

79. Agile Today from the 2020 UMD Symposium

with JJ Sutherland, Christine Brennan Schmidt & Richard Wyatt

00:05 JJ Sutherland: Scrum really fosters a sense of team and a connection with the rest of your team that I think is translatable very easily into a remote world.

Christine Brennan Schmidt: I think people felt more like they understood what their new web product was and how it came about, and therefore it helped them as they think about the future. They understand what they might not want to do, or what they might want to do, because they understand how we got there.

00:37: Richard Wyatt: Because the world is changing so fast, organizations need to be agile, so the question is, Where do all the project managers go?

00:47 Kendall Lott: Welcome to Episode 3 in our 2020 series from the University of Maryland's annual Project Management symposium. This time the focus is Agile: Agile in a virtual environment; Agile for website development and design; and finally, the role of project managers in an agile world. You thought you had heard it all on Agile, but you haven't. Think pandemic and lockdown as new triggers for getting key parts of Agile correct. Here we provide highlights of the presentations of JJ Sutherland, Christine Brennan Schmidt and Richard Wyatt. Each of these presentations made me want to dig in for more extensive explanations and follow on conversations. Our episode takes you deeper into the presentations – the director's cut. I followed up on what made me curious, but you should go see their full presentations on the catalog at UMD's Project Center for Excellence website. Google it, you'll find it. 125 bucks, over 40 PDU's. Do the math.

Announcer: From the Washington DC chapter of the Project Management Institute, this is PM Point of View®, the podcast that looks at project management from all the angles. Here's your host, Kendal Lott.

KL: These are the days of disruption. As the rate of change accelerates – again – our organizations are forced to adopt more Agile practices. And that was even before we experienced one of the great disruptions of our time, the pandemic. The pandemic has forced organizations and individuals to quickly and drastically alter the way they operate, and move to a virtual environment. And it probably comes as no surprise that Agile organizations have handled that pivot more easily than older, more entrenched entities.

We kick off this episode with JJ Sutherland, talking about Distributed Teams: Scrum Teams in a Time of Disruption.

JJ Sutherland (02:32): And so I've been thinking a lot recently about, obviously the rapid change that we're going through, and this is why Agile and Scrum have become so prevalent in project management, is because the plans you have are going to change rapidly. And to be able to do that, you need to have an organization that can change rapidly. And that has really been clear this year, more than any other time in my lifetime. Any kind of plan that was made in January of this year,

when you're doing the annual plan, that plan vanished in the course of about a week.

And this is a quote that's often attributed to Darwin. "It's not the strongest species that survives, nor the most intelligent, but the one most responsive to change." And if your work and business and project environment is dependent on things being one way and then you have, all of a sudden, this suddenly remote world, and you can't make that leap, your business will die. That's going to happen to a lot of companies.

JJS (03:33): My name is JJ Sutherland. I'm the CEO of ScrumInc. ScrumInc. is a training, coaching and consulting company that helps organizations go through Agile transformations, and also individuals, but mainly corporations, and we've been doing that since 2006. And it's the company founded by my father, Jeff Sutherland, who invented Scrum back in 1993.

KL: Wow, so your dad invented Scrum?

JJS: So it was officially called Scrum in late 1993. The first Scrum sprint was January of 1994. There's a paper in *The Harvard Business Review*, published in 1986, I believe, by two Japanese professors, business professors, Takeuchi and Nonaka, and they were looking at the best teams in Japan. And they found that the best teams in Japan – and the best teams – had common characteristics: they were cross functional, they were collaborative, they were self-organizing, they're self-managing, they were reflective, they tried to get better all the time, and that was sort of what they're trying to do. And these two Japanese professors thought it reminded them of the scrum formation in rugby, where everyone works together to get the ball down the field.

JJS (04:50): So when this world is suddenly remote, thinking about some of the challenges to wrestle with, you have to worry about communication, because how people communicate in a virtual space is very different. Because you announce your presence when you go into the office, when you're in a physical space, just by showing up. But in a virtual space, you announce your presence by adding value. So you have to figure out how you do that, and how do you communicate to your peers, to your teams, to your company? People are like, "When does work start?" "When does work end?" "When do I deliver something?" "When is that deliverable?" "How does that work?"

And figuring out how your daily life works is incredibly important. Because you want to know what you're supposed to do. You want to know what other people are doing, because you don't want to be doing a bunch of work, because other people are doing different stuff and you don't have that communication about what *is* being done. But it's really easy to go down paths that the customer doesn't want. So you really want to make sure that everyone's doing what *needs* to be done. And that can be very difficult. So the key, of course, is communication.

KL (05:56): What you're really about was this idea of the distributed Scrum team. I guess, driven by the business problem of remoteness.

JJS: Yeah, so in the Agile community, there's always...there's always been a thing of co-located teams are better than remote teams, and I think we've proven that wrong now. But all of my stuff was live, in-person, consulting and training and coaching. And on a Saturday in March, I called up the product owners and said, we can no longer do in-person trainings; it's just irresponsible. And this is about a week or two before everything got shut down, and I said everything has to be online,

and our next class was on Tuesday. I remember talking to the woman at ScrumInc who's the head of our Public Education Unit, and she said, "Really, you want to be on online on Tuesday?" And I said, "Yeah." I said, "You know what? The first one probably will be bad. But you know what? The tenth one will be a lot better. Let's get to the tenth one as fast as we can."

