today. However, some organizations trust more than others. In many cases trust is just not there at all. In that environment it would be very difficult to do an agile-type of development without a lot of change in, okay, culture in the general environment.

**Suzanne:** So, you and I have both seen some settings where a development organization, usually a contractor, is trying to use Agile principles. Yet, at the same time, they are being asked to do, at least the same if not more, documentation than they had in the past.

They are asked for the detailed team metrics not just the typical management metrics. They are being asked for a lot of information that would make you think they are not very trusted. In those settings we’ve seen not very good success with Agile methods. I would assert that that’s actually one of the reasons—that there isn’t a feeling of trust. That transparency and that ability to be motivated—because when you’re not trusted, it’s hard to feel motivated to do your best work. That’s why this principle has all these things together because it all plays together. I think trust is what a lot of it comes down to.

**Mary Ann:** Not only are you not motivated, but you start to shrink back into your little shell to protect yourself because you never know when someone’s going to take a potshot at you. One thing I’ve noticed, it’s not just OK, everybody says, *Oh, the government’s the bad guy here*.



---

Well, no, they're not. I mean, they can be the guy that's not trusting. Some of these development teams and the programs themselves on the contractor's side, the team itself and their immediate managers are gung ho, and they're going to do this right. But, their senior managers aren't necessarily....

**Suzanne:** Inside their organization...

**Mary Ann:** ...Inside their own organization they don't have this trust either because, again, they've been developed and trained in the old system. This is a different paradigm. We know that. It does require you to change your way of thinking and step out there in that scary world where, *Yes, I want to trust them to do their job. Well, maybe not..*

It's very difficult. So, not only do these teams have to deal with a customer—government, in this case—who may not be familiar with this mode of operation and over the years has been, let's say, burnt a few times...

**Suzanne:** We've seen some of that.

**Mary Ann:** We've seen some of that. We all have, at least once. But, they have to deal with the same kind of trust issues within their own organization because some of the senior managers, for one thing, aren't trained well. They think it's a good idea but don't understand what it means to go that route. They still expect to see the same old stuff they've always seen and don't realize that you're going to see the same intent, but it won't look the same.

**Suzanne:** Right. Let's talk about the opposite. You and I have both had exposure to one organization where trust really is prevalent within one of the Air Force projects that we've worked with. This is a case where the program manager, who is the only government person on the project, is working with four different contractors. I can't tell you who works for which one.

**Mary Ann:** Well, you can't tell.

**Suzanne:** Because they all work together for the same goal. And, you see the trust. When they tell her that they need a different tool set or that they need to re-architect or whatever it is that they need to do, they have a discussion about it. The discussions that we've observed always start from the basic belief that they are giving her the best information possible, that they're not trying to just get more money. It's a whole different feeling when you observe these kinds of interactions. And, oh, by the way, a different result. This place has got more than 700 adoptions, voluntary adoptions, of their software throughout the Air Force and other organizations.

**Mary Ann:** Right, and it's not mandatory. It's not a program of record, although they're headed in that direction, but I don't think it's got there yet. They chose to do it on their own. *Yes, I want to adopt this because it's a great program.* And it's interesting, the contractors, when you go talk



---

to them and when we first met them, we kept saying, *Okay, who do you work for?* They tell you the name of the program. You say, *No. Who do you work for?*

**Suzanne:** *Who signs your paycheck?*

**Mary Ann:** You actually had to say, *Who signs your paycheck? Oh, that's so and so.* Their environment was the program and the program culture, and the trust was there.

I've actually run across some other programs in the Navy and Marines as a couple of other examples. I don't think that you met these folks, but they have some of that. The parts of the organization that have that trust really were working well. Then there were the guys that were a little skeptical, and you could see the tension and the difference and the potential problems if there was no trust there. The fact that they didn't have the right environment to build that trust. Some of them were headed in that direction. As they got better and better in creating the environment for it, you could see the change. It was so evident when you walked in the room.

**Suzanne:** I don't want to assert or tell anybody that this organization [the Air Force program] got to that point overnight.

**Mary Ann:** Oh, heavens no.

**Suzanne:** They have been using Agile methods for more than 9 years now. This is something that did occur over time, and this is something that happens over time.

Let's go back to some of these other elements: *motivated individuals* and *providing an environment and support that are needed by the team.* Those are the other parts of that principle. They are actually one of the ways to build trust, in my mind. If you help people become motivated—by giving them the right skills, by helping them to understand the mission, by actually believing what they say, unless you have a really good reason not to—you get that motivation. But, one of the other ways you get the motivation is if the environment and support are there for them to do their best work, right?

**Mary Ann:** Very true.

**Suzanne:** That is one of the things—when I talk to people about this principle—as soon as we talk about environment and support, they say, *Oh, well, we don't have enough money for big automated tests suites. We don't have the money to set aside all this extra space* and all the rest of it.

