Scrum is Simple!

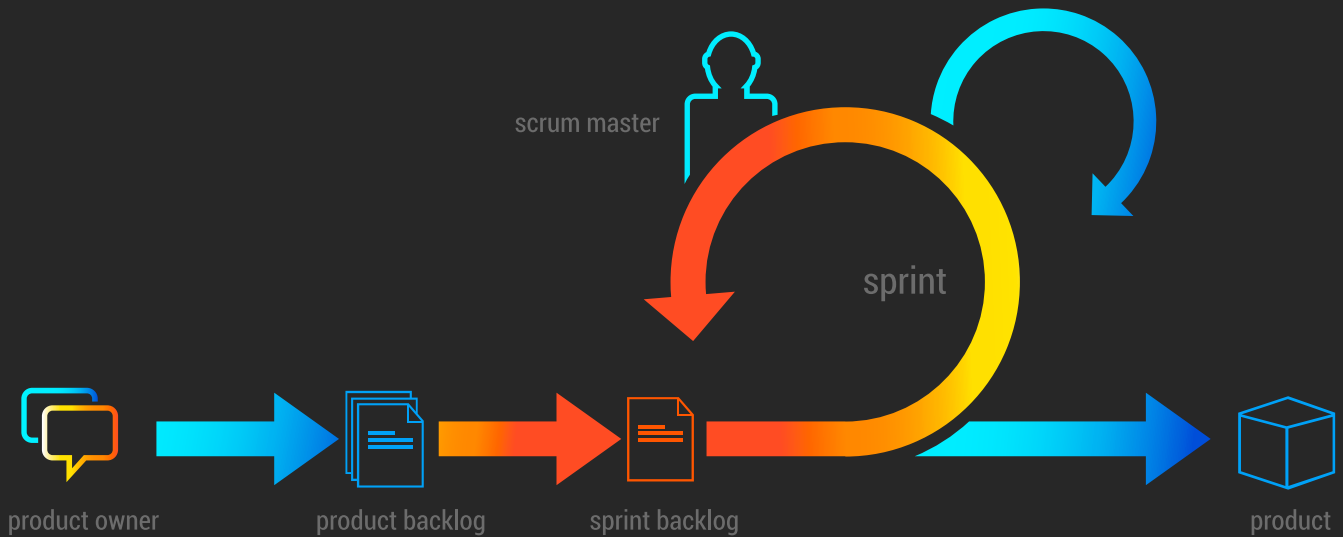scrum master

sprint

product owner → product backlog → sprint backlog → product

Dick Carlson
Earle Soukup

Abstract. Scrum implementation requires a significant cultural shift from traditional project management methods. Scrum roles are different from traditional roles and frequently cause confusion in engineering and business communities. Scrum terms are alien to most, and a radical change from the "status quo" is not only difficult but also initially resisted by most. Although Scrum was originally recommended for software development projects, it is applied easily to any type of project. This article provides a brief historical background and includes the basic concepts of Scrum.

Introduction

What is Scrum? [1] Scrum employs an iterative and incremental approach for managing projects. Scrum has no ties with the Project Management Institute (PMI) [2], and it is not part of the Project Management Body of Knowledge (PMBoK). [3] Although Scrum was originally recommended for software development projects, it is easily applied to just about any type of project. This means that Scrum can be and has been used as a framework to manage a wide range of project types within the activities and industries involved in or supporting software engineering, systems engineering, IT, finance, real estate, manufacturing, community service, fitness, health, science, defense, aerospace, religion, and household management, among others.

Scrum's History

In 1986, Hirotaka Takeuchi and Ikujiro Nonaka described a new approach to commercial product development that would increase speed and flexibility. The approach was based on case studies from manufacturing firms in the automotive, computer, photocopier, and printer industries. They called this the "holistic" or "rugby" approach because the whole process was performed by one cross-functional team across multiple overlapping phases, where the "Scrum" (or whole team) "tries to go the distance as a unit, passing the ball back and forth." This was another way of describing concurrent planning and execution. [4]

In 1991, Peter DeGrace and Leslie Hulet Stahl first referred to this as the "Scrum approach." In the early 1990s, Ken Schwaber used such an approach at his company, Advanced Development Methods, and Jeff Sutherland, with John Scumniotales and Jeff McKenna, developed a similar approach at Easel Corporation. They were the first to refer to it using the single word "Scrum." In 2001, Ken Schwaber teamed up with Mike Beedle to describe the method in their book "Agile Software Development with Scrum."

