

Scrum for Hardware and System Design
for Machine Design Magazine
David G. Ullman and Joshua Tarbutton

You have probably heard of “Scrum”, the Agile framework for software design, and wondered: Should I be using Scrum to design hardware and systems? This article briefly introduces Scrum, describes how it is changing software design and gives enough details so you can decide if you want to explore using it in your organization.

All Agile methods are an effort to enhance the efficiency of the design process and manage the uncertainty that is part of all design efforts. For many years the software industry had a bad reputation of delivering product late (if at all) and with flawed functionality. In the late 1990s Jeff Sutherland developed the Scrum methodology to help improve the software development process. His Scrum method gained popularity in software design through the 2000s and has more recently been adopted for hardware systems.

The software world had a great incentive to improve its design process. Consultants such as the Standish group do annual surveys focused on IT project success and publish the results in the Chaos Report¹. One recent result found that 31.1% of all IT projects in large organizations are cancelled before they ever get completed. Another 61.2% of the projects are “challenged” meaning that they only meet two of the three project success measures: cost, schedule or scope. This means that only 9% of all projects are successful in meeting all three measures. Similar, but less draconian results are found for small organizations.

Another consulting company examined a different group of organizations and found that software projects using Agile are nearly twice as likely to succeed than those using the traditional waterfall or sequential methods.² Further, 80% of those who report doing Agile are doing Scrum or a hybrid of Scrum.

The 2019 book **Scrum for Hardware Design** is available both as an ebook and hardcopy at www.mechdesignprocess.com/scrum. This book includes details of how to apply scrum to hardware problems and two case studies: the design of the Saab Gripen fighter and a student project to develop a prosthetic arm. This book is a supplement to **The Mechanical Design Process** 6th edition and **Case Studies for the Mechanical Design Process** also by David G. Ullman.

Method	Successful	Challenged	Failed
Agile	42%	50%	8%
Waterfall	26%	53%	21%

¹ The Standish Group Report “Chaos”, 2018

² How to Transition from Waterfall to Agile: A white paper, Vitality Chicago, 2019, <https://vitalitychicago.com/blog/how-successfully-transition-waterfall-to-scrum/>

The top reasons organizations decided to move to Agile methods and the benefits they realized are listed in the table below³.

Measures	Reason for adoption Agile	Benefits of adopting Agile
Accelerate product delivery	75%	61%
Manage change	64%	71%
Increase productivity	55%	62%
Better business/engineering alignment	49%	65%
Increased product quality	46%	47%
Enhanced delivery predictability	46%	49%
Improve project visibility	42%	66%
Reduce project risk	37%	47%
Improve team morale	28%	61%

What we can conclude from these surveys is:

- The software industry has a very high percentage of failed or challenged projects. There are no equivalent surveys for hardware/systems projects but there is no reason to believe that the failure/challenged rates are much different.
- The reasons software organizations try Scrum apply to hardware/systems as much as software, and the benefits should also be similar. Early adopters such as: Tesla, John Deere, Saab Aerospace, Raytheon, Oak Ridge National Labs, Bosch, SpaceX do report such benefits, just there are no surveys or statistics yet.
- Using Scrum nearly doubles the chance for software project success. There is anecdotal evidence that it can have similar impact of hardware/systems projects.

What is Scrum

Before describing the process, note that, like every process, Scrum has its own jargon.

Each Scrum-speak term used here is capitalized on its first usage. Only the minimum set of terms are used, there are more. Further, this is a very brief introduction, for details and case studies see *Scrum for Hardware Design*. One of the case studies in this book is about the team responsible for the oxygen system in Saab's JAS 39 Gripen E, the Swedish strike fighter. This fighter is faster than the Lockheed-Martin F35