I've seen successful teams take over a conference room for two weeks, and then give it to the next team. I've seen people, even in the DoD, use open-source tools, which are free, that help them get started with automated testing. As a matter of fact, some of my friends in the automated testing domain say the best tools that are out there tend to be the open-source tools. So, it doesn't



---

always cost management a huge amount to create the environment that the teams are going to need to succeed. But, they've got to be willing to ask what's needed and figure out how to provide that support.

**Mary Ann:** That and they also have to be willing to be the cover, if you will.

**Suzanne:** The top cover.

**Mary Ann:** The top cover for the team so that the folks who are not privy to what's really going on, the guys that sit by and say, *Those guys over there, they're screwing off*. Because they're having what looks like parties, but they're not. They're having meetings where the team's animated and putting things on the wall and running around and doing stuff that you would do in a normal Agile environment, which is totally different than your waterfall environment.

**Suzanne:** Typing in documentation.

**Mary Ann:** Typing in documentation. People look at it and say, *Something's going on over there*. Then, they start taking potshots at the group.

I want to go back to the motivated individual thing. Every once in a while you run into somebody who's not motivated to do their job, but most people are motivated. There's also a part of the motivation that is kind of a corollary to this. A lot of people—when you're changing from an older environment, the old traditional, to the more Agile environment—they may be very, very motivated, but they just cannot work in that more open environment. They just can't change to do that.

They're waiting for somebody to feed them the information. They just can't do that, and that's fine. But you have to make sure that, when you go there and you identify those people, and a lot of times they self-identify and say, *I have to find something else to do*.

**Suzanne:** *I don't want to play*.

**Mary Ann:** *I don't want to play this. I can't do this*. This is totally anti what I'm expecting. I just can't make the transition. You need to be able to say, *That's fine*, and find them a home someplace else so that they're not punished for the organization changing. That's very hard. Because sometimes, that's one of the fears. People say, *Well, where do I fit in this new organization? I don't see anything that says program manager, for instance*.

**Suzanne:** And testers? You want the tester to be on the team and maybe even code.

**Mary Ann:** And, work next to and actually almost touch this developer. I mean, there's some cultural things here.

**Suzanne:** There's cultural aspects that come in here.



---

**Mary Ann:** You have to be aware that motivation has many aspects, and you have to make sure the people have the right mind set.

**Suzanne:** Another aspect of motivation is skills.

**Mary Ann:** Yes.

**Suzanne:** I know one of the methods in Agile, in extreme programming, is to do pair programming. I've seen examples of that that turned into a real disaster because you had this very senior developer that knew everything about the system and somebody more junior. And, the senior developer felt like they had to continually prove their worth. So, the way to do that was to not work productively, not mentor the other person but, essentially, show them up, which is a disaster when you start trying to build trust among the team let alone outside of the team.

**Mary Ann:** That gets back to the fear factor of, *Where's my job going to be if I tell you everything I know?* Historically, the one with the knowledge is the one that's going to survive.

**Suzanne:** *They can't fire me because I know too much. I'm the only one who knows about X.*

**Mary Ann:** *Even though it's documented, I am the only one that knows why it got there and how this happened and this, that, and the other.* We all see that, and we've all been exposed to that. That's another force that you are playing against when you want to go to the Agile environment.

**Suzanne:** Because it's about the team.

**Mary Ann:** It's about the team. It's not about the individual.

**Suzanne:** ...and whatever the team needs to succeed at the mission.

**Mary Ann:** Right. And there are no heroes so to speak. The team's the hero. Now, there might be somebody that says *A-ha* and that breaks the code for whatever needs to be done and helps the rest of the team get along.

**Suzanne:** But, the next time it will be somebody else.

**Mary Ann:** It'll be somebody else. That's one of the reasons why some of the places allow the team to decide who gets the bonuses.

**Suzanne:** We've seen that.

**Mary Ann:** That's another principle, rewarding your people.

**Suzanne:** When I think about the environment—let's just talk about *environment that's needed to support* and how it may or may not fit with traditional sort of DoD kinds-of-environments. One is from a physical-space viewpoint, when you have a co-located team, which you can't



---

always have (we recognize that.) when you have a co-located team, you want all those people to be able to be working in the same space. Sometimes you need to move them physically into an area. That can be a problem if you've got people in different functional offices that are accustomed to having facilities deal with each department individually.

**Mary Ann:** That gets back to how a lot of the DoD personnel, they're allocated to functional groups because that's how they're trained, and that's their career path. To violate that, on one hand, you've given them a really cool opportunity, but you've violated where they see their career path going.

**Suzanne:** Which is their success.

**Mary Ann:** Which is their success and their increased earnings, their promotions, and all that stuff. That whole part of it has to be considered when you're dealing with the environment. Environment is not just the technical environment,

**Suzanne:** That's right.

**Mary Ann:** It's the rest of the environment. We know with a lot of the teams, you might have one team in one building, one in another, [or] even worse, in different parts of the country.