KL: What is it about that learning curve? Is it just simply people are uncomfortable, unused to it? What's the block there?

JJS: What I have found is that the teams that were Scrum teams adapted far more quickly to working remotely than the teams who didn't. And I think it's for a couple of reasons. One is one of the major ideas of Scrum is making work visible. And because, what happens a lot is that everyone's not sure what everyone else is doing, so there's a lot of re-work. People are like, "Oh, I thought Ann was going to do that, but no, I did it too." And so we both duplicated work because there was no transparency. So that's one, it was just the visibility of work.

I think the other thing is that Scrum teams are inherently close and the human side of work and life is part of Scrum and it's part of business, whether you say it or not.

KL: Wait, let me check. By "close," you mean shared experience, intimacy.

JJS: Yeah, shared experience, intimacy. I think one of the things that bonds people closer than anything else is working together on a shared problem. I think that's an incredibly powerful human bond. And Scrum is set up to do that.

To also inspect that. Like how is the team working? Can we work better together? And certainly in my team and talking to everyone at ScrumInc, there's a certain amount of fear. For the people who are totally alone, there's loneliness. And fear is a completely rational response to a pandemic. A sense of disconnected-ness, and how do we decide when to work? How long to work? What's done mean? And Scrum tries to address all of those things, and really fosters a sense of team and a connection with the rest of your team that I think is translatable very easily into a remote world.

KL (08:56): So then we move on to the Scrum remote team checklist.

JJS: These are two, the three things to do to be remote: Basic technology for collaboration; frictionless and secure access; and a culture of trust, inclusion, and making work visible. And that's really important, especially in a remote environment.

What is the work that is being done? How do we make it visible to everyone on the team or team of teams? What is being worked on? And that becomes much more difficult in a remote environment, so it takes effort, it takes more discipline.

Then the third part is intentional transparency and vulnerability on the part of leaders and key members. And that really rings true to me. And it's really important for leaders and managers, and team members, to allow those feelings out. To say this, Listen, this is tricky, we're going to figure out, but we have that culture of trust, trust in each other, to get through this.

And one way to do that is what we call storming, which is people working together to solve a problem. For example, I've been working on a large document that I have to deliver by tomorrow,

with three or four other people. One of the things we do is we all work on it at the same time. We're using Google Docs, but there's plenty of other collaboration tools you can use, and what it allows us to do is really share our thinking. It really builds connection with each other and pulls teams together, and remember, that's what you really need to build with your teams in a remote setting. It's some sort of sense of a connection.

KL (10:29): You talked in your presentation also, I thought it was interesting, you have a discussion on it, I'll invite people to go look and listen to on their own, but about trust and shared vulnerability. And I was wondering, is that a concern we face in the context of being more Agile, or is that more related to the virtual nature of the problem? You decided to highlight that, and I was wondering if that's about how we work together in this way, or about the nature of being virtual and remote?

JJS: I think it's much more broadly applicable. I think it's really important in the remote world, because it is very easy for people to be scared, to feel alone, to not know what the future holds, and those are not irrational things to me. I mean, it's scary. But also it's important to say, as a leader, "I'm scared too." And showing that vulnerability, whether you're remote or not, allows you to connect with your team, your people and your employees, as people. And showing yourself as being a real person breeds connection, and I think that connection breeds trust, and I think that's what really what we need to try to instill in our teams. Trust. Loyalty. A sense of role, a sense of self-worth, a sense of, "If I do this, I know they'll do that." Like a basketball team you know you see a professional baseball team, somebody throws the ball over their shoulder, not even looking, because they know, their teammate will be there to pick it up.

JJS (12:05): Co-location, where everyone's sitting at the same table, masks bad communication. you can be really sloppy when you're all at the same table and everyone can hear each other. But when you have to be disciplined about it, because everyone is not in the same room, it's a little harder, but it just takes some discipline and actually improving what you're doing.

I've been talking to CEOs recently, and calling around to ScrumIncs clients, and saying, "Hopefully, this will end. People will go back to work." And I said, "Well, what are you going to keep? What's good about what has happened? Like what have you learned?" And a lot of it is staying in deep communication all the time, rapid iteration, and making sure that everyone's connected. One group I work with is a global material science company, and they had a lab in the United States, they had one in Europe, and they had one in Asia. And before Covid, the lab in the United States was definitely the big dog in the room. They sort of handed out to the other labs, what they were going to do. But once everyone was remote, all of a sudden, the labs came to...these global labs came together as one large team to work on things. And it wasn't uncommon, where the person in Singapore or wherever would say, "Okay, I have an idea, let's go down this research path." And they're working together in teams, because everyone was remote!

KL (13:39): If I had been operating as a Scrum in the before times, now that I must be remote, what do you highlight as the most sensitive to the remote factor?