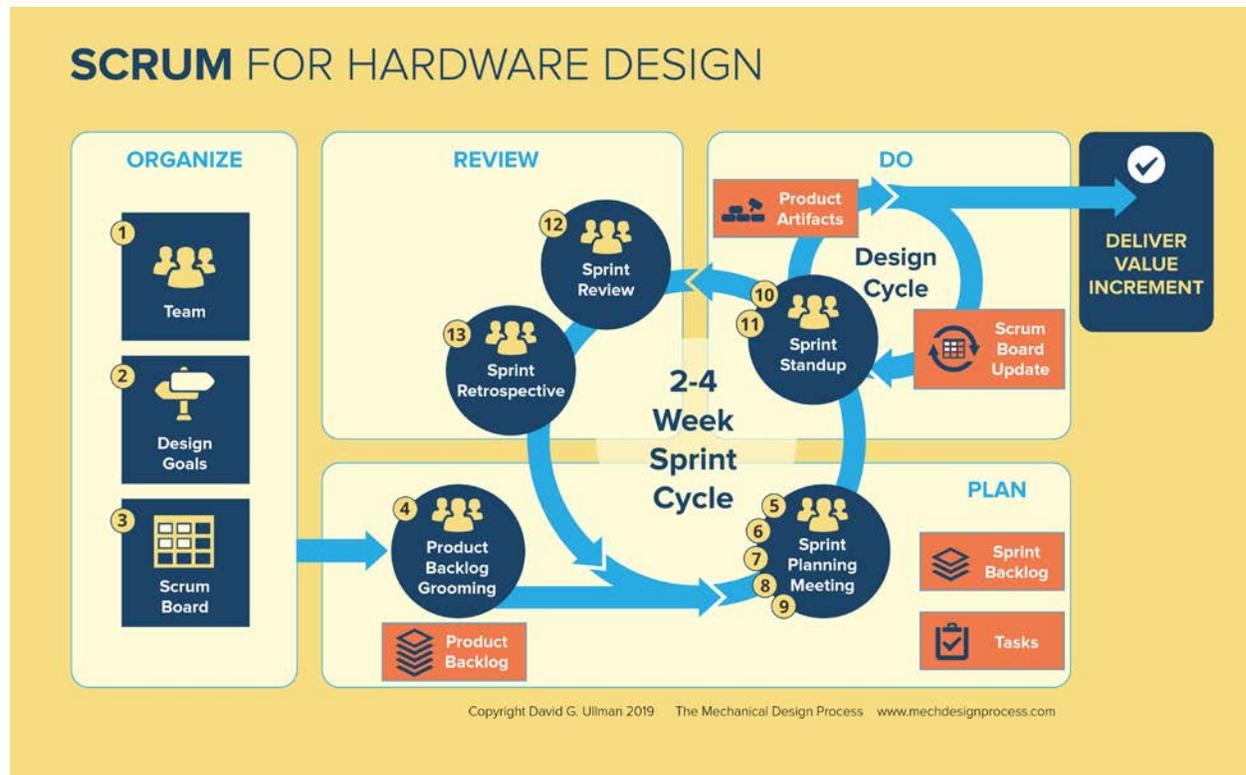
Lightning, has similar combat range, but is not stealth. Its initial cost is about one half of

³ 12th State of Agile Report, Colab.net Versionone.com, <http://stateofagile.versionone.com/>

⁴ Software to manage Scrum Boards include Jira, Trello, and many others. These are often referred to as "Kanban tools".

that of the F35 and its operating costs are about 25% of that for the F35. These cost benefits are attributed in part to the use of the Scrum process. Examples from this case study are included here.

The structure of Scrum is shown in an infographic. Scrum is built around 2 – 4-week design efforts called Sprints, the heart of the graphic. This slicing of the design process into short increments allows better planning, communication and incremental design maturation. Volvo’s truck division uses 2-week sprints, Saab Aerospace uses 3-week sprints in all its divisions.



Each sprint is built on a plan-do-review cycle as shown by the light-yellow areas. Every 2 – 4 weeks, the tasks for the sprint are planned, the tasks are done and the results and the process reviewed. Then the process is repeated for a new sprint and a new set of tasks. Teams at Saab have been working in 3-week cycles for many years.

The dark blue circles in the infographic represent the five types of sprint meetings. While the number and types of meetings may seem excessive, they are structured so that they actually save time as they keep communication open, brief and on-point. Typically, these meetings take 10%-15% of each team member’s time. This leaves 85% -90% of their time to actually get work done.

The orange boxes are Artifacts - information or physical objects generated. The numbered circles indicate 13 distinct activities that make it all happen each of which will be briefly introduced here and noted by the step number in brackets, (X).

Prior to the sprints is the need to organize – build the team and establish high level design goals and specifications. Scrum teams (1) have 4-9 members who carry out all the design tasks: plan, conceive, analyze, build, test, decide, and document; and they self-manage the process. Scrum teams are stable: some at Saab have been together for over ten years.

