



Distributed Agile Team Kickstart



Table of Contents

Agile With Distributed Teams	4
How important are Tools in (Distributed) Agile Teams?.....	8
Where to go next?.....	11

All the articles in this document have been originally published in Velocity Partners [Blog](#)

Velocity Partners is committed to continuously improving our agile practices and those of our clients. We regularly add posts to our blog (at <http://www.velocitypartners.net/blog>) to contribute back to the community at large. We hope that by sharing our experiences and expertise we can help individuals and teams on their own agile journey.

Distributed teams are the way we do business today. Having some guidelines and advice on managing those distributed teams can be valuable to those new to working with distributed teams. Even those with experience working with distributed teams can still find some valuable tips and tricks in these posts. I've been working with distributed teams for almost 20 years and still find new ways to collaborate and communicate.

Thank you for downloading this guide and I hope it helps you and your teams succeed.

Bill DeVoe

Principal Agile Evangelist

Velocity Partners

Agile with Distributed Teams



Article written by Bill DeVoe, an Agile Methodologist with over 25 years of experience and almost 20 years of that in agile. Bill's role at Velocity Partners encompasses content creation, training, coaching, and mentoring VP's delivery teams. He also speaks at various conferences and works with clients on their agile practices. Bill is a Certified Scrum Professional, a SAFe Program Consultant, and licensed Management 3.0 Facilitator.

Whenever I work with companies, the question of using agile with distributed teams comes up. So much so that I presented on that topic at Mile High Agile in 2016. Part of the reason is that one of the principles in the Agile Manifesto states:

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

While we tend to interpret that as meaning that physically co-located teams are best, some people take it to mean that you can't do agile with a distributed team. Mostly those people are the ones who don't want to do agile at all, so they tend to see this as an "out" of adopting agile. But agile can – and does – work with distributed teams.

Some Thoughts Around Distributed Teams



Distributed Teams can be global

There are many different types of distributed teams. And there are many different reasons why those teams can't be co-located sometimes. Here are some examples:

- Team members are all in the same building but spread out. Facilities (or other “powers that be”) have determined that it would be too expensive to move the team members together.
- Team members are on the same campus, but in different buildings.
- Team members are remote workers and there is no building. WordPress is a great example of this, per Scott Berkun's book [The Year Without Pants](#).
- Team members are distributed across the US (or another country)
- Team members are distributed across the globe (gack!)

In all of these cases, there is a good guideline to follow: if one person is remote, everyone behaves as if everyone is remote. This prevents those remote people from getting caught unawares of things happening on the team or in the company and ensures that everyone is connecting in the same way. It helps level-set the team.

So what do you do if your team has remote members or is totally remote?

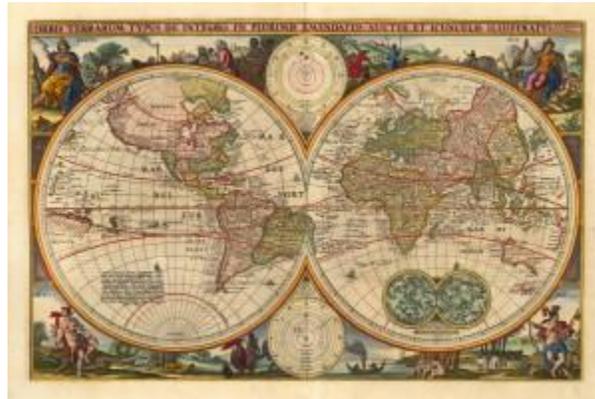
The Easy Cases

If everyone is in the same general area, the team has to deal with the communication challenges more so than the scheduling ones. I've seen several things that are very effective in helping make remote workers feel like part of the team:

- Video conferencing. This can be via Skype, Google Hangouts, Lync, whatever, but it's very helpful if everyone uses a video camera. Imagine you're holding a meeting and no one is responding. Is it because they're checked out? Because they're muted? Without a camera, you can't tell and face-to-face communication is the most effective form of conveying information. Show those faces!
- Participate as if you were all together. If you're in a stand-up/Scrum – stand up! Yeah, it might look a little weird if you're just standing at your desk, but it does help with the intent (keeping it short and focused) and with creating a sense of team.
- Open each ceremony with a personal ice breaker. Have everyone share something about themselves every time you get together. This would normally happen in off-hand conversations in the hallway or over lunch. With remote teams, you need to force this a little bit.
- Staying connected. Be online whenever you're at your desk. It's critical that people be able to reach you (when appropriate). Using Slack, HipChat, or some other communication tool that logs conversations can be really useful.
- Do everything electronically. Store your documents and backlogs in an online tool (Jira, Rally, VersionOne, etc). And use them when you do planning. Don't have something that only one person or a few people can access. Automatically update your Slack or HipChat channel with build messages or errors. Use technologies to help your teams stay connected.