The Scrum Framework

Scrum is an Agile [5] approach for managing a project. Scrum was formalized originally for software development projects, but over the years, I have applied Scrum practices to a wide range of project types that required an innovative scope of work. The possibilities are endless. The Scrum model is deceptively simple, as shown in Figure 1.

The following summary describes a Scrum team's activities during each sprint:
— A Product Owner creates a prioritized list called a "product backlog."
— The team has a certain amount of time, called a "sprint," to complete its work — usually two to four weeks. The team meets each day to assess its progress during its "daily stand-up" meeting or "stand-up."
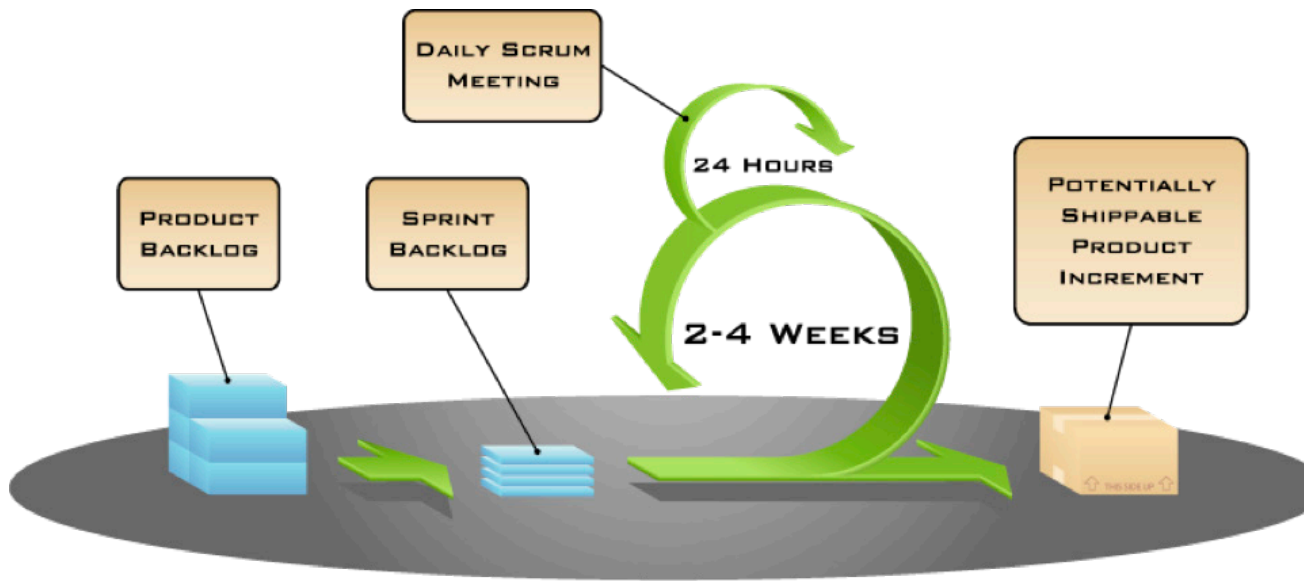
Figure 1. Scrum Framework

—During sprint planning, the team selects a small chunk of the highest priority items from the top of the product backlog, adds the selected work to a sprint backlog, and then decides how to implement that work.

—The assigned Scrum Master keeps the team focused on its goal and protects the team from organizational influences.

—At the end of each sprint, completed work should be deliverable, meaning ready to hand to a customer, placed in a repository for future additional functionality, deployed to a user, or demonstrated to a stakeholder.

—The sprint ends with a sprint review, which demonstrates all completed work followed by a retrospective meeting where the team identifies and implements potential process improvements.

—As the next sprint begins, the team chooses another chunk of the highest priority items in the product backlog and begins working again.

The cycle repeats until enough items in the product backlog have been completed to the satisfaction of the Product Owner and the customer, the budget is depleted, or a deadline arrives, which marks the end of all project work. No matter which impetus stops work, the implementation of Scrum assures all stakeholders that the most valuable work has been completed when the project ends.

Scrum Roles

Scrum's simple framework consists of three roles (and a few supplemental roles), three critical artifacts, and five low-overhead work activities. The roles include the following:

–**Product Owner:** The Product Owner represents the voice of the customer and is accountable for ensuring that the Team delivers value to the business. The Product Owner writes customer-centric items (typically in user story format), prioritizes the user stories, and adds the stories to the product backlog. Scrum teams should have one Product Owner, and while he or she may also be a member of the development team, it is recommended that this role not be combined with that of the Scrum Master.

