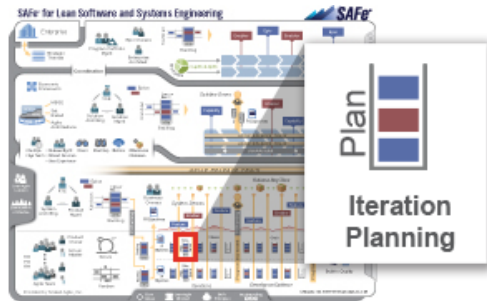


<https://twitter.com/ScaledAgile>
<https://www.linkedin.com/grps/Scaled-Agile-Framework-4189072?>

<https://www.youtube.com/user/scaledagile>
<http://www.slideshare.net/ScaledAgile>



(/)

**||** *Stay committed to your decisions, but stay flexible in your approach.*

—Tom Robbins

## Iteration Planning Abstract

Team-based *Iteration Planning* is one of the key mechanisms for decentralizing control and empowering fast, local decision-making. In Scrum, teams plan by selecting Stories from the product backlog and committing a set of them for execution in the upcoming Iteration. This basic process is fundamental to SAFe as well, but the context is broader, as SAFe teams are part of an Agile Release Train (ART). As such, the team's backlog has already been seeded and partially preplanned during the PI Planning meeting. In addition, the teams have feedback, not only from their prior iterations but from the System Demo and other teams as well. That, and the natural course of events and changing fact patterns, provides the broader context for iteration planning. However, the iteration planning process is largely the same as in traditional Scrum, as each Agile Team plans the upcoming iteration in a timeboxed meeting. The output of this iteration planning meeting is:

1. The iteration backlog, consisting of the stories committed to the iteration, with acceptance criteria where appropriate
2. A statement of Iteration Goals, typically a sentence or two stating the business objectives of the iteration
3. A commitment by the team to the work needed to achieve the goals

# Details

The purpose of *Iteration Planning* is to organize the work and define a realistic scope for the Iteration (/iterations/). Each Agile Team (/agile-teams/) agrees on a set of Stories (/story/) for the upcoming iteration (the iteration backlog) and summarizes those stories into a set of Iteration Goals (/iteration-goals/). The iteration backlog and goals are based on the team's capacity and allow for consideration of each story's complexity, size, and dependencies on other stories and other teams. At the end of planning, the teams commit to the goal of the iteration and adjust stories as necessary to achieve the larger purpose. In return, management does not interfere or adjust the scope of the iteration, allowing the team to stay focused on the goals.

## Inputs to Iteration Planning

In SAFe, iteration planning is a refinement of the level of detail and an adjustment of the initial iteration plans created during Agile Release Train (/agile-release-train/) PI Planning (/pi-planning/). Teams approach iteration planning with a pre-elaborated Team Backlog (/team-backlog/) (they have usually held a backlog refinement meeting during the previous iteration). There are a number of inputs to the planning meeting:

- The Team (/pi-objectives/) and Program PI Objectives (/pi-objectives/), created at PI planning
- The team's PI plan backlog, which consists of stories that were identified during PI planning
- Additional stories that arise based on local context, including items such as defects, Refactors (/refactoring/), and new stories that have come about since the planning session
- Feedback from the prior iteration, including any stories that were not successfully completed (did not meet the Definition of Done; see "A Scaled Definition of Done" in the Release (/release/) article) in that iteration
- Feedback from the System Demo (/system-demo/)

## Planning the Iteration

Prior to the meeting, the Product Owner (/product-owner/) will have prepared some preliminary iteration goals, based on the team's progress in the Program Increment (PI) (/program-increment/). Typically, the Product Owner starts the meeting by reviewing the proposed iteration goals and the higher-priority stories in the team backlog. During the meeting, the Agile Team discusses implementation options, technical issues, Nonfunctional Requirements (NFRs) (/nonfunctional-requirements/), and dependencies, then plans the iteration. The Product Owner defines the *what*; the team defines *how* and *how much*.