JJS: So I think that one thing is, I think one of the most important things in Scrum is Ready Backlog. And that means it's... It's refined, the team knows what they need to do, there's some sort of definition of done. They need to know what done means, and it's visible to everybody. Now, every Scrum team should do that all the time.

KL: Yeah.

JJS: But you know, they don't. Not everyone does. And one of the things that remote does, it makes that really clear when that doesn't happen.

What backlog is, is just a long list, a prioritized list, of what needs to be done. A product backlog items, those things on that list, you have to remember, are place holders for a conversation. Say, Okay, we need to update whatever. But the person who's putting together that list (and in Scrum that's called a product owner) they need to be able to talk to the team about what that means. And so it's really a conversation, rather than, "Here's a requirements document," thrown over the transom.

What's really good is everyone needs the same access to it, and this can be tricky with corporate firewalls and all the other kind of stuff, so figure out some way to make sure everyone has access. And then you want to bring those people together, to have those conversations, what we call backlog refining, to getting that list both ordered and estimated, and really ready. Because we want people to engage with the work together. And so, what we call a definition of ready is, ready to actually work on it. Because what we found is, if that's not really clear, people can spin around for days trying to figure out what that thing that needs to be done actually is. So it's really important to make sure you have a very actual, ready backlog.

KL (15:30): You highlighted back here that the ready and transparent backlog was the magic of Scrums, in the remote one as well. And you mentioned something in your presentation that I wanted to ask about, which was the definition of "ready." You said, "Don't let the definition of ready slip." You said distributed work requires more discipline, engage with it. Tell me about this definition of "ready." What's going on there?

JJS: So whenever a team has something that they're going to be working on in the next sprint and they want to bring that work, that product backlog item into the next sprint, it needs to be, to meet some definition of "ready," which is, Okay, what's the value? Why are we doing this? What does "done" mean? Is it...when I show this at the end of the sprint, how do I know when I'm done? And does the team understand enough of what needs to be done that they don't have to ask anybody? Because a huge waste is when people are like, "Okay, this is what you need to do, what do you mean exactly?" I don't quite understand, so I spent three or four days figuring that out. Rather than figuring that out ahead of time.

KL: Wouldn't a clear requirement tell you what done looks like?

JJS: Not always. So I'll give you an example. I was talking to this marine, and his job was doing the sort of playbooks for the Navy. Sort of, "Okay, this is how we do this." And he had a deadline of, I don't know, every eight weeks or so, like we want this new playbook and how we handle this set of problems.

And I said, "Well, when are you done?" And he said, "Well, I'm done when the date comes." I said, "Well, what if you define done up front, so you're not just spending the last three weeks polishing an apple?" And he looked at me and said, "You have saved half of my life. Because once I know what it means to be finished, and having a very clear definition of what finished means, I can stop."

JJS (17:33): You write a plans document, that's when you know the least. You know the least about how hard it is to do the work, you know the least about whether people actually want this, and if you don't test your beliefs regularly, you're going to make the wrong thing. People usually don't know what they want until they see it. And they can spend weeks or months or years talking about what they want, and describing it very carefully. But once they see working things, they change their mind. And I think that that is the reason – because it's the illusion of control that the world won't change, that the needs of the market won't change from the time we write these requirements. And as we were saying at the start of this conversation, of course they'll change.

And so what we want to do is say, “This is what we think we're going towards. Let's define really carefully what the stuff we're going to work on in the next month or two. But then let's do some work, produce something that we can show customers or stakeholders or whoever, and test whether what we're doing is actually what they want.”

JJS (18:43): We know that rapid change is not a surprise. And we don't know what's going to happen. No one does. Over the next months or years, this is a time of rapid rapid change, so what we really need to do is, don't wait. This is a perfect time to transition your organization from one way of working to another way. Because you have to. But I'm noticing with a lot of ScrumIncs clients is, for the first four to six weeks, they kept saying, “Well, when things go back to normal, we'll start up; when things get back to normal, we'll continue our transformation. When things get back to normal...” And now they're calling up and saying, “We don't know when that's going to happen, and we don't know what normal is. What is it going to be? A month, two months, three months, six months, we just don't know. So we're going to start now. We're going to start working on our transformation now.” So much has been thrown up in the air, actually doing some real organizational change is easier than it was when you have all the, sort of, forces, that, as many of you know, resist the change.

So just a couple of things that I've found really important. One is that scaled daily Scrum, where a representative from each team that needs coordinate/talk to each other, and not – just like the daily Scrum gives the individual teams a connection – that scaled daily Scrum gives the group of teams a sense of purpose, a sense of connection, a sense of alignment, a sense of sharing solutions and victories. And so I really think that that is a... It's a really powerful tool, and it only takes 15 minutes. It's not really hard to do.

So what we're also doing is not just a scaled daily Scrum, but also a shared retrospective, where the Scrum masters from all the teams will get together and say, “These are the common problems we're struggling with. Have people found solutions?” And they want a...part of it is also sharing. Also resolving issues. How...are there tensions between teams?

