

Extracted from:

No Fluff Just Stuff Anthology, Volume 2

The 2007 Edition

This PDF file contains pages extracted from No Fluff Just Stuff Anthology, Volume 2, 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 © 2007The 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

1	Preface	11
2	Introduction	12
3	Learning to Love JavaScript	15
	by Glenn Vanderburg	
3.1	Introduction	16
3.2	Comfortable Confusion	16
3.3	Collateral Damage	18
3.4	Reconstruction	20
3.5	Revolution	22
3.6	Stop Worrying, and Love the DOM	23
3.7	Conclusion	28
3.8	Web Resources	28
4	The Joys of Continuations and Asynchronous Architectures	29
	by Rebecca J. Parsons	
4.1	Introduction	30
4.2	Neat and Nifty Tricks with Continuations	30
4.3	Why Continuations Are Useful...	33
4.4	What's the Catch?	35
4.5	What Does It Mean, and Why Should I Care?	36
4.6	Web Resources	36
5	The Case for Groovy	37
	by Scott Davis	
5.1	Introduction	38
5.2	News Flash: Java Is Complex	38
5.3	assertEquals(Power,Simplicity)	39
5.4	Beyond Java	40
5.5	Java Is Dead. Long Live Java!	41
5.6	Enter Groovy	42
5.7	Inviting Groovy to the Party	44

5.8	How Did He Know How to Do That?	46
5.9	Rapid Web Development with Grails	47
5.10	Wrapping Up	51
5.11	Web Resources	52
6	Give It a REST	54
	by Brian Sletten	
6.1	Introduction	55
6.2	REST Defined	57
6.3	WS-Tenacity	61
6.4	Complexity	62
6.5	REST and Security	65
6.6	REST and Metadata	66
6.7	REST Environments	68
6.8	Where REST Does Not Fit As Well	70
6.9	Supporting Both Styles	71
6.10	Conclusion	72
6.11	Web Resources	72
7	Guerilla Testing with Cobertura, TestNG, and EasyMock	74
	by Howard Lewis Ship	
7.1	Introduction	75
7.2	Testing and Measurement	75
7.3	TestNG: Testing, the Next Generation	79
7.4	EasyMock: Weird, Wonderful, and Invaluable	93
7.5	Conclusion	103
7.6	Web Resources	103
8	An Introduction to the Google Web Toolkit	105
	by David Geary	
8.1	Introduction	106
8.2	The Example Application	106
8.3	Using GWT Widgets	108
8.4	Internationalization	111
8.5	Remote Procedure Calls	113
8.6	Using the GWT's History Mechanism	117
8.7	Ajax!	119
8.8	JavaScript Integration	121
8.9	Conclusion	123
8.10	Web Resources	123

9	Web Testing with Selenium	124
	by Neal Ford	
9.1	Introduction	125
9.2	Origins	125
9.3	Using Selenium in TestRunner Mode	126
9.4	Selenium Concepts	131
9.5	Remote Control Selenium	136
9.6	Tools	140
9.7	Summary	142
9.8	Web Resources	142
10	IntelliJ Tips, Tricks, and Techniques	143
10.1	Introduction	143
10.2	Stop Typing the Left Side	143
10.3	Remote Debugging Setup	144
10.4	The IDE That Reads Your Mind	144
10.5	Quick File Finding	148
10.6	Quick Navigation	149
10.7	Column Select and Replace	150
10.8	Super-Smart Live Templates	151
10.9	Macro Recorder	152
10.10	Key Promoter Plug-In	153
10.11	Summary	154
10.12	Web Resources	154
11	Pragmatic Tips, Tricks, and Techniques: Eclipse	155
11.1	Introduction	155
11.2	Generating by Your Own Rules	155
11.3	Pasty-White Pasting	156
11.4	Tickling the Keys	156
11.5	Filtration	157
11.6	Blinded by Source Code	157
11.7	Managing Plug-Ins	157
11.8	Just Browsing	158
11.9	Window Shopping	158
11.10	Do-It-Yourself DBA	158
11.11	Emacs Key Bindings	159
11.12	Summary	160
11.13	Web Resources	160

