Inspecting Distributed Team Practices

Tamara Sulaiman

May 12, 2008

Geographically distributed teams face additional challenges when transitioning to agile development practices. For example, if face-to-face interaction is the best method of communication, what's the best way to communicate when you are in different locations? Are agile frameworks like Scrum strong and flexible enough to support the distribution? How often does communication between business and the development team need to take place? Is more process necessary with distributed teams? If so, how much more process is just enough? What about tooling? Where do Post-it notes and tooling solutions meet in a distributed environment? How agile do we need to be, given our distributed team situation?

Don't give up hope…there are distributed teams that are successfully implementing agile practices. Yes, some practices need to be adjusted to the reality of the situation with distributed teams. So how do you know where to adapt and where to leave well enough alone? I generally refer back to the values and principles expressed in the Agile Manifesto.

Agile practices all have the manifesto in common. I think of the values expressed in the manifesto as the foundation of the "House of Agile" and the 12 agile principles as the pillars supporting the house. The four values of the manifesto state that we value:

- People and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more

As I am working with distributed teams adapting agile practices to their particular situation and environment, I use these values as one guide to keep us from straying off the agile path. Here are some of my key recommendations for organizations with distributed teams who are looking to make the transition to agile software development.

Recommendation: Implement Scrum as a project management framework--and stay true to the "Rule of Three".
How much process is enough to facilitate communication and collaboration without becoming a drag on productivity? This is a central question that organizations with distributed agile teams have to answer. In my experience, the Scrum framework is flexible enough to be implemented across widely distributed teams. The "Rule of Three" is something I teach in my Certified ScrumMaster classes. It is an easy mnemonic to describe the basic Scrum practices and provides a minimum structure that, in my opinion, must be implemented for Scrum to be effective:

- There are three roles in the Scrum framework: the Product Owner, the ScrumMaster and the Delivery Team.
- There are three deliverables or artifacts created through the Scrum framework: the Product Backlog, the Sprint Backlog and a potentially shippable product increment (i.e. working software.)
- There are three meetings in the Scrum framework: the Sprint planning meeting, the Daily Scrum (stand up meeting) and the Sprint Demo and Retrospective. (Okay, I know that last one is really two separate meetings. However, they are often held back to back, so work with me here…)

How much communication is enough? By keeping to the rule of three, you have just enough structure to keep those lines of communication open. I am not advocating that no communication outside the three meetings is necessary; I'm saying that this is the minimum amount that you have to have. For distributed teams, it's the difficulty with communication that causes the most problems.

Recommendation: Schedule meetings in such a way as to share the pain.
In cases where multiple time zones are involved, it is important that all sides take turns in giving up convenience when scheduling planning and other collaborative sessions. If teams are 8, 10 or 12 hours apart, someone will be working late (or very early). I recommend scheduling sessions in such a way that each team has the chance to be collaborating during their normal working hours, or as close as possible. It can be very disheartening to be the team that always has to work very late for planning sessions for the convenience of another team.

Recommendation: Leverage technology by establishing a global communication infrastructure, including team and project wiki's, shared sites, video and web conferencing, instant messaging, digital whiteboards and other useful tools.
Valuing people and interaction over processes and tools doesn't mean no tools. In all cases of distributed teams, leveraging technology is a critical success factor for successful collaboration. I have worked with teams leveraging technology in a variety of ways. The most simple is a conference call for the daily meeting. For longer planning sessions, conference calls leave a lot to be desired. Linking video technology with conference calls works much better, allowing parties to view the person that is speaking. Digital whiteboards and web conferencing technology brings additional sophistication to the communication and collaboration. Across iterations, Web-based backlog management tools are invaluable to distributed teams. Project wiki's and other collaboration technology

add value to team member's efforts. On one project, the wiki was used to post technical and requirement questions between remote teams. Answers were input, discussed and saved on the wiki. IM, Skype and other communication tools improve the ease of communication between team members.

Recommendation: Train everyone on the teams in the process and practices.
At times, remote teams get left out of this equation. An agile transition across dispersed teams has a much better chance of success if you are able to arrange training for everyone. If possible, have an onsite mentor available to teams for coaching and training. If that isn't possible, arrange for all teams members to have a basic level of training in the chosen practices so you are all using the same terminology with the same understanding. Yes, it costs money--but so does failed software. Where would you rather invest your effort and dollars? (Or rupees, dinars, euro's...)

Recommendation: Create cross functional, self organizing teams in each location.
One of the agile principles states that most effective method of communicating to and within a team is face-to-face conversations. Unfortunately, this is not always possible with teams in multiple locations. In order to reduce complexity, the best way to leverage multiple teams across locations is to create cross functional teams in each location, as much as possible. Rather than dividing teams by activity--for example, the dev team in the Romania and the test team in Mumbai--work toward having both dev and test skill sets on teams in Mumbai and Romania. If possible, have the remote teams co-located. Carefully consider the completely dispersed team where team members all work from home. There is a productivity tradeoff involved there. Is it truly cost effective?

At times, resources with scarce skill sets will have to be shared across locations (UI designers, DBAs and architects are a few of the most common examples that I've seen.) Carefully consider how that can best be accomplished.

Dividing single teams across multiple locations is rarely a good idea. Have each team work as independently as possible, reducing the critical number of lines of communication, and decoupling the work allocated to each site. It is important that each and every team has a ScrumMaster either on site or very nearby. I've seen organizations divide teams across sites, with the ScrumMaster on a different continent. That combination usually brings added complexity to communications and slows things down.

Recommendation: Bring team members together at key events.
Project kickoff and release planning are good examples of these types of events. By collaborating in person at these events, working relationships develop that strengthen the future distance collaboration. Even having a handful of team members physically present for these events--while remaining team members collaborate remotely--can add enormous value in sharing knowledge. Sensitive issues can be discussed and resolved while project team members spend valuable face time together. It can be expensive, yet that expense is more than offset by the shared knowledge base, improved working relationships and joint vision for the release--all of which contribute to increased productivity and quality.

Recommendation: Have product owners, subject matter experts and other crucial team members visit distant teams on regular basis.
The results of these visits is that collaboration across teams remains high, team members remain tuned in to the project vision and the communication impediments related to cultural norms fade into the background as cross team members relate on a personal basis. Visitors can better understand the issues and impediments experienced on location, and original ideas can be easily seeded amongst the other related teams by these visitors.

In general, working with distributed agile teams means that product owners, project manager types and other subject matter experts have to spread themselves around--maintaining the big picture view, constantly communicating any changes to the release, and in general working to integrate these dispersed teams into an integrated whole.

These are not the only recommendations for easing the pain of transitioning distributed teams to agile practices, but they are a start. Other recommendations center around having a global development infrastructure that acts as a single development environment, working to bring test forward so that each team is truly delivering tested code that is integrated and tested within the code base, and developing a program of standards, guidelines, training and templates as shared parameters for each self-organizing team to work within.

Geographically dispersed teams are often a fact of life. Our challenge is to develop and implement agile practices that stay true to the values and principles in the manifesto while remaining practical and pragmatic solutions to the unique challenges involved.

The URL for this article is:
https://www.projectmanagement.com/articles/242543/Inspecting-Distributed-Team-Practices