Throughout the meeting, the team elaborates the acceptance criteria and estimates the effort to complete each story. Based on their velocity, the team then selects the candidate stories. Many teams break each story down into tasks and estimate them in hours to confirm that they have the capacity and skills to complete them. Once confirmed, the team commits to the work and records the iteration backlog in a visible place, such as a storyboard or tooling. This meeting is timeboxed to a maximum of four hours.

Establish Velocity

## ESTABLISH VELOCITY

First, the team quantifies their capacity to perform work in the upcoming iteration. Each team member determines their availability, acknowledging time off and other potential duties. This activity also takes into account other standing commitments (see the “capacity allocation” section in Team Backlog (/team-backlog/))—such as maintenance—that are distinct from new story development.

## Iteration Goals

Once team member capacity has been established, the team turns their attention to understanding and agreeing on one or more iteration goals that are based on the team and program PI objectives from the PI planning session. The closer this iteration is to the PI planning session, the more likely the program objectives will remain unchanged.

## Story Analysis and Estimating

With the proposed iteration goals acting as a reference point, the team backlog is reviewed. Each story is discussed, covering relative difficulty, size, complexity, technical challenges, and acceptance criteria. Finally, the team agrees to a size estimate for the story. There are typically other types of stories on the team backlog as well, including Enablers (/enablers/) that could constitute infrastructure work, refactoring, research Spikes (/spikes/), architectural improvements, and defects. These are also prioritized and estimated.

## Tasks

Many teams then break each story into tasks. As the tasks are identified, team members discuss each one: who would be the best person(s) to accomplish it, approximately how long it will take (typically in hours), and any dependencies it may have on other tasks or stories. Once all this is understood, a team member *takes responsibility* for a specific task or tasks. As team members commit to tasks, they reduce their individual iteration capacity until it reaches zero. Often, toward the end of the session, some team members will find themselves overcommitted, while others will have some of their capacity still available. This situation leads to further discussion among team members to more evenly distribute the work.

## Commitment

When the team’s collective capacity has been reached in terms of committed stories, no more stories are pulled from the team backlog. At this point, the product owner and team agree on the final list of stories that will be achieved, and they revisit and restate the iteration goals. The entire team then commits to the iteration goals, and the scope of the work remains fixed for the duration of the iteration.

## Attendees

Attendees of the iteration planning meeting include:

- The Product Owner

- The Scrum Master (/scrum-master/), who acts as facilitator for this meeting
- All other team members
- Any other stakeholders, including representatives from other Agile Teams or the ART, and subject matter experts

## Agenda

A sample iteration planning meeting agenda follows:

- The team determines the velocity available for the iteration
- The Product Owner presents the iteration goals; the team discusses and agrees on the goals
- The team discusses each story in ranked (sequenced or prioritized) order. For each story, the team:
  - Sizes/resizes each story in story points and splits them if necessary
  - Elaborates acceptance criteria through conversation
  - Based on size and value/time/risk, the PO may re-rank stories
  - Optional: In ranked order, the team breaks stories into tasks, estimated in hours, and takes responsibility
  - Planning stops once the team runs out of points
- The team and PO negotiate and finalize the selected stories
- Everyone commits to the iteration goals

## Guidelines

Below are some tips for holding a successful iteration planning meeting:

- Timebox the meeting to 4 hours or *less*
- This meeting is held *by* and *for* the team
- A team should never commit to work in excess of its adjusted capacity or historical velocity

## Relative Estimating, Velocity, and Normalizing Story Point Estimating

Agile Teams use relative estimating [2, 3] to estimate the size of a story in story points. With relative estimating, the size (effort) for each story is estimated relative to the size of other stories. The team's velocity for an iteration is equal to the sum of the size of all the stories completed in the iteration. Knowing a team's velocity assists with planning and is a key factor in limiting WIP, as teams simply don't take on more stories than their prior velocity would allow. Velocity is also used to estimate how long it takes to deliver larger Features (/features-and-capabilities/) or Epics (/epic/), which are likewise estimated in story points.

