



[Agile](#) / Sprints

Sprints

A sprint is a short, time-boxed period when a scrum team works to complete a set amount of work.



BY MAX REHKOPF

What are sprints?

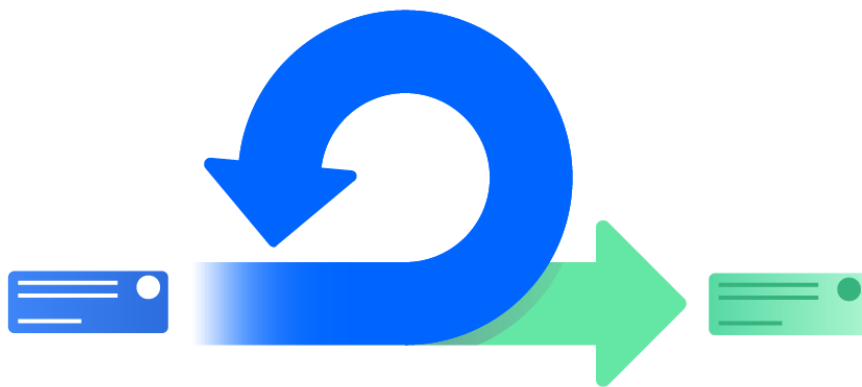
A sprint is a short, time-boxed period when a scrum team works to complete a set amount of work. Sprints are at the very heart of scrum and agile methodologies, and getting sprints right will help your agile team ship better software with fewer headaches.

“With Scrum, a product is built in a series of iterations called sprints that break down big, complex projects into bite-sized pieces,” said Megan Cook, Group Product Manager for Jira Software at Atlassian.

Up Next
[Learn Versions with Jira Software](#) →



| to adapt to change."



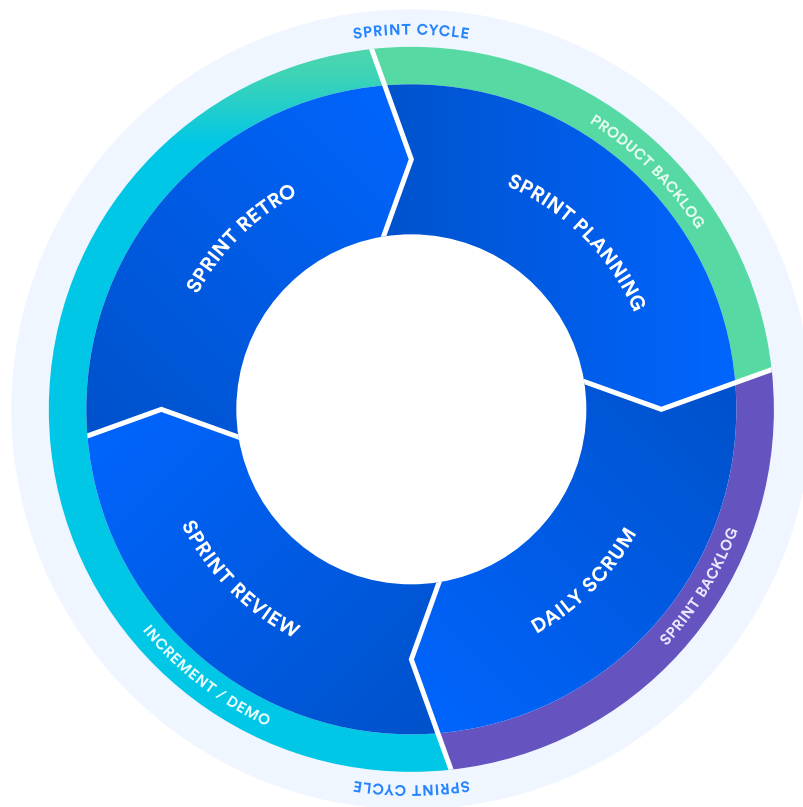
Many associate scrum sprints with agile software development, so much so that scrum and agile are often thought to be the same thing. They're not. **Agile** is a set of principles and **scrum** is a framework for getting s#it done.

The many similarities between agile values and scrum processes lead to a fair association. Sprints help teams follow the agile principle of "delivering working software frequently," as well as live the agile value of "responding to change over following a plan." The scrum values of transparency, inspection, and adaptation are complementary to agile and central to the concept of sprints.