The design goals need to be established (2). In the ideal software world, this upfront, macro-planning is minimized. However, for hardware and systems this is essential just as in traditional design processes. Here the planning focuses on the engineering specifications that need to be met and not so much on the timing. There are no detailed Gantt charts in Scrum with all the detail planning pushed down to the sprint and daily level as will become clear. In Scrum-speak, the design goals constitute the Product Backlog. This is a list of all the design requirements that will need to be resolved during the design of the product.

One feature common to most Agile methods, but totally foreign to traditional methods is the Scrum Board (3). This is usually physical board with sticky notes on it as seen with the Saab Gripen oxygen system design team. The scrum board can alternatively be managed in one of the many software packages available⁴. On a scrum board the product backlog items (the design requirements) are written on sticky-notes and posted on the left edge. As they are resolved they will be moved to the right (see the description of activities 8 and 11, below). In software design these goals are often called “stories” as they are the stories customers tell to describe the functionality they desire. For hardware, customer stories need to be supplemented with methods like Quality Function Deployment (QFD)⁵ to get a complete picture of the goals. QFD is a method for capturing the voice of the customer and translating it into engineering specifications.



Sprint planning begins with the product backlog. In the first of the five types of meetings, Product Backlog Grooming (4), the design goals are rank-ordered based on dependency, uncertainty, and importance. This gives the team a long-term vision of all the that needs to be accomplished during the design of the product and what is most important to do first.

⁴ Software to manage Scrum Boards include Jira, Trello, and many others. These are often referred to as “Kanban tools”.

⁵ See Chapter 6 of *The Mechanical Design Process* 6th edition, by David G. Ullman.

A second meeting, the Sprint Planning Meeting actually kicks off each sprint. Here the team chooses which goals to address during the sprint from the product backlog (5). They only choose those they can complete during the 2-4-week sprint period. This selection process is iterative as will be shown.

During sprint planning the team identifies the tasks, what needs to be done to meet the highest ranked goals. Each task is complete with measures, targets and tests that define when it is done (6). Defining tests up front is often referred to as “test-driven development”. In software design, the ideal is to have something the customer can test at the end of every sprint. It is generally understood that this ideal is not met by most software projects. Clearly, it is unrealistic to expect testable hardware at this cadence. More realistically the tasks are iterations where during the first sprint the “test” may be the number of concepts developed, in the second sprint the “test” may be the selection of the best concept, a solid model may be the “test” in the third, and so on.

There are three important actions happening here: First, each goal is broken down into the tasks needed to meet it. Second, as part of the task description, there must be something that can be tested at the end of each iteration. The tasks worked on during each sprint must deliver something that measures that it is done. Third, there is a tight interaction with the customer. Instead of periodic design reviews at each gate, there is a mini design review at the end of each sprint (the Sprint Review (12)). This increased interaction with the customer is very important to project success.

The team next estimates the time each task will take (7). Part of the Scrum philosophy is that a task not completed during a sprint might as well have not been started. This forces the decomposition of the design effort into bite-sized tasks.

Still part of the sprint planning meeting, the team now chooses which tasks to tackle during the sprint (8). These tasks are posted on the scrum board and are referred to as the Sprint Backlog. They are written on sticky notes that are posted in the “ToDo” section of the scrum board, just to the right of the product backlog.

The final planning step is to align team members with the tasks (9). The Technical Team self-organizes to best apply the members’ technical skills to the tasks. Notice that the word “align” is used here and not the word “assign.” In highly technical fields the tasks may be assigned to individual team members with sufficient expertise to resolve the issue. But often, there are many team members who can do the task. This “planning-in-the-small” makes the best use of each person’s time.

The “Do” part of the scrum is drawn as a smaller cycle in the infographic as it has a daily rhythm. Here the technical work is done and reported to the team during a daily 15-minute stand up meeting (10). Here each team member answers three questions:

- What did I do yesterday to help finish the sprint?
- What will I do today to help finish the sprint?
- What obstacles are blocking the team or me from achieving the sprint goals?

The goal of these questions is not to judge anyone’s work, but to find what is and is not going well. If there are obstacles, then the team can address them so the task goals can be met by the end of the sprint. This meeting is “planning in the small.” Here the team re-plans their sprint every day as needed to complete the tasks during the sprint period.