–**Scrum Master:** A Scrum Master facilitates Scrum and is accountable for removing impediments that could prevent the team from delivering the iteration or the sprint's goals/deliverables. The Scrum Master is not a team leader but acts as a buffer between the team and any distracting influences. The Scrum Master ensures that the Scrum process is followed and enforces the agreed-upon rules. A key part of the Scrum Master's role is to protect the team and keep them focused on the tasks at hand. The role has also been referred to as "servant-leader" to reinforce these dual perspectives.

–**Team:** The Team is responsible for delivering the product. A Team is typically made up of five to nine people with cross-functional skills who do the actual work (analyze, design, develop, test, review, technical communication, document, etc.). Also, it is recommended that the Team be self-organizing and self-managing. During every sprint, the Team works closely with the Product Owner, grooming the product backlog to ensure it reflects current customer needs and priorities and that duplicate and no longer needed items in the backlog are removed. Other backlog activities often include decomposing user stories that are too large to implement in a single sprint, improving user stories that are poorly written, re-estimating user stories based on changes in scope, design, and other factors, and adding to or revising acceptance criteria.

Supplemental Scrum Roles

The ancillary roles in Scrum are filled by those with no formal role and who have infrequent involvement in the Scrum process but who nonetheless must be taken into account. These roles include the following:

–**Stakeholders:** These are the people (customers, users, suppliers, support groups, and anyone else with a vested interest in the project) who enable the project and for whom

the project will produce the agreed-upon benefits that justify its production. They are typically involved directly in the process during sprint reviews.

    –Management: People, including project managers, who are responsible for the establishment of the product development environment, personnel resources, budgets, and requisite technical support when needed.

    –Technical Owner: A person who advises on technical issues related to the product, the requirements, the environment, the infrastructure, and the team's capabilities.
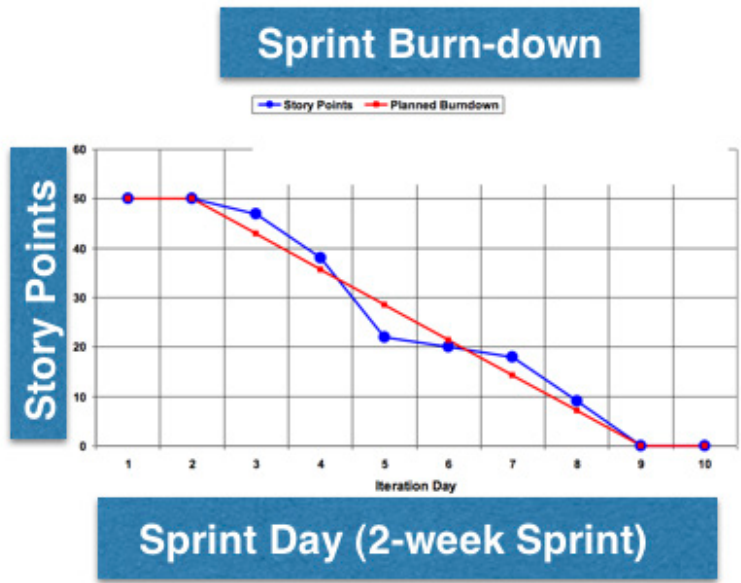

Figure 3: Sprint Burn-down

Scrum Artifacts

Scrum has three critical artifacts and one important artifact. They are:

    –The Product Backlog (Critical): The product backlog is a high-level list that is maintained throughout the entire project by the Product Owner. It aggregates backlog items — that is, broad descriptions of all potential features — prioritized as an absolute ordering by business value. It is therefore the "what" that will be built, sorted by importance. It is open and editable by anyone and typically contains rough estimates of business value and/or development complexity. These estimates help the Product Owner gauge the timeline and, to a limited extent, priorities. For example, if the "add spell-check" and "add table support" features have the same business value, the one with the smaller development complexity will probably have higher priority, because the return on investment (ROI) is higher. The product backlog and business value of each listed item is the property of the Product Owner. However, the associated development complexity is determined by the team. A common and very simple tool used to create the product backlog is a spreadsheet.

During each sprint and throughout the project life cycle, the product backlog is groomed through a coordinated effort of the team and the Product Owner to:

  –Remove backlog items (user stories) that no longer provide value.

  –Add or write new user stories as a result of customer-defined needs.