So that sort of a scaled retrospective has been really powerful for us. And I really have found if that, you know one team has come up with a solution to some of the emotional issues that come up in this environment, and they're able to share that. Like one of our teams, what they do at the end of every day is they have a dance party. Someone picks a song. They're all on Zoom. And they dance for five minutes.

KL (21:09): So it's all about the team – newsflash, eh? But working head-to-head on a project, even in a virtual world, builds a healthy team spirit. And it's that much harder. Plus, the transparency requirement of Scrum lends itself well to the virtual operation. You just have to be more intentional

about making your status clear, and be able to clearly define your “ready” state, as well as your definition of “done.”

KL (21:42): The focus of our next presentation is website development and design. Christine Brennan Schmidt has been designing and managing websites for two decades, and is well-versed in many of the methods and principles in a project manager's tool kit. Not long ago, she decided to go Agile on her web design projects, and she hasn't looked back.

Christine Brennan Schmidt (22:03): I work at the American Chemical Society, and I work in a group called Web Strategy and Operations. We handle the main website, plus other digital channels, and I do a lot of project management. I do some product management. I do some consulting in terms of other projects, where User Experience and Usability are involved. I do a little bit of everything, and I'm a big proponent in terms of project management.

KL: I took... Your presentation is like Agile for Reals. Like this was someone who was saying, “Okay, so here's how I really use these concepts in Agile.” Your title was officially, “Adapting Agile in a Historically Non-Agile Environment.” But we hear so much about Agile, and have for 20 years. We've had podcasts, everyone does it as a track at the symposiums. So in your mind, why was this needed?

CBS: Well, all the reading and all the stuff I see with Agile, it works really well when you're doing software development. I'm actually working in something that's not quite software development. It's website development, content-oriented, design-oriented. And when I talk to folks, a lot of folks don't do Agile when they're talking about design or content strategy or some of those other things. But a lot of people work on sort of an agency model: tell me what you want, and we'll go off and do it, and we'll bring it back to you and then you'll have it.

So we wanted to work as partners with these groups in developing how this product developed as equal partners, and not they tell us what they want. We bring some expertise to the table, we help them work things out, and we learn a lot about them and they learn about us. And so Agile is really, if you take the principles, it's a really great way to build that partnership and develop a really good product.

KL (24:07): It's not so much just about the development concept, but that it was a way of understanding or developing a partnership to do something that's not traditionally developed in an Agile thing. It actually drove you organizationally for interaction.

CBS: Absolutely, it's the basis for how I get the whole team to interact because they're in different departments and they're used to... Historically, there has been some, okay, tell us what we're going to do, everybody goes off to their corners, they do it, and then they come back and say, “Here it is.” Versus actually working through it in a way that's more collaborative. And again, there's more information sharing. We see a lot of buy-in and a lot of learning on that. And understanding. It's really remarkable.

KL: You actually open with that, with this is the Agile principles applied to people.

CBS: Basically, this is going to be a story about how I use Agile principles to do Agile project management in web development. We typically don't use Agile for this entire process and so, I'm

going to talk a lot about the principles and how we apply them to people. When I start a project, I want folks to understand that we want to have an Agile mindset. And I use the Agile Manifesto to clarify the values. And these are the values: individuals and interactions over processes; working software over comprehensive documentation; collaboration over contract negotiation; and responding to change over following a plan. Those four values, which are in the Manifesto, are usually applied to software development.

But we can modify and take those values and apply them to web development, so that we can enable collaboration among the entire team. And I wanted to see the team determine how to get the work done at a sustainable manner. And all those things will enable us to build a better product.

KL (26:08): You describe that it's a mindset, and in fact, you start out by teaching people about Agile, even when you're sitting them down to get started on it. What's going on when you say, "I start out with getting them in the Agile mindset...?"

CBS: Well, I want them to understand what I think the Agile mindset...or the principles, showing them those principles at first really helps them to understand that things aren't going to be how we might have done them before. For some of the earlier projects. I'm actually on my fourth project now, and people are going, "Oh, this is how Christine works." And they're good with that.

But they need to understand that they have to be at meetings every time I have a meeting, which might be every week. Right now, we're running a slower project every two weeks, and they're not really used to that. But they need to understand why we're doing that, and what the gains are. So it's really about level-setting and making sure everybody understands what I'm thinking when I run a project.

CBS (27:11): So why do we want to do it in Agile? Well, there's a lot that goes into web development. It includes design, not only visual design, but how does it lay out? How do we get the content? How do we get the content in? And there's Analytics and all sorts of things that go into it, and we really didn't understand what that final product would look like.

And so we wanted to leverage the iterative cycle that you get out of Agile. And I've taken a meme that you see often with this circle, showing the iterations, and I sort of changed the language around it, where we have a user story, we go to ideate a solution, we prototype, we then test or review it, then we edit to create something, we get feedback and then that either ends up being refined further as we learn more, or it feeds into the next step of the development of the website.

KL (28:09): I found it interesting, when I just kind of re-capitulate, or re-listen to the whole thing again: You are less blending techniques, but rather that you selected. You kind of made Agile Agile.

