Tips for Agile Distributed Teams

Agile Six Applications, Inc.
Robert Rasmussen, CEO

Introduction

There are a myriad of benefits to be offered by unleashing a team from the constraints of geography. The distributed team is one in which members of the team cross time zones and regularly work without physical access to each other. Benefits of team distribution are many and include the ability to leverage the best talent regardless of where it resides, cover multiple time zones and facilitate the team members' individual lifestyles and residence preferences.

Here at Agile Six Applications, we consider ourselves a distributed or "Virtual Company." We chose this configuration because we believe

- It's best for our clients. For example, it reduces overhead costs, increases the availability of human resources, and increases the size of the talent base that can be leveraged.
- It's best for our employees. For example, it keeps them out of traffic, maximizes productive time, and helps them better integrate work alongside other priorities.
- It's best for the environment. For example, eliminating or reducing commute times reduces traffic and carbon emissions.

We also believe strongly that distributed teams thrive best in distributed companies. While we keep a physical address and we can certainly host clients in our San Diego office, if you wish to visit our "headquarters", we are happy to send you a Webex link!

Having managed large internationally distributed teams, the author of this paper brings over 10 years of distributed team experience. The goal of this paper is to share some of that experience and improve both the product of, and the experience of the distributed work force. We are extremely aware of the challenges of distributed teams as well and so we will also address those. The tips included in this paper are not all based on our experience, we read a lot and we freely credit many of them to the articles in the resource section at the bottom of this paper.

Agile and Distribution

We practice Agile Scrum and Scaled Agile Framework (SAFe) in our projects and we find that they are both particularly well suited for distribution. "Ongoing collaboration is one of main principles of Scrum, and a rich communication infrastructure is especially important in Scrum "ceremonies" (i.e., Daily Standup, Sprint Review, Sprint Retrospective, Spring Planning, etc.). Given the importance of maintaining a collaborative environment, consider the following tips for distributed teams:"

- Create a set of *Team Rules/Contract*, with the participation of the entire team; create a basic set of rules that govern the behavior and manners for civil and effective communication. Some of the proceeding tips should be great input for those rules.

- Invest in effective *communication tools* and make sure the team (if not the company) commits to the same toolset. Some tools that we have used and recommend include Slack, Hangouts, Webex, and Skype.
- Invest in effective *work management tools* and make sure the team (if not the company) commits to the same toolset. Some tools that we have used and recommend include Jira, Trello and TFS.
- *Synchronize your watches*: This is a lesson we learned in the military. Pick one time zone and equip your team to live by that. In the military we used GMT, in the contractor world it's often EST (DC Time), but agree to one and stick to it.
- Adopt a *"remote first"* mindset, meaning that we assume all team members are remote, all meetings have remote access and team members do not generally assemble in meeting rooms. If teams do assemble in meeting rooms, they need to avoid local chatter that can ruin the remote members experience.
- Build a *web-based training program*. We believe strongly that all team members should be provided with process training (in our case Agile Scrum) for their positions. This is not limited to Scrum Masters, Product Owners or SAFe roles, but individual team members also need to understand the foundations of the process. This is even more critical in distributed teams as conflict management is complicated and it's exponentially more difficult to settle disputes. We recommend web-based training because of our "remote-first" posture and so that members can attend together. We find that local scrum coaches can provide divergent advice.
- *Budget for travel*. At the end of the day, some things require face to face interaction. Some of these (e.g. relationship building) pay dividends in team effectiveness that we will not try to defend in this paper. Additionally, some ceremonies (e.g. Product Increment Training and Risk Management workshops) simply will not be effective without colocation. We recommend that distributed teams invest some portion of the facility savings back into travel requirements.
- *Embrace Asynchronous mode*. While we ask our teams to compromise on core working hours (e.g. 11:00-14:00 EST) to facilitate ceremonies, we also embrace the idea that they can chose their working hours otherwise. These core hours are used for synchronous communication and non-core hours can mostly be dedicated to undisturbed efficiency. We find this to be particularly advantageous to developers. We also find that developing and following some basic team rules that support asynchronous work – such as identifying smaller tasks, moving task cards when they are completed, verifying that checked-in code does not break the build or block others, and asking each team member to reach out to the team before signing off for the day to make sure they are not blocking any task – will result in shorter "queue lengths" and greater overall team efficiency.
- *Embrace architecture best-practices that promote standardization, modularity and loose-coupling*. While such architecture best practices were certainly not conceived to support distributed teams, we have found that following them facilitates distributed development. For example, if an architecture includes components with well-defined responsibilities separated by well-defined interfaces, then each component can be worked on in relative independence once the interfaces and basic collaboration schemes are worked out. We believe that,

in general, Agile is propelling the industry towards micro-service based architectures and away from monolithic architectures, and it just so happens that this trend facilitates distributed development as well.

