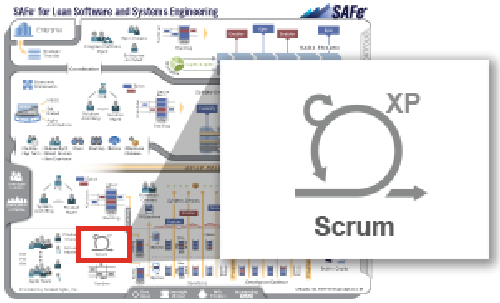


(/)

> . . . a holistic or "rugby" approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today's competitive requirements.
>
> —Nonaka and Takeuchi, The New, New Product Development Game

ScrumXP Abstract

Most SAFe teams use Scrum as their primary, team-based project management framework. Scrum is a lightweight yet disciplined and productive process for cross-functional, self-organized teams to operate within the context of SAFe. Scrum prescribes three roles: Scrum Master, Product Owner, and development team members [3]. The Scrum Master is a servant leader who helps the team adhere to the rules of Scrum and works within and outside the team to remove impediments. The Product Owner (PO) is responsible for defining what gets built. He or she is the owner of the team's backlog (*Product Backlog* in basic Scrum, *Team Backlog* in SAFe). When extended by Lean quality practices and engineering techniques inspired by eXtreme Programing (XP), the SAFe *ScrumXP* team provides the basic Agile building block for SAFe.

But, of course, SAFe ScrumXP teams do not work in isolation. Each team is part of the larger Agile Release Train (ART), where they cooperate in building the larger system, which is the subject of the endeavor. To this end, to ensure that the entire system iterates and evolves incrementally over time, all Agile Teams *plan together*, *integrate and demo together*, and *learn together*.

Details

The *ScrumXP* Agile Team (/agile-teams/) is a self-organizing, self-managing, and cross-functional group of five to nine people, collocated wherever possible. The size and structure of the team are optimized for communication, interaction, and the ability to deliver value. Self-organization implies that there is no team leader or manager role that tasks the team members, estimates their work, commits the team to specific objectives, nor determines how exactly they will advance the Solution (/solution/). The team is presented with the intent of the Iteration (/iterations/) and is solely responsible for determining how much of that scope they can actually commit to, and how they are going to build that increment of value. The team is cross-functional, so it has all the roles and skills needed to deliver an increment. The self-organization and cross-functional nature of the team, along with constant communication, constructive conflict, and dynamic interaction, can create a productive team and a more enjoyable work environment for the team members.

Scrum defines two specific roles, the Product Owner (/product-owner/) and Scrum Master (/scrum-master/), who are members of the SAFe ScrumXP (/scrumxp/) team, each with a unique and specific set of responsibilities. Each of these roles is elaborated further in a SAFe article by that name. A summary of the responsibilities for each is provided below.

Product Owner (PO)

Each ScrumXP team has a Product Owner who is responsible for the Team Backlog (/team-backlog/). The interaction of the PO with the rest of the team is a significant, daily, and highly focused effort. Therefore, the most effective model is to have a dedicated PO for each team, or to share a PO across no more than two teams. That allows the PO to effectively support the team during Iteration Execution (/iteration-execution/) by answering questions, providing more detail on the functionality under development, and reviewing and accepting the completed stories into the baseline.

Scrum Master (SM)

The Scrum Master is the facilitator and Agile coach for the team. Primary responsibilities include ensuring that the process is being followed, educating the team in Scrum, XP, and SAFe practices, and providing the environment for continuous improvement. The Scrum Master is also typically charged with leading the removal of impediments. The Scrum Master may be a full- or part-time role for a team member. Alternately, some dedicated Scrum Masters may support two to three Scrum teams.

The Scrum Process

The Scrum process itself is a lightweight project management framework that fosters quick, iterative advancement of the solution Capabilities (/features-and-capabilities/) and facilitates continuous improvement in support of higher productivity and quality and better outcomes. The process is centered around the iteration (note: Scrum uses the term *Sprint*; SAFe uses the more general term *iteration)* —a short timebox, two weeks in the case of SAFe—during which the team defines, builds, tests, and reviews results. The Scrum process is further described in the sections below.

process is further described in the sections below.