Now, initially, way back when Agile started, that was a real problem. Fast forward 10 years, 12 years now, and there's so much available from a technology standpoint.

**Suzanne:** Although I still prefer working head-to-head, face-to-face.

**Mary Ann:** I know, face-to-face.

**Suzanne:** And, most people do.

**Mary Ann:** And most people do, but you can get around it. You can't always have co-location. And you can get around it because you can use Skype and VTC's. Some organizations actually have a flat screen on the wall and it's on all the time. You need to talk to Joe, and he's in California, and you're in Texas. You'll say, *Hey, Joe, are you there?* And Joe pops up and says, *What do you need?*

**Suzanne:** You can't build the trust without the face to face contact.

**Mary Ann:** You can't build the trust if they haven't at least had some face-to-face meetings, and that's an expense.

**Suzanne:** I want to talk a minute about some of the things we've seen that are workarounds, if you will, in terms of trying to build an environment.



---

There is the one team I talked about that had one conference room, and teams would rotate in and out of that conference room. They wouldn't get it all the time, but at least once every three or four cycles, they would have that work space for their two-week sprint.

Another environment just took over. They had people all in the same building, but they couldn't move their cubicle space because of their facility's restrictions. They had some empty cubicles. They got facilities to basically pull the walls out and just create a corral. They brought in rolling tables and things like that. They got IT to put the cable drops in that they needed for network access.

So, they used, essentially, social capital with their friends to get the workspace. That allowed them to prove, over the course of a release, that they could do a lot better this way than if they were just working at their individual terminals. That was a pretty innovative way of doing things. It did mean that later they had to grovel before the facility gods, and get some things done later. They at least piloted it. They tried it, and they showed what the environment change could do. That was important.

**Mary Ann:** That's one of the things, *trust them to get the job done*, and they'll figure out how to do it.

Do they have to figure out how to get all in one space? Do they have to figure out how to solve some technical problem? If you give people a challenge and give them the boundaries and say, *Go figure it out*.

**Suzanne:** And it's a reasonable one.

**Mary Ann:** And it's a reasonable one.

**Suzanne:** It's not a death march

**Mary Ann:** Yes, it's not a death march. Most people will rise to the occasion, and some of the stuff they come up with is totally amazing.

**Suzanne:** I'm going to put a plug in right now for [our blog series](#). One of the things that we're talking about in our [Agile blog series](#) is the kinds of indicators that help you to understand if you have an organizational, project, or team environment that is going to be conducive to Agile. If you are interested in this environmental discussion, we do already have some [blogs](#) posted that start dealing with this. We will have some more coming up.

**Mary Ann:** The keywords there are [readiness and fit](#).

**Suzanne:** Oh, yes, in terms of the blog, yeah, look for [readiness and fit](#). I happen to be the author, so...



---

**Mary Ann:** She keeps forgetting to tell everybody. It's readiness and fit. If you looked at it, and didn't have a clue what it meant, now you know.

**Suzanne:** Anything else on this topic that you think we should cover? I could talk for days on this.

**Mary Ann:** We could for days on end on any of these.

**Suzanne:** I think we have given people an idea that the environment and the trust are key elements of making an Agile transition, and if you don't have it already, you've got to find ways to build it.

**Mary Ann:** Right. It's trust, not just between the contractor and the government, but trust within the government side and within the other stakeholders. And, even within a contractor company across the board there, you have to have trust up, down, sideways. All over. If you do, you are going to have much better chance of success.

**Suzanne:** Exactly.

**Mary Ann:** It doesn't mean you'll be totally successful, but you'll certainly have a better chance because it takes work.

**Suzanne:** All right. Mary Ann, thanks for joining me today.

**Mary Ann:** You're welcome.

**Suzanne:** In the next episode in this series, we will explore the sixth Agile principle, which we sort of brought in a little bit of in this discussion. Which says *the most efficient and effective method of conveying information to and within a development team is face-to-face conversation*. We did kind of preview that.

**Mary Ann:** We did.

**Suzanne:** We do know that's a challenge in today's geographically distributed environment. But, we'll talk some more about that next time.

Listings for papers, blog posts and podcasts related to our research on Agile adoption in DoD can be found at [www.sei.cmu.edu/acquisition/research/](http://www.sei.cmu.edu/acquisition/research/)

If you'd 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 [sei.cmu.edu/library/reportspapers.cfm](http://sei.cmu.edu/library/reportspapers.cfm).



---

This podcast is available on the SEI website at [sei.cmu.edu/podcasts](http://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 e-mail us at [info@sei.cmu.edu](mailto:info@sei.cmu.edu). Thank you.

*Editor's Note: This podcast has been edited to increase readability.*