- Encourage collaboration. Use screen sharing applications (again – Skype, Google Hangouts, Lync, etc). This helps with knowledge sharing but also helps the team feel like a team.

The Hard(er) Cases



Antique World Map

The Hard(er) cases are those where your team members are spread across different time zones so there's little or no overlap. This means people coming in very early or very late. There are also a lot of other concerns that must be considered:

- Transportation availability for coming in very early or staying very late
- Holidays for foreign countries
- Technology/infrastructure
- Cultural differences

None of these are insurmountable – far from it! – but they do add complexity to the conversation. When I worked with teams off-shore in India for a major health care client, we had to move our planning sessions to very early in the morning and around noon local time the Indian teams had to leave to make the last buses of the night. That really impacted the effectiveness of the planning because they missed some of the conversations that happened after they left (we still had several hours of work) and they had to catch up the next day. Additionally, they missed the final plan review and acceptance.

In general...

In all cases, though, encourage the team to engage with each other as much as possible. Perhaps using some form of “merit money” (although it doesn't need to be financial) can help the team collaborate. Provide a small reward – a candy bar and a small toy, for instance – to the person who earned the most points in the previous sprint. Or correctly answered the most questions about their coworkers. Be creative!

@Velocity

Because Velocity Partners has teams that are aligned with the US time zones, our teams tend to fall into the “Easy Cases” category. That’s a major benefit to working with a near-shore partner. Our teams go from Eastern time plus one hour (in Buenos Aires and Montevideo) to being aligned with Central time (in Medellin). That helps create a lot of overlap on schedule with our clients and helps facilitate collaboration.

One other benefit is that with our teams in South America, there is a very similar culture. There is far more overlap than difference – as I can say from personal experience. And the teams are very used to working with US clients, and that familiarity aids collaboration dramatically.

Closing Thoughts

Distributed teams are the way many companies work in the 21st century. As a result, we need to adapt how we adopt agile for those teams. And while it can be a challenge, distributed teams can be just as effective and productive as teams that are co-located. I don’t recommend giving up the fight for co-location – it has **many** other benefits – but don’t let it be the only reason your teams can’t be agile.

Feliz entrenamiento, mis amigos! (Happy coaching, my friends)

PS – There is a fabulous podcast over at [Collaboration Superpowers](#). Check it out for even more ideas!

How important are Tools in (Distributed) Agile Teams?



Article written by Bob Galen, an Agile Methodologist, Practitioner & Coach based in Cary, NC. In this role he helps guide companies and teams in their pragmatic adoption and organizational shift towards Scrum and other agile methodologies and practices. Contact: bob@rgalen.com

You've all heard me say this before, but that won't stop me from saying it again. I teach and share on agile methods at a large number of events each year. There are some common questions that I inevitably receive over and over again.

One of them relates to tooling and distributed agile teams. It seems as if everyone is searching for a "Silver Bullet" tool that will magically make agile work incredibly well for his or her distributed teams. Far too often the question relates to a specific tool, for example, "Bob, should we use Rally or VersionOne for our teams"? Tell us which one to select.

I usually use this as an entry point for a larger and broader discussion. That is – are tools the number one concern in initiating a new set of distributed agile teams? The usual answer is yes. In fact, it's often the case that tooling is the first thing that organizations focus on when they're transitioning to agile and I think that's the wrong approach.

I'd much rather they focus on teams, organizational training, and getting a healthy, low-fidelity start than immediately trying to buy their agile implementation.

So, what's wrong with tools?

The short answer is nothing.

Have you even been dumped by a girl or boy friend and the reason was – "It's not you...it's me"? Well, it's the same answer with Agile Lifecycle Management (ALM) tooling. It's not the tools – it's us.