Planning the Iteration

The iteration starts with Iteration Planning (/iteration-planning/)—a timeboxed event (four hours or less) during which the Product Owner presents the Stories (/story/) for the iteration. The team then reviews the stories, defines the acceptance criteria, splits larger stories into smaller ones where necessary, estimates them in story points, and, finally, commits to what they can build, based on their known velocity (story points per iteration). Many teams further divide stories into tasks, estimating the tasks in hours to better refine their understanding of the work ahead. Finally, the team commits to a set of goals for the iteration.

Even before the iteration starts, the Scrum team is preparing content for the new iteration by *refining* the team backlog items to make sure they understand the work to be delivered during the upcoming iteration.

Visualizing Work

During execution, the team builds and tests stories, with a goal of delivering a story or two every few days. This serialization limits work in process and helps avoid "waterfalling" the iteration. Teams use "big visual information radiators" (BVIRs) to understand and track progress during iteration execution. The team's s*tory board* visualizes the stories and their progress throughout the iteration. In so doing, they often use development steps as the columns and move stories left to right over time, as Figure 1 demonstrates.

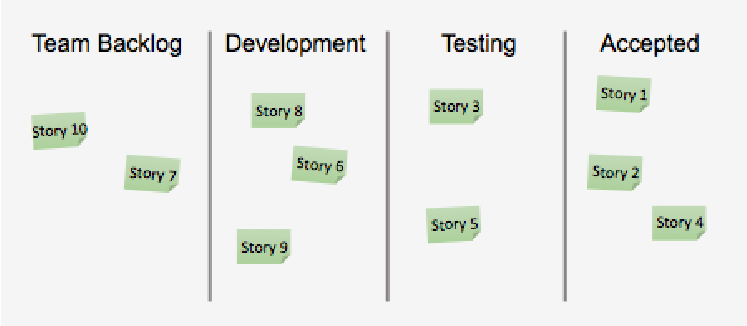


Figure 1. An example of a team's story board

Some teams also apply Work-in-Process limits (/visualize-and-limit-wip-reduce-batch-sizes-and-manage-queue-lengths/) to some steps to create a "pull" process within the iteration, and to continuously balance the work to increase throughput. Indeed, many teams integrate the best practices of Scrum and Kanban to facilitate the flow of work through the iterations. In this case, the simple story board above evolves into a more structured Kanban board. See the Team Kanban (/team-kanban/) article for more on the use of Kanban by ScrumXP teams.

Coordinating with Daily Stand-up Meetings

Each day, the team has a formal ceremony—the *Daily Stand-up Meeting* (DSU)—to understand where they are, escalate problems, and get help from other team members. During this meeting each team member describes what they did yesterday, what they are going to work on today, and any “blocks” they are encountering. As this is a *daily* coordination meeting, it has to be kept short and to the point by the Scrum Master. The DSU should take no more than 15 minutes and is done standing up in front of the story board.

Team communication does not end there, however, as team members interact continuously throughout the iteration. The facilitation of such communication is the main reason why ScrumXP prefers that the team be collocated whenever possible.

Demonstrating Value and Improving the Process

At the end of each iteration, the team conducts a Team Demo (/team-demo/) and an Iteration Retrospective (/iteration-retrospective/). During the demo, the team demonstrates each story accomplished during the iteration, the summation of which is the *team’s increment* of value for that iteration. This is not a formal status report; rather, it is a review of the tangible outcomes of the iteration. Thereafter, the team conducts a brief retrospective, a time to reflect on the iteration, the process, things that are working well, and current obstacles. Then the team comes up with improvement stories for the next iteration.

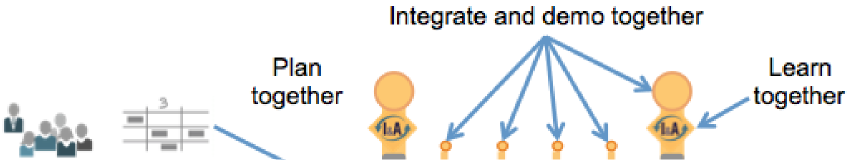
Building Quality In

As a tenet of SAFe says, “You can’t scale crappy code.” Therefore, one of the Core Values of SAFe (/safe-core-values/) is “Built-in Quality.” Built-in Quality (/built-in-quality/) begins at the code and component levels by those who create the solution, otherwise it is difficult or impossible to ensure quality later as the solution integrates and scales from component to system and solution.

