

Guidelines for Team Formation



Leadership

Agile Teams are a solution to the problem of complex communication requirements. A team has people interacting regularly, building interpersonal relationships, developing the “Scrum Values” of courage, commitment, respect, open-ness and focus. The people of a team work together daily to solve joint problems and deliver value including definition, design, build, integration and sometimes release to customers. Over time a team goes through the Tuckman stages of forming, storming, norming and performing. A goal of the team is that it becomes self-managing and granted decision authority, with clarity on strategy and guardrails from leadership. Over time the team becomes the differentiator and product-based projects are brought to the team instead of the team formed around projects.

Small

By Invitation

Organize around Value

Clean Interfaces

Get Started

Small... If you have 60 people.. per the 7 +- 2 rule, this translates to 7-12 teams. Collaboration is always more effective with a smaller group. Skills needed may mandate a larger number of people be included. To foster innovation, diversity, predictability, personal growth, “T” shaped skills, and to build a high performing team, we recognize that collaboration is key - “equal talk” and “active listening” at every interaction. When people don’t have time to effectively collaborate, the old relay handoffs of waterfall typically set in. We also sacrifice innovation and the opportunity to learn.

We want to keep the people dedicated, not part time, not shared. Being on more than a single team adds to that person’s overhead, included in the planning and execution meetings of both. The only reason to not have dedicated people on each team is when the specialty skill is included in fewer people than there are teams. In this case, have them dedicated to a primary team and support other teams on as needed basis. Encourage pairing so the specialty skill is helping others learn and reduces other team’s dependency on them.

Conversely when we do collaborate and take a genuine interest in what other people are doing or saying and the whys behind it all, we are more likely empathize, help, pair and learn from them, the "Flow" of work becomes faster and faster, feeding the learning loop to become faster yet again. A small self-managing dedicated team with a mission... the foundation is in place to satisfy all three intrinsic motivators of autonomy, mastery, and purpose.

Invitation to the three roles of Scrum (not their HR Title) – Invite people to join the new teams so they are well matched to the ambiguity to operate with a new system. Geoffrey Moore, crossing the Chasm identifies early adopters of any new technology are people willing to cope with Ambiguity.

Building the right thing, Building it right, and Building it fast are important perspectives to keep in balance while the team is self-managing and making decisions. Decentralized decision authority to these roles allow the Product Owner to determine "what" and the Developers to determine "how" and "how much." The Scrum Master identifies and removes impediments to flow, ensuring continuous improvement and celebrating the pillars and values of Scrum as the team experiments with new approaches.

Organize around value and focus on flow... the best team makeup is one that has all the skills needed to prioritize, define, build, integrate, deploy, and release their work so they can operate completely independent from any impediment that would slow them down. Ideally they have a mission that is aligned with their customer mission, perhaps aligned with their customer's success measures. Sometimes a team will share backlog with other teams who all are aligned around value and may share some higher-level intake and release teams. When there are multiple teams working from the same intake process, there may be special skills across teams. Again to balance any load (prioritization) that may be in the intake for all the team of teams, the nirvana state is that all teams are completely fungible across each other and can execute on the highest value backlog item sequentially (initially with cross dependencies, over time, learn to fill gaps). Separation of maintenance teams with new development teams is another antipattern as the loads on the different missions will be cyclical, thus neither can be long standing. A way to address this challenge is to agree on quarterly boundaries what % of capacity will be dedicated to maintenance vs new development; have the teams tag their stories accordingly and periodically report out actual vs plan.

Clean interfaces... the value of a Scrum team is to maximize communication within the team. If one must have multiple teams throughout the value stream, the goal is to define team boundaries where communication is "clean" or least complicated.

In the case of development and test or operations, because of communication complexity and often a wall exists between the two, the best solution is to include development, test, and operations people

on the same team; as a team, they will learn from each other and improve together. When a separate "integration" or "system" team must exist, then the goal is to enable the development teams to perform the integration and deployment testing (automation) frequently rather than to do it for them. Customers often are not dedicated on the team and the Product owner operates as their proxy to approve the completed increment.

Get Started... 99% of the time it is not possible to follow all the above guidelines. Scrum is designed to be an empirical system; it is the scientific method applied to business. Only when the system is run, can data be collected and the system improved. Hence there needs to be a sense of urgency in getting started and there is also an discomfort with the unknown. When a team begins operating as an agile team, there is reason to celebrate: "Let the learning begin;" and do reinforce that we are not at the end, rather at the beginning of the journey of learning and continuous improvement.

Keep an open mind... what is working, what is not, where is the next opportunity. Agile is a mindset of continuous improvement. Scrum is the initial framework that enables learning.