

Extracted from:

Agile Retrospectives

Making Good Teams Great

This PDF file contains pages extracted from Agile Retrospectives, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragmaticprogrammer.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 printer versions; the content is otherwise identical.

Copyright © 2005 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.

Contents

Foreword	xiii
Preface	xv
Introduction	xviii
1 Helping Your Team Inspect and Adapt	1
1.1 Set the Stage	5
1.2 Gather Data	8
1.3 Generate Insights	11
1.4 Decide What to Do	11
1.5 Close the Retrospective	13
2 A Retrospective Custom-Fit to Your Team	15
2.1 Learning About the History and Environment	15
2.2 Shaping the Goal for the Retrospective	16
2.3 Determining Duration	17
2.4 Structuring a Retrospective	19
2.5 Selecting Activities	22
3 Leading Retrospectives	28
3.1 Managing Activities	29
3.2 Managing Group Dynamics	31
3.3 Managing Time	36
3.4 Managing You	37
3.5 Taking Your Skills to the Next Level	38
4 Activities to Set the Stage	40
4.1 Check-In	41
4.2 Focus On/Focus Off	43
4.3 ESVP	45
4.4 Working Agreements	48

5	Activities to Gather Data	50
5.1	Timeline	51
5.2	Triple Nickels	56
5.3	Color Code Dots	59
5.4	Mad Sad Glad	61
5.5	Locate Strengths	63
5.6	Satisfaction Histogram	66
5.7	Team Radar	71
5.8	Like to Like	74
6	Activities to Generate Insights	77
6.1	Brainstorming/Filtering	78
6.2	Force Field Analysis	81
6.3	Five Whys	85
6.4	Fishbone	87
6.5	Patterns and Shifts	90
6.6	Prioritize with Dots	92
6.7	Report Out with Synthesis	95
6.8	Identify Themes	97
6.9	Learning Matrix	99
7	Activities to Decide What to Do	102
7.1	Retrospective Planning Game	103
7.2	SMART Goals	107
7.3	Circle of Questions	109
7.4	Short Subjects	111
8	Activities to Close the Retrospective	113
8.1	+ /Delta	114
8.2	Appreciations	117
8.3	Temperature Reading	119
8.4	Helped, Hindered, Hypothesis	122
8.5	Return on Time Invested (ROTI)	124
9	Releases and Project Retrospectives	127
9.1	Preparing for Release and Project Retrospectives	128
9.2	Including Cross-Organizational Perspectives	133
9.3	Leading Release and Project Retrospectives	135
9.4	A Retrospective at Every Ending	141

10 Make It So	142
10.1 Provide Support	143
10.2 Share Responsibility for Making Changes	145
10.3 Supporting Larger Changes	145
A Facilitation Supplies	149
B Debriefing Activities	152
C Activities Quick Reference Matrix	154
D Resources for Learning Facilitation Skills	156
E Bibliography	157

Introduction

Suppose you are a member of a software development team. You're doing good work, but not great work. You're starting to see signs of interpersonal friction on the team, and some people you would like to retain on the team are dusting off their résumés. You know you need to adapt your practices and ease the interpersonal tension before things get worse. You want to introduce retrospectives to your team.

Maybe you are a team lead, and you've heard about retrospectives but have never tried one. You've heard retrospectives can help teams perform better, but you're not sure where to start.

Maybe you've been holding retrospectives for months, and your team isn't coming up with any new ideas. You need a way revitalize your retrospectives so the team doesn't lose the gains they've made.

Whatever the reason you've picked up this book, we assume you think retrospectives might help your team. Whether you're a coach, a team member, or a project manager and whether you're expected to lead retrospectives after every iteration or are initiating retrospectives for the first time, you'll find ideas and techniques that you can apply to your situation.

Our main focus in this book is short retrospectives—retrospectives that occur after one week to one month of work. Whether you are using Agile methods or more traditional incremental or iterative development, your team has an opportunity to reflect at the end of every increment and identify changes and improvements that will increase the quality of the product and the work life of team members.

Retrospectives are a natural fit in an Agile work environment—Scrum and Crystal explicitly include “inspect and adapt” cycles for the methods and teamwork along with mechanisms to examine and improve the product. While continuous builds, automated unit tests, and frequent

demonstrations of working code are all ways to focus attention on the product and allow the team to make adjustments, retrospectives focus attention on how the team does their work and interacts.

