SOFTWARE IN 30 DAYS

How Agile Managers Beat the Odds, Delight Their Customers, and Leave Competitors in the Dust

KEN SCHWABER and JEFF SUTHERLAND
Creators of SCRUM
SOFTWARE IN 30 DAYS
SOFTWARE IN 30 DAYS

How Agile Managers Beat the Odds, Delight Their Customers, and Leave Competitors In the Dust

KEN SCHWABER and JEFF SUTHERLAND

John Wiley & Sons, Inc.
To Ikujiro Nonaka, Babatunde A. Ogunnaike, and Hirotaka Takeuchi for their inspiration and guidance.
Contents

About the Authors xi
Acknowledgments xiii
Introduction xv

Section I Why Every Business in the World Can Produce Software in 30 Days 1

You are probably frustrated with your software organization. You would like it to be quicker, more flexible, understand your needs better, and help you become more profitable. We look at why you are frustrated and how to fix the problem.

1 The Crisis in Software: The Wrong Process Produces the Wrong Results 3

Many software organizations follow a development process that guarantees waste, uncontrolled risk, unpredictability, surprises, and low value. We will investigate why this process was chosen, how it guarantees failure, and look and some organizations that have recovered from it.

2 Scrum: The Right Process Produces the Right Results 17

There is a process that is appropriate for software development. When you get your developers to use it, you will immediately gain productivity, quality, value, control, predictability, and satisfaction. We look at how this happens in this chapter.
3 Try It Yourself: The Pilot
You have read our assertion that there is a better way for you to get software developed for you. However, a lot of people have made assertions and taken a lot of your money in the past, with little or no improvement. In this chapter we show you how to prove that our approach works for no money.

4 What Can I Do?
You learned how to do better and you’ve tried it yourself. You like the results and you know what to tell the software organization to do. In this chapter, we look at what you can do to help what you experience in the pilot project succeed.

Section II How to Produce Software in 30 Days
Having better software developed for your needs is not so much hard as it is different from what you are used to. In this section, we look at a progressively beneficial set of approaches to get you from where you are now to organizational agility.

5 Getting Started with Scrum
Our secret sauce for improving your benefits from software is called “Scrum.” Yes, this is the rugby event that keeps the ball moving down the field. We’ll discuss Scrum, how it works, and why it works in this chapter.

6 Scrum at the Project Level
Most persistent improvement in software development starts at the project level. You can use Scrum to further prove its utility, or on critically important initiative that must succeed. We’ll explore what you can tell your developers to do after reading this chapter.

7 Develop a Scrum Capability
Success often breeds success. As more software initiatives using Scrum succeed, more people will want to get on the wagon. Rather than changing the entire organization, let’s look at how we can set up a software development universe separate from the disappointing, existing department. You can increasingly reap benefits here on an increasing number of projects and releases.
8 Scrum at the Enterprise Level

Scrum at a project or release level provides initiative level agility, the ability to rapidly respond to opportunities or rise to challenges. To gain the most significant benefits, Scrum's empirical approach to software development must be fit into the organization as a whole. We’ll look at how to do this, and why some approaches are short-lived and others persist.

9 Enterprise Transformation: Profound and Persistent Change

You want to make your organization leaner, more efficient, and agile on your watch. Even more, you want these benefits and their underlying causes to persist and become the organizational culture. We’ll look at an enterprise change approach for achieving this in this chapter.

10 Scrumming Scrum

We devised Scrum for complex problem solving, like software development. We found Scrum a useful technique for managing organizational change, also a complex problem. The same benefits of transparency, waste removal, risk control, and predictability occurred. We’ll look at this use of Scrum in this chapter.

Appendix 1: Terminology

We slowing and progressively introduced some new terminology. This appendix is your reference for those terms.

Appendix 2: The Scrum Guide

Read the canonical guide to Scrum, its roles, artifacts, and events. This is the bible of Scrum.

Appendix 3: A Playbook for Achieving Enterprise Agility

This appendix presents a more detailed plan for enterprise change, as discussed in Chapter 10.

Index
Jeff Sutherland and Ken Schwaber are the creators of Scrum, a software development process that delivers software functionality in 30-day increments. Scrum was born when Jeff and Ken presented a paper at the OOPSLA conference in Austin, Texas, in August 1995. This paper, “Scrum Development Process,” was the result of their collaboration prior to that point. The works of H. Takeuchio and I. Nonaka in their seminal works on lean knowledge creation, bottom-up intelligence, and teamwork had profoundly influenced Jeff. Babatunde Ogunnike had profoundly influenced Ken in his work on industrial process control and the applicability of complexity theory and empiricism to software development.

In addition to being Scrum’s creators, Jeff and Ken have also served as its wards. With their guidance, Scrum has evolved over time; more recently, they have developed ways to speed up Scrum’s systematic evolution based on community experience and input. In “The Scrum Guide,” found in Appendix 2 of this book, Jeff and Ken offer the complete definition of Scrum.

Dr. Jeff Sutherland is the chief executive officer of Scrum Inc., in Cambridge, Massachusetts, offering training, guidance, and coaching to companies across the globe. Jeff is a distinguished graduate of the United States Military Academy and a Top Gun of his USAF RF-4C Aircraft Commander class. Jeff has advanced degrees from Stanford University and a PhD from the University of
Colorado School of Medicine. He is also a senior advisor to OpenView Venture Partners, helping them implement Scrum and agile practices in all their portfolio companies. Jeff has extended and enhanced Scrum at many software companies and information technology (IT) organizations over the years.

Ken Schwaber is a software development professional, having spent the past 40 years of his life as a programmer, analyst, consultant, product manager, and business owner. Early in his career, Ken tried unsuccessfully to make waterfall software projects successful; he later developed an alternative to waterfall. Ken has spent the past 20 years developing Scrum and working with organizations around world to help them take advantage of it. Ken is one of the original signatories of the Agile Manifesto and the founder of the Agile Alliance and the Scrum Alliance. He is currently working to improve the software profession through Scrum.org. Ken and his wife, Christina, live in the Boston area. He is a graduate of the United States Merchant Marine Academy and has completed additional study in computer science at the University of Chicago and in business at the University of California at Los Angeles Anderson School of Management.
Acknowledgments

This book would not be what it is without the excellent copyediting of Arlette Ballew, the overall direction of Richard Narramore, and the laser focus of Carey Armstrong.