### Normalizing Story Point Estimating

In standard Scrum, each team's story point estimating—and the resultant velocity—is a local and independent matter. The fact that a small team might estimate in such a way that they have a velocity of 50, while a larger team estimates so as to have a velocity of 12, is of no concern to anyone. In SAFe, however, story point velocity must be normalized to a point, so that estimates for features or epics

or no concern to anyone. In short, however, story point velocity must be normalized to a point, so that estimates for features or epics that require the support of many teams are based on rational economics. After all, there is no way to determine the return on potential investment if you don't know what the investment is.

In order to do this, SAFe teams start down a path where a story point for one team means about the same as a story point for another. In this way, with adjustments for economics of location (the U.S., Europe, India, China, etc.), work can be estimated and prioritized based on economics by converting story points to cost. This is particularly helpful in initial PI planning, as many teams will be new to Agile and will need a way to estimate the scope of work in their first PI. One starting algorithm is as follows:

- For every individual contributor on the team, excluding the Product Owner, give the team eight points (adjust for part-timers).
- Subtract one point for every team member vacation day and holiday.
- Find a small story that would take about a half-day to develop and a half-day to test and validate. Call it a 1.
- Estimate every other story relative to that one.

Example: Assuming a 6-person team composed of 3 developers, 2 testers, and 1 PO, with no vacations, etc., then the estimated initial velocity = 5 \* 8 points = 40 points per iteration. (Note: The team may need to adjust a bit lower if one of the developers and testers is also the Scrum Master.)

In this way, all teams estimate the size of work in a common fashion, so management can thereby fairly quickly estimate the cost for a story point for teams in a specific region. They then have a meaningful way to establish the aggregate cost estimate for an upcoming feature or epic.

**Note: There is no need to recalibrate team estimating or velocities after that point. It is just a common starting point.**

While teams will tend to increase their velocity over time—and that is a good thing—in fact, the number tends to remain fairly stable, and a team's velocity is far more affected by changing team size, makeup, and technical context than by productivity changes. And, if necessary, financial planners can adjust the cost per story point a bit. This is a minor concern compared to the wildly differing velocities that teams of comparable size may have in the non-normalized case.

---

## Learn More

[1] Leffingwell, Dean. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley, 2011, chapter 9.

[2] Leffingwell, Dean. *Scaling Software Agility: Best Practices for Large Enterprises*. Addison-Wesley, 2007, chapter 10.

[3] Cohn, Mike. *Agile Estimating and Planning*. Robert C. Martin Series. Prentice Hall, 2005.

Last modified 15 April 2016

The information on this page is © 2010-2016 Scaled Agile, Inc. and is protected by US and International copyright laws. Neither images nor text can be copied from this site without the express written permission of the copyright holder. Scaled Agile Framework and SAFe are registered trademarks of Scaled Agile, Inc. Please visit Permissions FAQs (<http://scaledagile.com/permission-faq/>) and contact us (<http://www.scaledagile.com/permissions-form/>) for permissions.

## FRAMEWORK

[Downloads \(/posters/\)](#)

[Latest Updates \(/blog/\)](#)

[SAFe 3.0 \(http://v3.scaledagileframework.com\)](http://v3.scaledagileframework.com)

## TRAINING

[Course Calendar \(http://www.scaledagileacademy.com/events/event\\_list.asp\)](http://www.scaledagileacademy.com/events/event_list.asp)

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

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

## PERMISSIONS

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

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

[Usage and Permissions \(/usage-and-permissions/\)](#)

## PARTNER

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

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

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

## GET SOCIAL

[Twitter \(https://twitter.com/ScaledAgile\)](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/)**

Nov, 30th 2016

**SAFe Website Updates and new Glossary (/safe-website-updates-and-new-glossary/)**

Nov, 29th 2016

**End of life for SAFe 3.0 website and courseware 12/30/16 (/eol-safe3/)**

Nov, 28th 2016

## SCALED AGILE, INC

### CONTACT US

5480 Valmont Rd., Suite 100  
Boulder, CO 80301 USA

Email: [support@scaledagile.com](mailto:support@scaledagile.com) (mailto:support@scaledagile.com)

Phone: +1.303.554-4367

### BUSINESS HOURS

Weekdays: 9am to 5pm

Weekends: CLOSED