

Extracted from:

# Fixing Your Scrum

Practical Solutions to Common Scrum Problems

This PDF file contains pages extracted from *Fixing Your Scrum*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

Copyright © 2020 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The Pragmatic Bookshelf

Raleigh, North Carolina

The  
Pragmatic  
Programmers

# Fixing Your Scrum

Practical Solutions to  
Common Scrum Problems



Ryan Ripley  
Todd Miller

*edited by Dawn Schanafelt*



# Fixing Your Scrum

Practical Solutions to Common Scrum Problems

Ryan Ripley

Todd Miller

The Pragmatic Bookshelf

Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic books, screencasts, and audio books can help you and your team create better software and have more fun. Visit us at <https://pragprog.com>.

The team that produced this book includes:

Publisher: Andy Hunt  
VP of Operations: Janet Furlow  
Executive Editor: Dave Rankin  
Development Editor: Dawn Schanafelt  
Copy Editor: Sean Dennis  
Indexing: Potomac Indexing, LLC  
Layout: Gilson Graphics

For sales, volume licensing, and support, please contact [support@pragprog.com](mailto:support@pragprog.com).

For international rights, please contact [rights@pragprog.com](mailto:rights@pragprog.com).

Copyright © 2020 The Pragmatic Programmers, LLC.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

ISBN-13: 978-1-68050-697-6

Encoded using the finest acid-free high-entropy binary digits.

Book version: P1.0—January 2020

*To my wife, Kristin. You make it all worth it.*

*—Ryan*

*To Jess, who makes the impossible possible  
for me.*

*—Todd*



# Preface

---

We're passionate about Scrum. In fact, we've both dedicated our lives to improving the profession of software delivery through training, coaching, and mentoring Scrum teams. In the pages that follow, you'll find strong opinions, new ideas, and a solid grounding in good Scrum practices. *Some of the words might sting a bit.* You may find an anti-pattern or two that your team is currently performing. Embrace empiricism—be transparent and honestly inspect what you are doing today. Then make the adaptations needed to get back to good Scrum practices. Don't worry, this book will help you. We've loaded it with tips, tricks, and techniques to get your Scrum teams moving forward again.

We're aware that Scrum is no longer the new and trendy framework that it used to be. In fact after over 20 years of doing Scrum, teams all over the world, companies of many sizes, and developers in particular have become jaded—and perhaps rightly so.

Bad Scrum is rampant in the software world. In many organizations, velocity is used as weapon, not a planning tool. Scope is still committed to, not managed. Product visions are unclear and value is not the center of conversations. Delivery seems like a magical event where the outcome of putting features in front of customers feels like a game of chance, not a strategic decision. Scrum masters preach self-organization, but then assign the work to the development team. Throughout this book, you'll find info about combating these anti-patterns and many others.

Worse is when companies rename their old practices with shiny new Scrum labels and immediately expect twice the work in half the time. Yet, the old patterns persist, and the empowerment promise made to developers and the commitment to quality turn out to be lies. In these circumstances, Scrum is just a bait-and-switch tactic.

So we get why some teams are upset.

Many organizations aren't getting the full benefits that Scrum—executed well—can offer. They may check a box after each Scrum event and go through the motions of Scrum. But this kind of “mechanical” Scrum is Scrum in name only, where teams use the Scrum framework without truly embodying its principles and values. Mechanical Scrum quickly degrades and becomes obsolete. Trust us, we've seen it happen.

You can rise above this by truly embracing Scrum. When you embrace self-organization, live the Scrum principles and values, and emphasize technical excellence, you empower your team to get products into the hands of customers sooner, and confirm that what the team is building is what the market actually wants. Isn't it better to invest in building products that people actually want to use?

If you want to use this wonderful framework to deliver valuable, high-quality features every sprint, read on.

## Who This Book Is For

This book assumes that you're currently on a Scrum team. In Chapter 2, we provide a very brief introduction to Scrum, but after that we jump straight into the framework. Experience will help you get the most out of this book, but you can also use it to bootstrap your Scrum knowledge and get started down the path of good Scrum.

We wrote this book primarily for Scrum masters. If you're a new Scrum master, we help you avoid common traps that are easy to fall into, and give you the tools to understand the “why” behind the deeper aspects of the Scrum framework. If you're an experienced Scrum master, we provide practical tips and concrete suggestions that can help you deepen your understanding of how Scrum works—or *could* work if your organization fully embraces the Scrum principles and values.

If you manage people who are on a Scrum team, there's lots of useful info here for you, too. We present real-world examples of bad situations and provide explanations of how we've helped those teams succeed.

## What's in this Book

In each chapter (starting with Chapter 3), we run through several common anti-patterns related to the topic of that chapter, and suggest ways to prevent or fix that anti-pattern on your team. At the end of each chapter is a section titled “Coach's Corner,” where we provide an exercise you can try with your

---

### Scrum at Scale

---



Although some of our stories and the techniques in this book are about or can be applied at scale, this book doesn't focus on ways to help you scale Scrum in environments where several development teams are working on one product. If you're considering scaling Scrum, we recommend getting really good at Scrum at a small level first, and then thinking long and hard about whether scaling is truly necessary.

team that applies to all the anti-patterns in that chapter, and which can help you narrow down the issues you're dealing with.