  –Re-prioritize backlog items as necessary.

  –Estimate new items added and re-estimate existing items that may have changed.

Figure 2: Task Board Sample

—Decompose stories that are too large to be completed in a single sprint.

—**The Sprint Backlog (Critical):** The sprint backlog is the list of work the team must address during the next sprint. Features are decomposed into tasks, which, as a best practice, normally should be between four and 16 hours of work. With this level of detail, the whole team understands exactly what to do. Tasks on the sprint backlog are never assigned; rather, tasks are selected by knowledgable team members as needed, according to the set priority and a team member's skills. This promotes self-organization of the team and developer buy-in.

The sprint backlog is the property of the team, and all included estimates are provided by the team. However, the Scrum Master may prefer to maintain this artifact to ensure that it reflects a sprint's status in real time. Often an accompanying task board or work-in-progress (WIP) board (Figure 2) is used to see and change the "state" of each story related to the tasks of the current sprint to "To do," "WIP," "Verify," and "Done." A common and very simple tool used to create the sprint backlog is a spreadsheet.

—**Sprint Burn-down Chart (Important):** The sprint burn-down chart (Figure 3) is displayed in plain view of the team to show work completed and work remaining in the sprint's backlog. Updated every day, the chart provides an easy-to-understand view of the sprint's progress. It also provides quick visualizations for reference. Note that the red line represents the planned completion of work within the two-week sprint. The blue line shows the actual daily burn-down of story points (product backlog work selected for the sprint).

—**Task Board (Critical):** The task board is an information radiator that supports the notion of transparency. It shows tasks in work decomposed from their user stories throughout the sprint. See Figure 2: Task Board Sample. The task board is updated by team members as they complete each task. Each item (story) selected from the product backlog for the sprint is placed on the task board for all to see the progress of work throughout the sprint. Team members check out the tasks as they accept them, then work on tasks until they are completed. When a task is completed, the team member moves it from the "WIP" column to the "To Verify" column. Then the Product Owner verifies its completion before initiating another task. The team maintains the task board up to the "To Verify" column. So, if a team member completes a task, he or she cannot take credit until the Product Owner verifies that it is done. This is among the best transparency tools used in Agile projects.

Scrum Activities

The Daily Scrum (or Stand-up): The Daily Stand-up is a work session that should be facilitated by the Scrum Master for new teams. Each sprint day, each member of the team tells other team members what they have completed and what they plan to do next. Experienced teams may become relatively autonomous in the conduct of stand-ups, therefore allowing the Scrum Master to spend time on other project-related activities.

Essentially, the team owns and runs the stand-up. The stand-up has specific guidelines that include the following:

—Starts on time at the same location and same time every day.
—Time-boxed at 15 minutes or fewer.
—Side conversations are not allowed.
—One team member talks at a time until all have spoken. All others listen.
—Problems are identified, but solutions are deferred to specific persons after the stand-up.
—Others are welcomed, but only the core roles are allowed to speak.

During the meeting, each team member responds to these three questions:

—What have you done since the last stand-up meeting?
—What are you planning to do next?
—What impediments do you have that prevent you from accomplishing your work?
—Do you have any issues?

The Scrum Master facilitates resolution of all impediments, although in order to keep the meeting to 15 minutes, the resolution occurs outside the meeting. The reason the stand-up is kept short is to avoid long-winded conversations and to limit member responses to one person at a time. I liked to conduct stand-up meetings in a private and quiet area without chairs. That's the main reason it is referred to as a "stand-up."

Sprint Planning: Sprint Planning is led by the Product Owner and is facilitated by the Scrum Master. At the beginning of every sprint, a two-part, time-boxed "sprint planning" session is conducted. Its purpose is to ensure that everyone understands all of the functionality to be completed during the next sprint.

During the first half of this session, the Product Owner explains to the team what needs to be completed or implemented during the sprint, prioritizing backlog items and allowing the team to estimate all work targeted for the sprint.

During the second half of Sprint Planning, the team does the following:

—Calculates its "velocity," or how much it feels it can complete with a high probability of success.
—Selects backlog items from the product backlog based on their velocity.
—Defines acceptance criteria that will eventually become systems tests.
—Determines how it will build the selected items by decomposing each selected backlog item into tasks.
—Coordinates with the Product Owner on sprint goals and helps define "Done."
—Decides the location and time for the daily stand-ups.

The Scrum Master prepares the Sprint Backlog to illustrate the time it will take to complete the work the team must complete.