To make sure teams build quality in to code and components, SAFe describes five engineering and quality practices that are inspired by the tenets of XP and that augment the project management practices of Scrum. They are Continuous Integration (/continuous-integration/), Test-First (/test-first/), Refactoring (/refactoring/), pair work, and collective ownership. Some teams use other XP practices in addition to these five, such as a pair programming and metaphor [2].

ScrumXP Teams Are “on the Train”

While the teams are cross-functional, it isn’t always realistic for a team of seven or eight to deliver end-user value when a large system includes different technology platforms and a spectrum of disciplines including hardware, software, and systems engineering. Many more teams are typically required. To address this, SAFe Agile Teams teams operate within an Agile Release Train (ART) (/agile-release-train/), which provides mission alignment and a collaborative environment whereby teams can cooperate with other teams to build the larger solution capabilities. As part of the ART, ScrumXP teams *plan together*, *integrate and demo* together, and *learn together*, as illustrated in Figure 2.



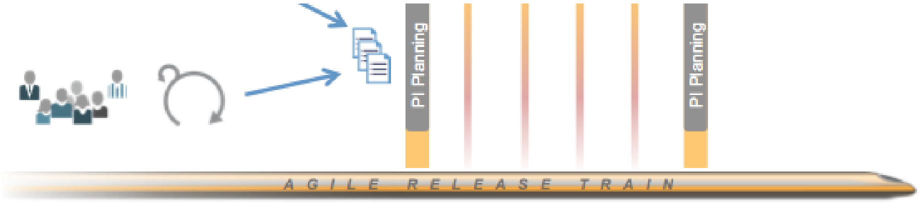


Figure 2. SAFe Agile Teams plan together, integrate and demo together, and learn together

Each team's participation in this "shared responsibility program" is further defined in the Agile Teams (/agile-teams/) article.

Leadership of ScrumXP Teams

Managers are typically not part of the cross-functional team. However, the initial organization of such teams around features, components, and subsystems and the design and structure of the ART is typically a management responsibility, based on input from the teams. Thereafter, the ongoing management of such teams typically undergoes a qualitative shift, from *manager as expert*, directing the team to specific technical achievements, to manager as a Lean-Agile Leader (/lean-agile-leaders/) and *enabler and developer of people*.

Learn More

[1] Kniberg, Henrik. *Scrum and XP from the Trenches.* lulu.com, 2015.

[2] Beck, Kent and Cynthia Andres. *Extreme Programming Explained: Embrace Change*, 2nd edition. Addison-Wesley, 2004.

[3] Sutherland, Jeff and Ken Schwaber. Scrumguides.org.

Last update: 6 April 2016

FRAMEWORK

Downloads (/posters/)

Latest Updates (/blog/)

SAFe 3.0 (http://v3.scaledagileframework.com)

TRAINING

Course Calendar (http://www.scaledagileacademy.com/events/event_list.asp)

About Certification (http://www.scaledagileacademy.com/?page=WhichCertification)

Become a Trainer (http://www.scaledagileacademy.com/?page=becomeatrainer)

PERMISSIONS

Permission FAQ (http://www.scaledagile.com/permission-faq/)

Permissions Form (http://www.scaledagile.com/permissions-form/)

Usage and Permissions (/usage-and-permissions/)

PARTNER

Becoming a Partner (http://www.scaledagile.com/become-a-partner/)

Partner Directory (http://www.scaledagile.com/listingcategory/directory/)

Partner Event Calendar (http://www.scaledagile.com/event-list/)

GET SOCIAL

Twitter (https://twitter.com/ScaledAgile)

Linkedin (https://www.linkedin.com/grps/Scaled-Agile-Framework-4189072?)

YouTube (https://www.youtube.com/user/scaledagile)

SlideShare (http://www.slideshare.net/ScaledAgile)

RECENT POSTS

New Essential SAFe guidance article (/new-essential-safe-guidance-article/)