Retrospectives are also a natural fit in a team environment—where membership in the team is less than ten and the work is interdependent. Retrospectives help people improve practices, handle issues, and surface obstacles on a regular basis.

Iteration retrospectives focus on real problems that affect teams. During retrospectives, teams discover real solutions that they can implement without waiting for management’s permission. Since experiments and changes are chosen, not imposed from above, people are more invested in their success.

When we started leading retrospectives more than a decade ago, most retrospectives looked at whole projects that had run for a year or more. In the past ten years, there has been a shift. More and more teams are working in shorter iterations and releasing software more frequently. These teams no longer wait until the end of a long project to inspect and adapt. They look for ways to improve at the end of every iteration. Team coaches, team leads, and team members now lead their own retrospectives.

Even if your team isn’t using Agile methods, you can adapt the advice in this book to inspect and adapt your processes and teamwork before the end of a project: hold a retrospective every month or so or at project milestones.

You may need to convince your managers that this is a good use of your time and company dollars. A growing body of data—both financial and empirical—shows that consistent retrospectives result in real savings and improvements.

In this book, we’ll introduce a structure for retrospectives and walk through the process of planning, designing, and leading a retrospective. We’ll supply activities and guidance on how to use them, and we’ll share stories from real retrospectives.

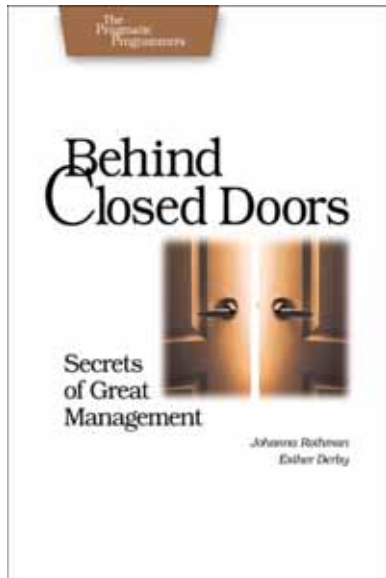
We’ve also included a chapter on the role of the retrospective leader. We believe that most people can lead retrospectives with confidence and competence—and help the team achieve results—with a good structure and the right tools.

And, we've included examples of how you can adjust the basic retrospective structure for a three-month release or a yearlong project—and anything in between. Even if the team disbands after the release or project, the organization can learn from a retrospective, and individuals will take the learning with them.

Also by Esther Derby

If you liked this book by Esther Derby and Diana Larsen, you might also enjoy the following title by Johanna Rothman and Esther Derby.

Behind Closed Doors



Secrets of Great Management.

You can learn to be a better manager—even a great manager—with this guide. You'll follow along as Sam, a manager just brought on board, learns the ropes and deals with his new team over the course of his first seven weeks on the job.

From organizing who works on what project when, to helping team members grow and prosper, you'll be there as Sam makes it happen for the entire team.

You'll find powerful tips covering:

- Delegating effectively
- Using feedback and goal-setting
- Developing influence
- Handling one-on-one meetings
- Coaching and mentoring
- Deciding what work to do—and what not to do
- . . . and more.

Full of tips and practical advice on the most important aspects of management, this is one of those books that can make a lasting and immediate impact on your career.

Behind Closed Doors Secrets of Great Management

Johanna Rothman and Esther Derby

(192 pages) ISBN: 0-9766940-2-6. \$24.95

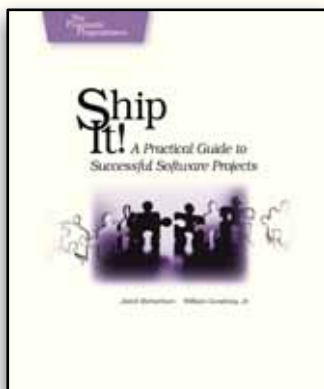
Visit our secure online store: <http://pragmaticprogrammer.com/catalog>

Competitive Edge

We've got you covered—whether it's titles to maintain your competitive edge, or the latest cutting-edge technologies.

For a full list of all of our current titles, as well as announcements of new titles, please visit www.pragmaticprogrammer.com.

Ship It!



Agility for teams. *Ship It!* lets you do just that: on time and on budget, without excuses. You'll see how to implement the common technical infrastructure that every project needs along with well-accepted, easy-to-adopt, best-of-breed practices that really work, as well as common problems and how to solve them.