In [Chapter 1, A Brief Introduction to Scrum, on page ?](#), we provide some sage advice on how to learn and explore the Scrum framework. This will give you a great refresher on Scrum and will prepare you for the upcoming chapters.

When Scrum goes bad, it isn't just the product that suffers. In [Chapter 2, Why Scrum Goes Bad, on page ?](#), you'll see what happens when trust is lacking in an organization and empiricism isn't safe. Spoiler: Scrum doesn't thrive in this kind of environment.

But take heart! In [Chapter 3, Breaking Bad Scrum with a Value-Driven Approach, on page ?](#) we'll explore the Scrum values. These values can promote the characteristics and behaviors that a Scrum team needs to overcome organizational dysfunction and achieve high performance. These values are really, really important. You may want to read this chapter twice.

As the “mini-CEO” of the product, the product owner plays a fundamental role in defining the vision, assessing the value, and validating the success (or failure) of the product in the marketplace. In [Chapter 4, The Product Owner, on page ?](#), we'll look at what happens when the product owner role is misinterpreted and/or poorly performed.

Do you have a future vision of your product? If not, we'll show you how to build one in [Chapter 5, The Product Backlog, on page ?](#). The product backlog is where Scrum teams store anything and everything that they may need for their products in the future. It's the single source of truth for the Scrum team, who uses it to plan, forecast, roadmap, and guide the direction of the product.

[Chapter 6, The Development Team, on page ?](#) is dedicated to the folks who develop the product. The development team is a cross-functional, self-organizing team of people who can turn product backlog items into working

software. Closely examine the anti-patterns we discuss in this chapter. A dysfunctional development team doesn't often deliver product.

A Scrum master must ensure that Scrum is well-understood and enacted by the Scrum team. [Chapter 7, Embracing the Scrum Master Role, on page ?](#) explores what happens when the Scrum master doesn't act as a servant leader. Honestly, many of the anti-patterns in this book can be tied directly to a weak Scrum master. Solve these anti-patterns quickly.

Management is not the enemy. In [Chapter 8, Management, on page ?](#), we discuss how important it is for a Scrum master to have empathy for leadership and suggest meaningful and effective ways to work with middle managers. Get this right and many of your organizational impediments can be resolved much faster than previously possible.

The sprint is often the overlooked and underappreciated Scrum event. By the end of [Chapter 9, Thinking in Sprints, on page ?](#) you'll have a deeper understanding of the rules of the sprint and what can happen when Scrum teams violate these rules.

Sprint Planning is a complex event that can go wrong in many different ways. [Chapter 10, Sprint Planning, on page ?](#) examines common missteps during sprint planning, including skipping product backlog refinement during the sprint. This chapter could save you hours of wasted time if you're able to adapt the way you approach this important planning event.

The development team owns the sprint backlog. If that doesn't ring true with you, [Chapter 11, The Sprint Backlog, on page ?](#) will help change your mind. Commitment is also a big topic we'll discuss as we explore scope issues, progress, and being transparent about work during a sprint.

Did you know that you don't have to stand up during the daily scrum? This and many more earth-shattering revelations come to light in [Chapter 12, Reclaiming the Daily Scrum, on page ?](#).

[Chapter 13, Deconstructing the Done Product Increment, on page ?](#) covers a lot of ground. We examine the pitfalls of not having a definition of "done," why "visible" doesn't mean "transparent," and what to do when Scrum teams and the organization don't adopt the concept of done.

Did you know that the sprint review isn't a demo? [Chapter 14, The Sprint Review, on page ?](#) covers the anti-patterns that come out when the sprint review isn't properly implemented or is facilitated poorly. Learn how to engage your stakeholders, collaborate with your customers, and get the info you need

to update your product backlog and start the next sprint with the best information possible.

If you're using the "What Went Well," "What Didn't Go Well," and "What Do We Need to Change" format—STOP. Read [Chapter 15, The Sprint Retrospective, on page ?](#) and update your retrospective practices. Continuous improvement is essential to good Scrum. A well-executed sprint retrospective generates the insights a Scrum team needs to continuously improve their process, practices, and interactions.

## How to Read This Book

It would be odd for two Scrum trainers to tell you what to do. So we'll simply say, "It depends."

Just kidding.

Feel free to jump to any chapter in the book. If you're looking to learn about Scrum and the many anti-patterns that teams perform, we suggest going through the chapters in order. However, if there's an immediate problem that you want to work on, turn to the chapter related to the event, artifact, or role that contains your anti-pattern and let the improvements begin.

## Online Resources

This book has its own web page<sup>1</sup> where you can find more information about it. Help improve this book by reporting errata, including content suggestions and typos.

Let's begin.

---

1. <https://pragprog.com/rscrum>