

Developing Agile Team Maturity

Tamara Sulaiman, PMP

January 24, 2008

Successful agile project teams make decisions and act on those decisions; plan and schedule their own work; and take full responsibility for getting the work done. But these highly productive teams don't just happen. They require guidance and support from team leaders, be it project managers or Scrum Masters, who modify their own behaviors as they help the team evolve into a self-organizing, independent force.

An agile team's highest priority is to regularly deliver business value to the customer. By meeting the goal to "commit and deliver" potentially shippable code in each iteration, a mature agile team can easily double or quadruple its productivity. The mature agile team can also attain significant increases in the quality of delivered code in the form of fewer bugs, and realize significant improvements in the level of customer satisfaction.

Empowered, self-organizing agile teams mature over time, often with many bumps and thumps along the way. As agile team leaders — whether in the role of Project Manager, Scrum Master, Program Manager — we must cultivate the maturation of an agile team by understanding its changing needs through the stages of team development. By providing appropriate levels of support and guidance, we nurture the team's evolution into a more mature, self-organizing group. To do this, project leaders must understand where the team is on an Agile Team Maturity Scale, and how to modify their own behaviors appropriately to facilitate team progress.

An agile team must be empowered to deliver their work. Kimball Fisher, in *Leading Self Directed Work Teams*, defines team empowerment as a function of having authority, resources, information and accountability — $\text{Team Empowerment} = f(\text{Authority, Resources, Information, Accountability})$. If any one of these things is missing, team empowerment is zero — $\text{Team Empowerment} = 0$ if either Authority, Resources, Information or Accountability = 0.

Truly self-organizing teams don't just happen. If you are new to the team, or team members are new to each other, it is important as a team leader to emphasize the authority and responsibility for defining the *how* rests with them. As a team leader you are there to remove impediments, communicate with stakeholders outside the team, and guide the team to follow the agile values and practices. Team members are often unsure how to plan and act on their own decisions as a group, and often are influenced by the belief that management does not want them to pursue that course. As team leader, your behavior is critical in convincing team members to act and demonstrate traits of being empowered and self-organizing.

Self-organizing teams make decisions and act on those decisions; plan and schedule their own work; have the responsibility for getting the work done; and take responsibility for completing the work. It is critical that the organization supports the idea that the delivery team, not the project manager or team leader, is directly responsible for delivering work. In contrast, traditional project team structure generally invests the authority for making decisions, planning work, and the responsibility for making sure the work gets done in the project manager and other people in management.

In 1965, Bruce Tuckman wrote an article describing developmental sequence in small groups. This became the basis for the classic "forming, storming, norming and performing" stage model of team development. Your understanding of where the team sits in this model can help you to assist their progress into the performing stage as an Agile team. It is important to note that there are no hard and fast guidelines for how long teams take to move through the stages. Development stages are not necessarily static either. Events or influences can move a team "back" from one stage to another. One example of an event that will impact a team's development is losing (or gaining) a team member. Some particularly difficult impediments can move a team from the performing to the norming or storming stages. Some simple observations and questions will help you understand where the team is in this model:

Forming — New teams are figuring out what the task is that they've been asked to complete, who they are working with, and what is acceptable behavior. By providing and emphasizing the vision of what it is that the team needs to build, and facilitating the team's discussions as they explore the parameters of the what it is they are to do, ensuring the team understands that they are empowered to define the how so that they can commit and deliver the software, you can help your team move through this stage relatively quickly.

Storming — As the name implies, conflict is high in this stage. Team members are asking questions such as "Who are you? What are your strengths? What is your role? What is my role? How do we fit together as a team?" Emotions are evident as the individuals work through interpersonal communication styles while resolving impediments. This stage is probably the easiest to identify. As the team works through problems and conflicts they are establishing agreements on how they will continue to perform together. As a team leader, facilitating discussions on the "definition of done" and other working agreements can help the team move through this stage. Not all teams make it through the storming stage without significant intervention. Changing team members in order to allow the team to "gel" is not out of the realm of possibility.

Norming — Group cohesion has begun to form as group decisions are made, problems solved and team members begin to understand the communication styles of their teammates. The group has experienced some level of task success, and has developed formal and informal working agreements that define how the work is accomplished. Team members are answering the question, "How will we work together?"

Performing — In this stage, the team has experienced success in different areas across multiple iterations. Team members are committed to the group objectives and strive to maintain harmony amongst the group. Impromptu social activities outside of the work become evident. The team will gather for lunch on a regular basis or introduce some other social norm. One team I worked with had the use of a foosball table near their group area. I believe that playing foosball together helped that team solidify their team cohesion. This was apparent in their ability to quickly solve difficult problems without moving back into the storming phase. In the performing stage the question moves from “How can we work together to achieve success?” to “How can we improve as a team?” The retrospective, an important part of the Agile inspect and adapt cycle takes on increased value and visibility as the team tackles more difficult impediments and works together to improve their own performance.

When I’m coaching a team new to agile development, I like to hold a formal workshop where together we define a backlog of topics for working agreements that the team feels is important and then work through that backlog. Sample topics include understanding where testing fits into to the iterative development cycle, how often check-in and builds happen, group structure (who does what within the team) and agreements around the frequency of pair programming activities. I have found that by spending time together as a team discussing items directly related to work, the true team building occurs. Jean Tabaka, author of *Collaboration Explained*, recently validated this observation. In response to a question about team building activities, Jean explained that she “avoids the use of team building games, exercises and simulations that aren’t related to what the team has committed to produce. True team building occurs through the accomplishment of work.”

Leading a Self-Organizing Team

When I coach new agile team leaders, I emphasize that leading a self-organizing team does not mean behaving in a completely “hands off” manner. Teams new to agile need more guidance and support than teams that are experienced with agile processes. As teams mature in these processes and gain confidence in their ability to manage the work, less guidance is necessary. As team leader, you can judge the team’s progress along an Agile Team Maturity Scale by their behavior towards each other, and by growing team independence involving certain “milestones.” Such milestones on my Agile Team Maturity Scale include:

- __ **Holding daily stand up meetings**, even if the team leader is not actually present.
- __ **Collaborating and managing changes** in the work mid-iteration without prompting from the team leader.
- __ **Planning and conducting iteration demos** without the team leader guiding the process.
- __ **Identifying areas for improvement during the retrospective meeting**, and delivering on those actions identified during the sprint without the team leader’s reminder.
- __ **Taking ownership for iteration planning meetings**. The team leader’s participation and facilitation is still critical, but as the team takes on more responsibility for this meeting and the results, they are moving up the maturity scale.

Trust your judgment about what the team needs as they move through the iterations. By modifying your behavior appropriately, you can facilitate your team’s growth towards independent self-organization. Understanding the stages of team development and recognizing opportunities to nurture the team’s agile maturity is an important skill for agile team leaders.

Tamara Sulaiman is a managing consultant at [SolutionsIQ](#), where she coaches teams and organizations transitioning to Scrum. She is co-originator of AgileEVM materials and processes that integrate earned value management with the Scrum framework. Tamara is currently focused on Scrum and Agile at the program and portfolio level, and integrating quality and financial metrics in a ‘dashboard’ format for evaluating project and portfolio health.

Copyright © 2019 ProjectManagement.com All rights reserved.

The URL for this article is:

<https://www.projectmanagement.com/articles/240094/Developing-Agile-Team-Maturity>