12 Improving Code Quality Using Automation	161
by Paul M. Duvall	
12.1 Introduction	162
12.2 Code Quality	162
12.3 Maven	165
12.4 Code Quality Plug-ins for Maven	166
12.5 Summary	171
12.6 Web Resources	171
13 Capistrano: Application Deployment and More	173
by David Bock	
13.1 Introduction	174
13.2 Capistrano's Assumptions	174
13.3 Concepts	175
13.4 Kick-Starting Your Own Recipe File	177
13.5 Capistrano's Standard Tasks	177
13.6 Extending Capistrano	179
13.7 Rolling Your Own Tasks	181
13.8 Advanced Concepts	182
13.9 Tricks of the Trade	186
13.10 Automation Redux	189
13.11 Web Resources	190
14 Bootstrapping Agility	191
by Venkat Subramaniam	
14.1 What's Agility?	192
14.2 Status of Development	193
14.3 Agile Movement	195
14.4 Adaptive Planning	196
14.5 Agility and Architecture	197
14.6 Agility and the Architect	199
14.7 Agility and Modeling	200
14.8 Evolutionary Design	202
14.9 Evolving the Design	203
14.10 Testing and Integration	206
14.11 Agile Team	209
14.12 Essence of Agility	210
14.13 Web Resources	211

15 The Software Engineer's Guide to Usability	212
by Nathaniel T. Schutta	
15.1 Introduction	213
15.2 What Is Usability?	213
15.3 Usability Myths	215
15.4 Creating a Usable Interface	217
15.5 Usability Heuristics	223
15.6 Summary	224
15.7 Web Resources	224
16 Integrating Ruby with Your Legacy Code	225
by Jared Richardson	
16.1 A Well-Stocked Toolbox	226
16.2 Your Ruby Got Into My Java!	227
16.3 A Bridge Over Troubled Waters...	229
16.4 Access High-Performance C Code from Your Ruby	231
16.5 Wrapping Up	234
16.6 Web Resources	234
17 Executable Documentation with FIT and FitNesse	235
by David Hussman	
17.1 Stumbling Toward Innovation	236
17.2 Less Is More, Again!	237
17.3 Tools of the Trade	240
17.4 Where Do You Go from Here?	246
17.5 Web Resources	247
18 Introducing the iBATIS Persistence Framework	248
by Mark Richards	
18.1 Introduction	249
18.2 iBATIS vs. Traditional ORM Frameworks	249
18.3 Why Use iBATIS?	250
18.4 Setting Up Data Sources and Transactions	251
18.5 Mapping SQL Statements	252
18.6 Invoking iBATIS SQL Statements in Java	256
18.7 Using HashMaps vs. Java Transfer Objects	257
18.8 Caching in iBATIS	258
18.9 Namespaces	260
18.10 Conclusion	261
18.11 Web Resources	262

19 Data Access Using Spring, Hibernate, and JDBC	263
by Scott Leberknight	
19.1 Introduction	264
19.2 ORM vs. JDBC	264
19.3 When to Use ORM vs. JDBC	266
19.4 Why Use Spring?	267
19.5 Setting Up Hibernate in Spring	268
19.6 Implementing Hibernate-Based DAOs	272
19.7 Implementing Spring JDBC-Based DAOs	278
19.8 Mixing DAO Implementations	279
19.9 DAO Design Considerations	281
19.10 Testing	282
19.11 Summary	286
19.12 Web Resources	286
A The Authors Speak!	287
A.1 Glenn Vanderburg	287
A.2 Venkat Subramaniam	288
A.3 Brian Sletten	289
A.4 Howard Lewis Ship	290
A.5 Nathaniel T. Schutta	291
A.6 Jared Richardson	292
A.7 Mark Richards	292
A.8 Rebecca J. Parsons	293
A.9 Ted Neward	293
A.10 Scott Leberknight	293
A.11 David Hussman	294
A.12 David Geary	295
A.13 Neal Ford	297
A.14 Paul M. Duvall	299
A.15 Scott Davis	299
A.16 David Bock	300
B Bibliography	302
Index	307

Pragmatic Methodology

Welcome to the Pragmatic Community. We hope you've enjoyed this title.

Do you need to get software out the door? Then you want to see how to *Ship It!* with less fuss and more features.

And if you want to improve your approach to programming, take a look at the pragmatic, effective, *Practices of an Agile Developer*.

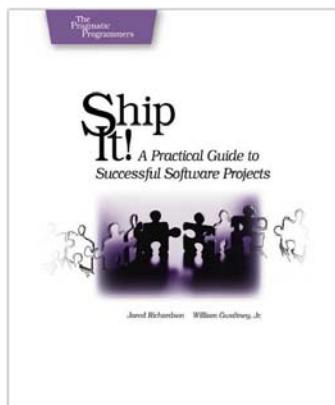
Ship It!