Ship It!: A Practical Guide to Successful Software Projects

Jared Richardson and Will Gwaltney
(200 pages) ISBN: 0-9745140-4-7. \$29.95

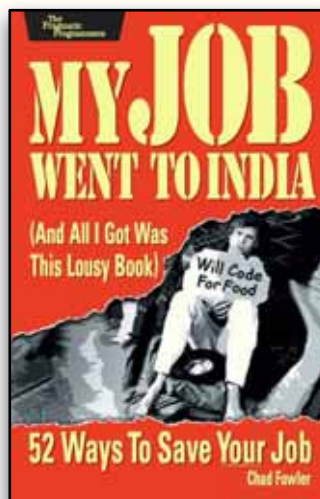
My Job Went to India

World class career advice. The job market is shifting. Your current job may be outsourced, perhaps to India or eastern Europe. But you can save your job and improve your career by following these practical and timely tips. See how to:

- treat your career as a business
- build your own brand as a software developer
- develop a structured plan for keeping your skills up to date
- market yourself to your company and rest of the industry
- keep your job!

My Job Went to India: 52 Ways to Save Your Job

Chad Fowler
(208 pages) ISBN: 0-9766940-1-8. \$19.95



Visit our secure online store: <http://pragmaticprogrammer.com/catalog>

Cutting Edge

Learn how to use the popular Ruby programming language from the Pragmatic Programmers: your definitive source for reference and tutorials on the Ruby language and exciting new application development tools based on Ruby.

The *Facets of Ruby* series includes the definitive guide to Ruby, widely known as the PickAxe book, and *Agile Web Development with Rails*, the first and best guide to the cutting-edge Ruby on Rails application framework.

Programming Ruby (The PickAxe)

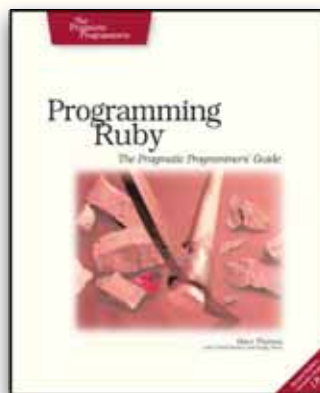
The definitive guide to Ruby programming.

- Up-to-date and expanded for Ruby version 1.8.
- Complete documentation of all the built-in classes, modules, methods, and standard libraries.
- Learn more about Ruby's web tools, unit testing, and programming philosophy.

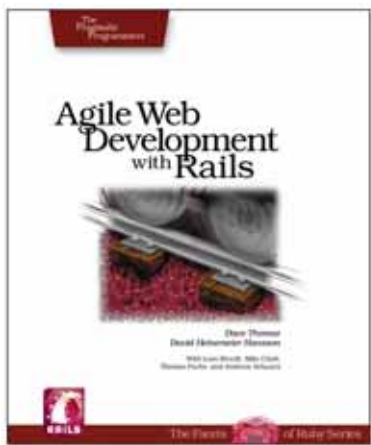
Programming Ruby: The Pragmatic Programmer's Guide, 2nd Edition

Dave Thomas with Chad Fowler
and Andy Hunt

(864 pages) ISBN: 0-9745140-5-5. \$44.95



Agile Web Development with Rails



A new approach to rapid web development.

Develop sophisticated web applications quickly and easily

- Learn the framework of choice for Web 2.0 developers
- Use incremental and iterative development to create the web apps that users want
- Get to go home on time.

Agile Web Development with Rails: A Pragmatic Guide

Dave Thomas and David Heinemeier Hansson
(570 pages) ISBN: 0-9766940-0-X. \$34.95

Visit our secure online store: <http://pragmaticprogrammer.com/catalog>

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style, and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help programmers stay on top of their game.

Visit Us Online

Agile Retrospectives's Home Page

pragmaticprogrammer.com/titles/dlret

Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates

pragmaticprogrammer.com/updates

Be notified when updates and new books become available.

Join the Community

pragmaticprogrammer.com/community

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy

pragmaticprogrammer.com/news

Check out the latest pragmatic developments in the news.

Buy the Book

If you liked this PDF, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragmaticprogrammer.com/titles/dlret.

Contact Us

Phone Orders:	1-800-699-PROG (+1 919 847 3884)
Online Orders:	www.pragmaticprogrammer.com/catalog
Customer Service:	support@pragmaticprogrammer.com
Non-English Versions:	translations@pragmaticprogrammer.com
Pragmatic Teaching:	academic@pragmaticprogrammer.com
Author Proposals:	proposals@pragmaticprogrammer.com