CBS: I think that's some of how I work my life in general is that, okay, these are tools. Doing Scrum is a tool, doing Kanban, doing Agile, doing Waterfall, those all give you sets of tools and methodologies to get something done. But if you understand why they're being done, you can select and say, "Okay, that doesn't work for this," or "That doesn't work for this, but this works better," and then you can start, if you understand how it all developed, you can actually develop a method, something that works for this particular environment.

CBS (29:01): Now, when we talk about Agile, a lot of times people think Scrum, and managing the Backlog. And Scrum is great. We use Scrum for DevOp. For technical... develops functionality on our content management system. But Scrum doesn't really work as well for the design process, when we're trying to do content evaluation and trying to figure out the IA. We're doing user testing, we're doing visual design. Some of that actually, I would think, falls more to something more like Kanban, but... there's a lot of diversity here.

I would say that the tasks are not as homogeneous as you see in software development, so I didn't choose a single methodology. I really just stepped back, and I took pieces, or aspects of the different methodologies, and applied them, and I really used Agile principles as guidance for how we ran the projects.

CBS (30:01): I did highlight some of the core principles of Agile that that are more important to us in this web development type project. I do want to say that the first one, the highest priority, is to satisfy the customer. Often times we think about the customer, as, oh those are the owners of the product, but actually our customer in web development are the those who use the website. So that means we are trying to develop the best user experience. And sometimes we have to use analytics, we use a variety of usability testing.

The other piece that's very, very different for our approach is that business people and the developers must work together. And when I say developers, I mean all of our web folks, working with all of the education folks. And they weren't quite used to doing that, you know, having a weekly meeting and us asking them questions constantly. One of the pieces of feedback that I got in our lessons learned was, "Wow, that was a lot of work. But that was a great investment."

KL: Why do you think he said it was worth the investment? What was he comparing it to?

CBS: I think people felt more like they understood what their new web product was, and how it came about, and therefore it helped them as they think about the future. "Okay, we want to do this." "Oh yeah, we talked about that." They understand what they might not want to do, or what they might want to do because they understand how we got there.

CBS (31:36): This is the other hat of the principles of Agile, and what I highlighted and bolded was Simplicity. That was really key when we go through working on a team like this, because oftentimes when we're looking at a website, we think, "Oh, I want this piece of functionality. I want a counter here. I want this piece of content to appear magically, or to have this functionality..." And Simplicity, if we take that value and apply it, and start saying, "Do we really need that to launch?" (And I will talk a little bit more about the MVP, the Minimum Viable Product, and how we approach that and what the benefits were of using an MVP approach.)

KL (32:25): So of the 12 principles, you flagged about 6 of them as important. And of those, you really picked on Simplicity. Tell me why that one's jumping out at you, and you also noted that that one does go to that Minimally Viable Product. Why is that thread the one that you're finding so important in using Agile in a non-Agile workforce?

CBS: Simplicity is really good, especially when we're talking about website work, because a lot of times we don't... We make assumptions when we build something that it's going to work like this, and the audience will love that. And unless there's a lot of testing done – we do some usability

testing on prototypes – but a lot of times you need to have that product that's actually working with real content, with real workflow, really out there with the audience before you understand that...how they will actually use it. So what we really want to do is start with that baseline, and then learn and grow from there.

KL: It's fairly common in Agile. This is a common thing. But you really seem to be pushing it. Why was that such a lynch pin for you? Had you had experiences trying to do these techniques and not focusing on that?

CBS: Yeah, I think we have, a lot of times people ask for more than they want. One of the earlier projects I did, they were blue skying at the beginning. They said, “Oh, we want automated this. We want automated that. We want this to show like this...” But once we launched the product, we realized that they didn't have enough content, they didn't have enough people working on content, their audience wasn't really what they expected, so really helping people focus down on that MVP really helps.

And if we had a deadline, we wanted to launch by this meeting, we have these series of national meetings and other meetings, and if people say, “Well, I need it to launch by this date,” we'll say, “Okay, we're here. And if we have time, we'll do that, but if we don't, what's the simplest thing we can we can launch that'll get us to what we need to do?”

Being involved in that whole process, I think, helps them to accept that, and understand risk, and understand deadlines and all of that. If you don't do that, it's a black box, and your business unit can really... It really puts them on a different page

KL (34:43): How do you handle risk?

CBS: I think risk and MVP are tied together very closely. So we talk about, “Okay, we have a need to do this by a certain time. How are we going to manage that?” And so one of the ways is, “Okay, what's the minimum we need to do? What are the risks of getting it... of not getting there?” I think MVP helps us keep things really focused, so that we do make those deadlines, or we understand what needs to happen by a certain time.

KL: Do you discuss risk outside of MVP? Do you discuss risk in and of itself at all with your team?

CBS: Well, and I guess we don't use the word risk, as what are challenges? What are the things that we need to watch out for? Like, again, I think one of the biggest ones we have is system upgrades. All of a sudden there's a... I find out that the whole CMS is being upgraded. So how does that impact it? Or there are some other projects that are higher on the list that the organization is doing. What happens if we get pulled into them? So we talk about them and we try to understand what we're going to do in those cases. What's the result? Does that mean we delay? Does it mean we have to scale back? Does that mean we have to figure out other resources?