- *In general, favor the richest communication mechanism that is easily available.* We understand that much communication – in fact often the most important communication – is non-verbal; hence we encourage remote teams to use whatever tool is at their disposal that provides the richest communication context. In other words, if choosing between a phone conversation and a video conference, choose the video conference if it can be used by all participants (and ask all participants to turn their cameras on). Favor phone conversations over chat for non-trivial communications, etc., etc.
- *Hire a workforce that can work without supervision.* Agile itself puts a certain amount of emphasis on self-direction. Distributed Agile teams makes self-directedness even more important. We've found that most associates fall into three broad categories with regard to their ability to function on a distributed team:
    - Those that naturally embrace a distributed, independent work-style and do not need a lot of oversight. These associates instinctively figure out how to contribute and just do it.
    - Those that may not "get it" at first, but after a few months find their way to becoming productive distributed team members.
    - Those that, for one reason or another, never really become productive on distributed teams.

  It's amazing to us how long teams and organizations enable and tolerate associates who fall into the third category on distributed teams. Associates who aren't productive on distributed teams not only represent an opportunity cost for their place on the team, but may reduce the effectiveness of others on the team – through their impact on morale and through their constant drain on others in needing to be directed. Teams who can quickly identify associates that fall into the third category, and remove them from distributed teams as soon as it is clear that they aren't a good fit for this work-style, will maximize the effectiveness of their distributed teams.
- *Focus on Results*. As we like to say "around here", "effort is not important, results matter". We track hours only where required by our contracts. Otherwise, we expect team members to govern their own hours and teams to govern the individual contributions. We find that people are hardest on themselves and teams are hardest on their peers.

Advice from 18F

Here at Agile Six Applications, we are very fond of the US Digital Services and the movement that these innovative young entrepreneurs are brining to Government. The General Services Administration has taken up that spirit with their "18F" division, which focuses on delivering the services outlined by the USDS (see playbook.cio.gov). This team has leveraged talent from all corners of our nation and for all the reasons we did (i.e. in order to attract this talent) they

embrace distribution. Please see the link in the reference section below (by Melody Kramer and Michelle Hertzfeld) to their advice on this topic. We find it particularly insightful and so we will summarize some of it below:

- **We have a "remote first" mindset.**
- **We have a five-hour overlap in our workdays.**
- **We share our screens frequently.**
- **We have face-to-face meetings at least once a week.**
- **We make our work transparent to each other.**
- **We over-communicate, especially with clients.**
- **We all think of ourselves as remote employees, even if we're in an office.**
- **Some of us shift our time zones**

Conclusion

Distributed teams offer both unique challenges and opportunities. They offer the Government a better Return on Investment (i.e. reduced facility costs, wider talent pool and time zone coverage). They offer potential employees more opportunities in their markets and employers more options for recruiting. They can contribute to more engaged and motivated teams. However, they are not without risk. The advice provided in this paper and the resources listed below, should prove valuable to any agency or company learning how to deliver better results from distributed teams.

Tools to consider

- **Trello** – Web based tasking tool.
- **Jira** – Bug, issue and project tracking
- **Google Hangouts** – free unified communications service that allows members to initiate and participate in text, voice or video chats, either one-on-one or in a group.
- **Webex** –Video conferencing, paid service
- **Switch**. Virtual phone system
- **UberConference**. A cloud based conferencing system from Switch.

Recommended Reading

http://www.forbes.com/sites/joshsteimle/2015/12/11/6-tips-for-managing-a-distributed-team/2/#460bd5df42d7
https://18f.gsa.gov/2015/10/15/best-practices-for-distributed-teams/
http://tech.co/distributed-teams-2013-08

About Agile Six Applications

Agile Six Applications, Inc. was established to serve those who have bravely served our country. We are passionate about our mission *to improve the lives of veterans and their families by delivering world-class software solutions*. Our collaborative and highly transparent Agile development process invites users and program representatives to participate in the development process, and results in better solutions, delivered more quickly, at a lower overall cost. Our firm was founded in 2014 by former executives from the federal and commercial space (i.e. DefenseWeb & Amazon) in direct response to the formation of the US Digital Services where "America's most capable problem solvers are striving to make critical services — like Healthcare, student loans, and Veterans' benefits — as simple as buying a book online". As such, we actively promote the tenets of the CIO Playbook:

Digital Service Plays [4]

1. Understand what people need
2. Address the whole experience, from start to finish
3. Make it simple and intuitive
4. Build the service using agile and iterative practices
5. Structure budgets and contracts to support delivery
6. Assign one leader and hold that person accountable
7. Bring in experienced teams
8. Choose a modern technology stack
9. Deploy in a flexible hosting environment
10. Automate testing and deployments
11. Manage security and privacy through reusable processes
12. Use data to drive decisions
13. Default to open

Please visit www.agile6.com for more information.

About Robert Rasmussen

Robert is a proud Navy Veteran who has delivered multi-million dollar programs in several countries in the fields of Telecommunications, IT and Enterprise Software. Robert is a certified Project Management Professional (PMP), Scrum Certified Professional (CSP), Certified Scrum Master (CSM) and Certified Scrum Product Owner (CSPO). Robert is President and CEO of Agile Six Applications, Inc.

Robert can be contacted at robert.rasmussen@agile6.com