How to plan and execute scrum sprints



The scrum folks really did think of everything. In order to plan your upcoming sprint, you use the sprint planning meeting! [Sprint planning](#) is a collaborative event where the team answers two basic questions: What work can get done in this sprint and how will the chosen work get done?





should achieve and the [product backlog](#) items that, upon completion, would achieve the sprint goal.

The team then creates a plan for how they will build the backlog items and get them “Done” before the end of the sprint. The work items chosen and the plan for how to get them done is called the sprint backlog. By the end of sprint planning the team is ready to start work on the sprint backlog, taking items from the backlog, to “In-progress,” and “Done.”

During a sprint, the team checks in during the daily scrum, or [standup](#), about how the work is progressing. The goal of this meeting is to surface any blockers and challenges that would impact the teams ability to deliver the sprint goal.

After a sprint, the team demonstrates what they’ve completed during the [sprint review](#). This is your team’s opportunity to showcase their work to stakeholders and teammates before it hits production.

Round out your sprint cycle with my favorite meeting, the [sprint retrospective](#). This is your teams opportunity to identify areas of improvement for the next sprint. With that, you’re ready to start your next sprint cycle. Onward!

Do's and Don'ts

Even with the basics down, most teams will stumble as they start to run sprints. Megan Cook rounds out this discussion with some Do's and Don'ts she's picked up over the years.





- Make sure the team sets and understands the sprint goal and how success will be measured. This is the key to keeping everyone aligned and moving forward toward a common destination.
- Do ensure you have a well-groomed backlog with your priorities and dependencies in order. This can be a big challenge that could derail the process if it's not properly managed.
- Ensure you have a good understanding of velocity, and that it reflects things like leave and team meetings.
- Do use the Sprint Planning meeting to flesh out intimate details of the work that needs to get done. Encourage team members to sketch out tasks for all stories, bugs, and tasks that come into the sprint.
- Leave out work where you won't be able to get the dependencies done, like work from another team, designs, and legal sign-off.
- Finally, once a decision or plan is made, make sure someone captures that information in your project management or collaboration tool, like your Jira tickets. That way, both the decision and the rationale are easy for everyone to see later.

While you're working on being a Scrum all-star with these "do's," watch out for a few red flags too:

Don't:

- Don't pull in too many stories, overestimate velocity, or pull in tasks that can't be completed in the sprint. You don't want to set yourself or your team up for failure. →



- Don't let the team have a fuzzy view of what's in the sprint. Nail it down, and don't focus so much on moving *fast* that you forget to make sure everyone's moving in *the same direction*.
- Also, don't take on a large amount of unknown or high-risk work. Break down stories that are large or have high uncertainty, and don't be afraid to leave some of that work for the next sprint.
- If you hear concerns from the team, whether it's about velocity, low-certainty work, or work they think is bigger than what they estimated, don't ignore it. Address the issue, and recalibrate when necessary.

Learn more about sprints

Sprints are so well known(and so effective!) that they're often seen as the first step on the path towards greater agility. As we've learned, mastering sprints requires a mastery of a handful of scrum and agile concepts that build upon each other. Please use [the rest of our articles on scrum](#) to round out your knowledge and inch ever closer towards scrum bliss.

SHARE THIS ARTICLE





MAX REHKOPF

As a self-proclaimed "chaos muppet," my discovery of agile was like turning on the lights. I'm interested in the future of work, and believe in a theory-heavy but dogma-light future for agile and lean. You can find me on twitter @maxrehallday, and on medium @maxreh.

TUTORIAL

Learn scrum with Jira Software

A step-by-step guide on how to drive a scrum project, prioritize and organize your backlog into sprints, run the scrum ceremonies and more, all in Jira.

[Try this tutorial →](#)

ARTICLE





Learn how to facilitate great agile ceremonies like sprint planning, daily stand-ups, iteration review and retrospectives.

[Read this article →](#)

Agile Topics

Agile project management

Scrum

Kanban

Design

Software development

Product management

Teams

Agile at scale

DevOps

Sign up for more agile articles and tutorials.

Email

email@example.com