During the design cycle as a task is being worked on, the scrum board is updated by moving its sticky-note from “ToDo” to “Doing” and then to “Done” (11). In this way the status of the sprint is always evident.

At the end of the sprint, there two types of meetings; a Sprint Review (12) and a Sprint Retrospective (13). The sprint review is much like a typical design review; however, it is limited to one hour. It need not be longer as it is held every 2-4 weeks.

During the sprint retrospective the team focuses on the sprint process itself identifying what worked well and what could be done better. This is a strong feature of the Scrum method - frequent reflection and improvement of the work methods and tools used to support the work.

In the software industry, not all teams do all of the thirteen activities. Those most used are: daily standup, 90%; sprint/iteration planning, 88%; retrospectives, 85%; sprint review, 80%; and 2-4-week iterations,69%.

The Challenge for Hardware Design

There is much potential in implementing Scrum for hardware. Companies like Saab have reduced bureaucracy and encouraged decision making at the lowest possible level in the project organization and they have delivered an aircraft for lower cost, with more advanced functions, and higher quality.

On a personal level: each Saab team and each team member knows what is expected of them; they know what is desired, not required, as that would stifle creativity; they make decisions and feel enfranchised, and there is clear communication between individuals and teams leading to product quality.

To get there requires both organizational and design process changes.

Saab’s results are impressive and promising for other organizations, but in moving to Scrum is not easy. Statistics on resistance to moving to Scrum in software organizations shows the need for organizational flexibility, and skill development through training and experience. Planning and decision making are moved down in the

Type of resistance to moving to Scrum	Percent of orgs. experiencing
Organizational culture at odds with agile	53%
Organization resistant to change	46%
Inadequate management support	42%
Lack of skills with methods	41%
Lack of training	35%

organization changing the control structure, so it is not surprising that the organization resists.

This short article has just scratched the surface of what it takes to move to scrum; study, training and experience is needed to make it happen. It is strongly suggested that you get help getting started. Just be sure that the help has experience in hardware design because **software design and hardware design are not the same**. Many scrum software trainers think that because they have had success in software organizations, this mirrors to hardware. It does not! There is not room here to explore the differences and how they affect Scrum, but these differences are detailed in the book ***Scrum for Hardware Design*** (see the box for details).

In spite of the organizational, training and discipline challenges, there is no reason to believe that hardware design organizations cannot experience similar dramatic improvements that have been realized in software development.

Author bio

David G. Ullman is an active product design and has trained over 5000 students how to develop products from need to delivery. In 1992 he published “The Mechanical Design Process” a text now in its 6th edition that is used internationally and has been translated into Chinese and Korean. It is a compendium of best design practices. In 2019 he published “Scrum for Hardware Design” bringing new design best practices to students.

Dr. Ullman holds a PhD. From The Ohio State University, is an ASME Life Fellow and Emeritus Professor from Oregon State University. He is a DTM founder. He holds six patents, and is an active product designer, aerospace engineer and product design consultant. He certified is scrum. He lives at an airpark where he designs, builds and flies airplanes.

Website: www.davidullman.com

Book website www.mechdesignprocess.com

Joshua Tarbutton

Dr. Tarbutton’s research is focused on exploring how innovative geometrical and process solutions can solve bottlenecks that exist between design and manufacturing. His primary focus is on how tool path generation in the 3D scanning, reverse engineering and machining processes can be automated to increase productivity during manufacturing of discrete parts. He also is investigating additive manufacturing of piezoelectric polymers and was the first researcher to create piezoelectric polymers during 3D printing using the EPAM method he invented. Dr. Tarbutton received a B.S. degree in Mechanical Engineering from the Georgia Institute of Technology in 2005. He received his M.S. degree and his PhD both in mechanical engineering from Clemson University in 2007 and 2011 respectively. He spent a year at IST Precision as an industry postdoctoral fellow under the ASEE-NSF Small Business Research Diversity Fellowship. He took a post as an assistant professor at USC in Columbia, SC in 2012

where he remained until 2016 when he moved to UNCC in Charlotte, NC. He has won two teaching awards, published 50 papers, holds a patent in displacement metrology and is an Army Veteran. He owns Bravo Team, an engineering design and product development firm of 7 engineering wizards who spend their days helping people make things. He is a certified SCRUM Master and practices SCRUM with his research group and his senior design teams at UNCC and his staff of wizards at Bravo Team LLC.