KL (36:14): You put a whole big chunk of your presentation around engagement through communication.

CBS: We have these regular project meetings, they were usually weekly. A lot of work went into making sure that we knew what would be presented, we knew what questions needed to be asked,

or what needed to be answered. And we also allowed for input from some of the non-core members. So we had, in some of these meetings, we had some of our marketing folks, or people who were on related projects, who were interested in this project. And we would ask them for input, too. And occasionally we would get some really good input that was germane to the progression of the development of this website.

After the meeting, we would do emails that included the slides. I always tried to set out an agenda. I wasn't always successful at getting it out more than ... I never went into a meeting without an agenda, but sometimes it was a couple hours before. I didn't want to send it out too far in advance, because I wanted to use that time in between to develop what the agenda was, and also you want to keep it up front into people's minds. You don't want to send it out a week ahead of time, and they read it and then they forget about what it is.

CBS (37:33): The term I don't think I used in this presentation at all, but means a lot to me, is that servant-leader, where I'm actually... I'm here to make sure that you guys can do the best work you can. How can I facilitate it? How can I make sure you're talking to each other, that you have what you need, that you feel uncomfortable asking that question, well why don't we ask it this way? And why don't I get some people starting to work on that. And so I put bugs in people's ears in terms of what they should think about or what should they look at. And then people come to meetings fairly prepared, and ready to talk, and get things done.

KL: Yeah, the preparedness, you had mentioned in there, and it sounds very... I think that was an example of your skill as a leader, to the extent that people come prepared, ready to do the work, and to contribute.

CBS: Yeah.

CBS (38:24): Good communication takes a lot of work and discipline, and maybe that's why it's not being done in these other projects. And so I think the biggest thing I see that came out of using the Agile principles is developing that rapport and trust so that we could do good work. Facilitate that through good communication. We feel like these websites were somewhat innovative. We actually created new logos, as well, and the business units were very, very happy. And I think one of the pieces that came out of our lessons learned was that, "Wow, that was a lot of learning on both sides, about product and development and websites and stuff like that." And everybody really speaks highly of having gone through these projects. Even though they had to put a lot of work into it.

KL: So, what's the punch line? In the end, what does a PM need to take away from this that you found captivating, that you want them to hear?

CBS: So I think as a PM is, we don't undervalue the importance of having good communication. And that communication, I think, helps build that rapport and trust. And once you have those things in place, you help... with good communication, you deal with risk and you deal with getting the work done. But it helps form that team. And I think once you have that sort of "teamness" going on, you get a better product.

KL (39:54): Agile, it turns out, forces more stakeholders to participate in the project process. This in turn gives them a deeper appreciation for what's really needed, and what it takes to get there. We heard all of this in the Agile documentation. She's living it with her business process owners.

Christine's emphasis on Minimum Viable Product serves her well in the weird and amorphous world of web development.

KL (40:26): I'm happy to bring back Richard Wyatt, who was featured in last month's episode (#78. Project Perspective) where he spoke about innovation and project management. Richard has extensive experience partnering with business and IT leaders to deliver change. Most recently, he worked as head of digital transformation program management for TIAA, where he managed Agile transformation and digital process transformation. Here, his topic is the role of project managers in an Agile world, and I want to alert you, this is not the old story. It seems our roles continue to morph, and that leads to some interesting outcomes when faced with an Agile approach.

KL: This presentation you gave was *The Project Manager is Dead, Long Live Project Management: Project Management in the time of Agile*, and you have a great slide in there about where are all the PMs? We've been talking Agile for 20 years now, so, how is this so important? What's going on here?

Ricahrd Wyatt (41:27): Well, the reason I did it, and I talk about it is, I was at one time ask to come back and manage all of the project managers in the organization, and we have a couple of hundred. The idea was that we can help train, have project managers report to project managers who can help develop them. That was the idea. And when I started to do that and figure out how we'd pull these resources together, all of a sudden, not a surprise, but the manager, the current managers of those project managers, would say, "Well, he doesn't really do project management, he really does this." And all of a sudden there's 200 project managers, all of a sudden evaporated.

KL: I remember people being intrigued by the lens, looked through the other direction, meaning, Wow, all these people have these job titles, but you know what they're really doing? Project management, right? It was this idea that project management's happening everywhere, and we just haven't put a rope around it, hadn't put a title to it, hadn't put functions around, and that was the role in fact, in part of PMBOK®, was to create that body of knowledge and the guide to it, and the ability for people to see it and have a profession. And so now we're discovering, Well, where are all the project managers? It strikes me that it's the same conversation, though, isn't it? That project management is just inherently embedded in a lot of getting work done, because things are projects, meaning they end. Many many things that aren't operations or programs are projects.