Sprint Review: A sprint review is conducted at the end of every sprint, and it is facilitated by the Scrum Master. Each team member is required to demonstrate completed functionality and

review all work products completed. Work not completed cannot be demonstrated but should be mentioned. The review must include all relevant stakeholders so that the team will receive critically needed feedback. At the conclusion of the review, the Product Owner decides whether to accept everything as presented or to defer the completed work until the product's functionality becomes useful enough to be deployed or shipped.

Sprint Retrospective: The purpose of the retrospective is to build the team's commitment and transfer knowledge learned to the next sprint and to other teams. The sprint's retrospective is a time-boxed meeting conducted by members of the team at the conclusion of every sprint to accomplish the following:
   —Discuss what was successful about the sprint or release.
   —Determine what could or must be improved before proceeding.
   —Learn from experiences and plan for the future.
   —Identify successes and improvements for future projects.
   —Discover, share, and pass along the learning experience.
   —Make changes for the next sprint.
A retrospective is not a:
   —Session to identify mistakes and place personal blame.
   —Place for personal attacks.
   —Meeting to resolve issues.
   —Planning meeting.

Scrum of Scrums

According to the Scrum Alliance, a Scrum of Scrums (SoS) is a technique used to scale Scrum up to large groups of 12 or more people. Each daily stand-up within a sub-team ends by designating one member as "ambassador" to participate in a daily meeting with ambassadors from other teams. [6]

In my experience, I have found that the Scrum of Scrums is needed for larger projects that involve more than three development teams consisting of 10 or more members. Large teams should be divided into smaller teams that focus on a subset of the end product.

The Scrum of Scrums should be held two or three times a week depending on project complexities or hurdles after the daily stand-up. Scrum of Scrums should be attended by the Scrum Master of each team or a responsible team member selected by the team. This session should be facilitated by an Agile coach or a very experienced Scrum Master to ensure discussions remain focused. The meeting is not time-boxed like other Scrum meetings because it does not take away from any committed time that team members have agreed to work. Specific guidelines for the Scrum of Scrums include:
   —Allowing clusters of teams to discuss their work, focusing especially on areas of overlap and integration.
   —The agenda is the similar to the daily stand-up:
   —What has your team done since we last met?
   —What will your team do before we meet again?
   —Is anything slowing your team down or getting in their way?

Conclusion

The title of this article is, "Scrum is Simple," but Scrum is not easy. Scrum implementation represents a significant cul-tural shift from traditional project management methods. The Scrum roles are different from traditional roles and frequently cause confusion to general engineering and business communities. Scrum terminology is foreign to most, and change from the 'status quo' is not only difficult but also initially resisted by most. Scrum is best used as a wrapper of an organization's existing engineering practices and is improved as necessary while product increments are delivered or deployed. Scrum provides a way for team members to feel good about their jobs, their contributions, and their performances.

REFERENCES

1. http://www.scrumalliance.org/
2. http://www.pmi.org/
3. http://www.pmi.org/PMBOK-Guide-and-Standards.aspx
4. http://hbr.org/1986/01/the-new-new-product-development-game/ar/1
5. http://www.agilealliance.org/
6. https://www.agilealliance.org/glossary/scrum-of-scrums/

Recommended Reading:
—"Lean Software Development: An Agile Toolkit for Software Development Managers." Mary and Tom Poppendieck, 2003.
—"Agile Project Management with Scrum." Ken Schwaber, 2004.
—"Writing the Product Backlog Just Enough and Just in Time." Mike Cohn, 2009.
—"Succeeding with Agile: Software Development Using Scrum." Mike Cohn, 2009.

ABOUT THE AUTHOR

Dick Carlson has a B.S. degree in business management and is certified as a Scrum Professional, Scrum Master, Scrum Product Owner, and in Lean-Agile Project Management. He has shared successful experiences of Agile, Lean, and Scrum implementations at conferences, workshops, and symposia. Dick's engineering career spans 50 years, and he has taught courses in mathematics, electronics, CMMI, configuration and data management, Agile, Lean, and Scrum for more than 30 years.

Mr. Soukup holds a Bachelor's and Master's degree in Electrical Engineering, a Juris Doctor in Law, certificates for software development, project management, functional management, systems engineering, and Agile and Lean including being a Certified Scrum Master. He was a development and test engineer, manager, and project manager for both hardware and software. Also he developed and taught courses in mathematics, electronics, ethics, Lean, and Agile. He is an accomplished analyst.