Many of us are looking for tools to help us solve really complex problems. We put the tooling before the teaming or the people. If you understand the underpinnings of agility, you realize that view is backwards. Tools never solve your hard problems...people do.

Let me illustrate my point with a story

I joined a company a few years ago as an Agile Coach and Scrum Master. They were quite new to agile. They had installed Rally's agile tool and asked Rally to come in for a bit of training and coaching. So, they did just that - installed the tool, got things started, and went away. The organization at this point was on their own in their agile journey.

Not necessarily bad and they had done a nice job of what they'd been contracted to provide.

I joined about 3 months later. As the "Scrum Master" I also inherited administration responsibilities for Rally. I'd used it at several previous jobs/clients, so I was fairly familiar with the tool. But I'd never been the sole administrator and was looking forward to gaining more detailed knowledge.

Around my 4-5th day on the job, one of the developers (Mike) came up to me really anxious about something. He said - "Bob, I need a Rally trigger configured right away to remind me to talk to Sally whenever I work on a backend stories of this specific type. Sally is expert in that area, and I feel her review and approval would be a good thing to get".

Not knowing any better, I could only agree. But when I looked to the right of Mike, I noticed a nametag on the seating space right next to him. It read - Sally Smith.

I asked Mike - "is this the Sally you want the tool to tell you to collaborate with"? I mean Mike; she's sitting right beside you.

Mike said - "Yes"!

You see, the entire organization blindly followed the to-do and working list produced by the tooling. Beyond collaborating at the daily standup for 15 minutes, there was simply no other collaboration...unless Rally reinforced it. I'm exaggerating a little bit, but not by much.

Guess what I did? So did I give Mike his "collaboration nudge"? No.

Instead, I pulled the plug on Rally. I felt that the team wasn't mature enough to leverage the tool AND demonstrate the collaborative behaviors of solid agile teams. So I decided to "retire it" for a while and go back to very low fidelity tools like whiteboards, flipcharts, and post-it notes.

I told the team that we would turn Rally back on after they had gained more experience and maturity in their agile journey. That in other words, they had to EARN the tools.

Maturity matters

I think the moral of my story is that tools in and of themselves are not bad. But, I truly believe agility is best learned with:

- The simplest possible tooling, paper and electronic, that can support your agile needs;
- Tools that don't get in the way of team collaboration, but rather reinforce and enhance it;
- Tools that the team influences based on their needs and NOT driven by management reporting/tracking needs.

The focus on new teams should be on face-to-face collaboration. And yes, even if the team is distributed in some fashion.

Now this recommendation is for new teams. As teams mature, they can expand or change their usage of reliance on tooling. But they're doing it from a position of experience rather than as a start-up silver bullet.

Finally, a Commitment to Agile Principles

This is another conversation I often have in conferences. I often get challenged that agile doesn't work in distributed team contexts. That if team members are 12 time zones apart, it is virtually impossible to collaborate.

I try to remind them that this is a choice teams make. If the teams are committed to agile principles, then they will find a way to support the collaboration and teamwork aspects of agile even if it's inconvenient because of the time or any other obstacles.

For example, they need to commit to the daily stand-up as a communications and collaboration synch-point as a team. That they're not doing it to "be agile", but they're doing it because they've made a collaborative commitment to their team and to a new way of working. Even when it's challenging because of time zones!

I also encourage them to get creative in how they collaborate. For example, rotating meeting times so that everyone gets a fair chance to get up early and using simple webcams so that they simulate face-to-face conversations. I'm always surprised at the solutions that teams come, to ease the challenges, when they commit to the principles.

Wrapping Up

Is implementing agile in distributed contexts harder than with co-located teams? YES!

Are tools incredibly useful in distributed team contexts? YES!

But don't let the fact that your distributed OR the tools you use move you away from operating solid, fundamental principles within your agile teams. And occasionally "dust off" a Post-it Note or two.

Stay agile my friends,

Bob.

Where to go next?

You can visit our website at **Velocity Partners** and read more articles in our blog [here](#).

Velocity Partners is a nearshore software development company that strategically aligns with companies to deliver results better, faster and more cost effectively. It does this by leveraging a distributed Agile model. Based in Seattle, **Velocity Partners** operates software delivery centers in Argentina, Uruguay, Colombia and Venezuela where it has found the culture, talent and business mindset to exceed its standards for success.

Copyright © 2017 Velocity Partners