RW: Absolutely, but I really think that the bigger part of the title is the *Manager* piece, which is you're really managing something. You're managing the delivery of something, and traditionally, project management's very much been, you know, predefine it and I'll deliver it at a certain time. That's the predictability piece. But really you're managing something. And I've seen all sorts of people who are really, really good at delivering, who aren't necessarily fully trained in the art of project management. And I think in many ways, to me, it has kind of gone circle.

RW (43:36): Business leaders want certainty. So business leaders are held to account. Shareholders are held to account by Boards of Directors. They're held to account on many different ways by the stock market to deliver the results that they say. And business leaders currently strive throughout a given period, a given year, to make sure that they hit the results that they've been charged with hitting, and they want certainty through that year that they're going to hit those.

And the thing about project managers, and why they love us...mostly...is that project managers

provide predictability. They want that job on time, on budget, a set amount of functionality and quality, so that's what we provide. We provide the predictability, which, to our aspect of the business, helps provide the leaders with the certainty that they crave.

Project managers, obviously do this in many ways, with a bunch of skills and competencies that project managers have. We have a number of tools and techniques that help us do that. It's basically about structuring everything, so you know when things are going off course. That's what we all do.

Project management was ideal, was created...was ideal when we were very Waterfall, we were very, again, predictable. At a time when the time horizons in an organization were longer, the markets weren't moving as fast, and it sort of grew up around that area.

KL: I think your logic was lovely...was actually elegant here. Because it seems like...when I first looked at, I was like, Okay, so here are some bullets. Agile's good, PMs are predictable, whatever. But then I see the logic in it, as you're describing it. PMs are really an extension of the business's need for predictability, right? In the sense that you just framed it as delivery, right? So now the strategy is there, the executives are there, but now we need to get things done, and PMs are like in that channel of, Bring me...I talk about it as order out of chaos. Bring me some predictability on these new things we're doing. But then you highlight, Well, and the world though, tends to increasing rate of change, which we've talked about for 40 years, right? Change is happening and the rate change is happening, and we have acceleration.

RW (45:58): The rate of change is increasing, we all know, we've gone digital, there's a whole bunch of good examples around how the rate of change is impacting us. It's changing the way we work on projects. So we've got customer needs that are changing rapidly, we've got the need to get to market, get products out, you know, it is dramatically increased. We've got new competitors coming in and a good example, something I saw the other day, where the world's biggest watch maker is no longer Swiss, it's Apple. So millions of examples about how new competitors are just jumping into our space, whatever space that you're in. Business leaders want faster and faster return on investment and get their money back more quickly, and want to be able to hit their numbers, and it's more and more uncertainty. We don't know, we experience it in project management, this continual change, change in requirements, changing needs. Particularly on long projects, they find that things have changed before you've even got halfway through building what you thought you had to do.

RW (47:11): Because the world is changing so fast, we kind of get into what would be Agile...organizations need to be Agile. And I always refer back to the Agile Manifesto, and I always ask the folks that I work with, and the folks that I manage, to always think and have the Agile Manifesto in mind. Because it kind of aids this ability to move quickly, and gets us away from what I call contract...management by contract. And I think in many ways, project requirements are a contract. You are saying that you will deliver this set of requirements for a given dollar figure. And what Agile Manifesto says is really, sort of, get away from that pre-defined contractual relationship. So the question is, where will all the project managers go?

KL (48:07): You show a nice diagram in there, and it reminded me of being in high school math where, how do you know the length of a curve? And it was like by taking small chunks of it, in cords, technically, is one way to do it, before you get to the calculus level. You're finding these small little chunks. And that's what agile can do. Agility itself is a manner of providing smaller

chunks, which allows you, as an organization, inherently to be more adaptive to change. So it wasn't just that it's a different development methodology to produce better products more easily, more quickly. It's that it allows an organization to be able to respond to its fundamental market or the constraints it sees in the market, was what I began to take out of this suddenly, and I was like, Got it! We don't need PMs to be so much the agents of predictability, but the agents of delivery in a world where delivery changes rapidly. So that's a squeeze on your PM, if they're viewed as predictable, agents of predictability.

RW: Yeah, that's exactly right. And I think what a lot of the methodologies do as I articulate, is to get that very small cord, and then say, Well, just to do that small piece, I don't need that skill set anymore. But again, it's this underlying, behind that. What else happens? And I think a lot of the change that goes on, and it's this fundamental changing underneath.

You can do Scrum, you can do Agile. You can have a backlog. But it doesn't mean that behind all that, you're necessarily totally changing the way you view things, and how you adapt, and how when you move on. And I think to make that work, sometimes, you still need that skill set to deliver... It's not just two weeks have elapsed, and I'll do these things off the backlog. You have to deliver real stuff in the real world, and I think that skill set... There's still conflict, there's still resource constraint, and there's still all these things that go on in the background that still need to be managed.

KL: Yeah, because I think almost, in a way, this is the punch line to me, and this one was confusing initially. Individual sprints don't need project management, but in aggregate, they must exist in an organizational environment that likes certainty.

RW (50:31): If you've got a two week body of work, with ten people, it doesn't need a project manager. And Agile doesn't allow...it discourages having project managers. There are project managers in many methodologies, they exist in some form in others. But typically you have a Scrum master, helping people stay on the methodology, but you don't need somebody making sure that... That you hit your two-week deliverables. The process kind of does that. It's not that big. It's ten people for two weeks. It's a tiny amount of work.

But then, sort of reality hits, right? So yes, you don't need a project manager for a two-week 10-person project. But then what happens? Okay, so then you start to roll up all these very small, little entities that in themselves don't need project managers. But now all of a sudden, let's just say, you've got 10 or 12 Agile teams, all working together. So now all of a sudden, you've got 125 people, and now you're actually delivering... you're delivering a bigger body of work, let's just say 12 months, even though it's a set number of people, you've really got 125 people working for a year, so you've got a 20 million development project.

Well, all of a sudden, when you roll up Agile, and view it in that way, you've actually now got quite a big project. And what we find is that the work is still there. You've still got compliance, you've still got infrastructures. So you still have all the negotiation, and all the issues that arise. You still have other executives in the firm re-prioritizing what they want their group to do, and does that have an impact on, maybe finance? Does it take away some finance? What you do? It's great to say, in Agile, we're going to fund these ten teams on a consistent basis. But what happens if someone changes the rules? You still need that more senior, that project management skill set, to interface and manage those relationships. They don't just happen. So if you take away project management

down at the unit level, yeah, sure, that makes sense. But once you roll all these things back together again, there is still a need for those capabilities that project managers have.

KL (53:05): So, tell me about what should be done then. Because if sprints don't need the project management, as we said, because that's predictability in a space where it's not needed, and we recognize that organizational environments do like certainty, right? You're insisting that they do, in aggregate. So what's to be done?

RW: Yeah, so the methodology may just say, Well, you have a release train engineer who coordinates it. But there's no human on the planet, individual, who can coordinate these ten-plus teams, plus manage all the conflict and coordination with all the other pieces of the organization. And until such time as an organization has totally changed, and doesn't have all these conflicts and whatever, you're always going to need that level of skill, that resource that can manage a lot of these things.

So to me, it's like I always...Imagine the clutch plate of a manual car, right? So the engine is...Let's say the engine is the Agile teams, knocking out software for a backlog. You know, just taking and delivering, bang bang bang. And the wheels on the ground is the rest of the organization that still has certain other constraints, that still has a culture, still has whatever, historically, that's made the organization what it is. In the car, though, you have a clutch to manage the different rates of change in spin and whatever. And when I look at this, I kind of see this role, this area, really, is the clutch plates of an organization. So how do I let the engine run at its most efficient in a point on the power curve and equate that with making the organization, which is different, unless you reach the Nirvana of being a perfectly Agile organization, how do you manage the frictions and the general workings to turn this into a functioning, profitable, or whatever, business? So that's how I can see it.

RW (55:20): When I look at project managers, the good project managers, they're able to foresee where problems are likely to occur. They're able to get in front of those. They know the organization, they know the reputations of different groups, partners, within, without an organization. They know who's good at doing what they say, and who lets you down 'til the last minute, and know those things. They're able to negotiate well, if there's a conflict, and they can't get a specialist or resource that they need. They have the skill set to know how to make things happen. Someone you can count on, that will deliver what you want. I don't care whether it's in a two-week sprint or a nine-month project, or any aspect of the business. You could be going through any business activity, you want people who will have it done, at the time you said it will be done.

KL (56:23): Long live project managers! It's clear, there's still an important role for PMs, and the skill set we bring to the table. The keyword, as Richard says, is *manager*. It's maybe not so much delivery, but coordinating and negotiating for multiple teams who are charged with the actual delivery.

So Agile is all about transparency, teamwork and communication. And all very doable, whether you're working in a virtual environment or not, and requiring the skill set specific to a project manager. It's all there.

Special thanks to my guests, JJ Sutherland, Christine Brennan Schmidt and Richard Wyatt. Richard can be reached at rwattoz@comcast.net. Christine Brennan Schmidt is at cschmidt@acs.org. And you can find JJ Sutherland at scruminc.com, LinkedIn or Twitter @jjsutherland.

Announcer: Our theme music was composed by Molly Flannery, used with permission. Additional original music by Gary Fieldman, Rich Greenblatt, Lionel Lyles and Hiroaki Honshuku. Post-production performed at M Powered Strategies.

KL: PMs who have listened to this complete podcast may submit a PDU claim, one PDU in the talent triangle “Strategic” with the Project Management Institute’s CCR system. Use provider code 4634, and the title PMPOV0079 Agile Today from the 2020 UMD Symposium. You can also use the PDU claim code 4634ASFWPF.

Tune in to our next episode, where we will feature executives talking about what they need, and what they expect from project managers. Visit our Facebook page, PM Point of View® to comment, to listen to more episodes and to get transcripts. Leave comments on the projectmanagement.com portal, evaluate us on iTunes, and of course, you may contact me directly on LinkedIn.

I'm your host, Kendal Lott, and until next time, keep it in scope and get it done.

Announcer: This has been a Final Milestone Production sponsored by M Powered Strategies.