Page after page of solid advice, all tried and tested in the real world. This book offers a collection of tips that show you what tools a successful team has to use, and how to use them well. You'll get quick, easy-to-follow advice on modern techniques and when they should be applied. **You need this book if:**

- You're frustrated at lack of progress on your project.
- You want to make yourself and your team more valuable.
- You've looked at methodologies such as Extreme Programming (XP) and felt they were too, well, extreme.
- You've looked at the Rational Unified Process (RUP) or CMM/I methods and cringed at the learning curve and costs.
- **You need to get software out the door without excuses**

Ship It! A Practical Guide to Successful Software Projects

Jared Richardson and Will Gwaltney
(200 pages) ISBN: 0-9745140-4-7. \$29.95
<http://pragmaticprogrammer.com/titles/prj>



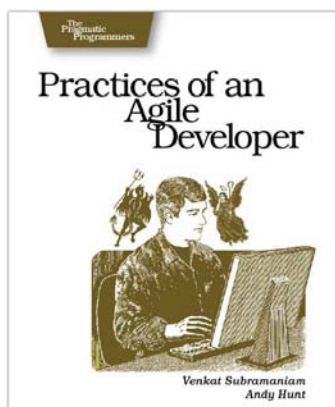
Practices of an Agile Developer

Agility is all about using feedback to respond to change. Learn how to apply the principles of agility throughout the software development process

- Establish and maintain an agile working environment
- Deliver what users really want
- Use personal agile techniques for better coding and debugging
- Use effective collaborative techniques for better teamwork
- Move to an agile approach

Practices of an Agile Developer: Working in the Real World

Venkat Subramaniam and Andy Hunt
(189 pages) ISBN: 0-9745140-8-X. \$29.95
<http://pragmaticprogrammer.com/titles/pad>



Pragmatic Tools

Interested in learning Ruby on Rails, but want a shortcut? If you already know Java, you can learn Rails easily using *Rails For Java Developers*.

In Rails or any Web 2.0 adventure, you need to know Ajax. What better way to start than with *Pragmatic Ajax*?

Rails for Java Developers

Enterprise Java developers already have most of the skills needed to create Rails applications. They just need a guide which shows how their Java knowledge maps to the Rails world. That's what this book does. It covers:

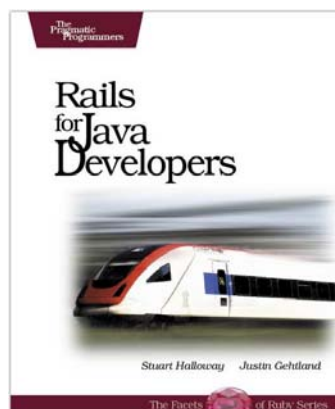
- The Ruby language
- Building MVC Applications
- Unit and Functional Testing
- Security
- Project Automation
- Configuration
- Web Services

This book is the fast track for Java programmers who are learning or evaluating Ruby on Rails.

Rails for Java Developers

Stuart Halloway and Justin Gehtland
(300 pages) ISBN: 0-9776166-9-X. \$34.95

http://pragmaticprogrammer.com/titles/fr_r4j



Pragmatic Ajax

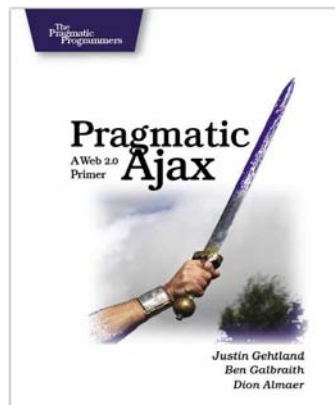
AJAX redefines the user experience for web applications, providing compelling user interfaces. Now you can dig deeper into AJAX itself as this book shows you how to make AJAX magic. Explore both the fundamental technologies and the emerging frameworks that make it easy.

From Google Maps to Ajaxified Java, .NET, and Ruby on Rails applications, this Pragmatic guide strips away the mystery and shows you the easy way to make Ajax work for you.

Pragmatic Ajax: A Web 2.0 Primer

Justin Gehtland, Ben Galbraith, Dion Almaer
(296 pages) ISBN: 0-9766940-8-5. \$29.95

<http://pragmaticprogrammer.com/titles/ajax>



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 you stay on top of your game.

Visit Us Online

No Fluff, Just Stuff 2007's Home Page

<http://pragmaticprogrammer.com/titles/nfjs07>

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

Register for Updates

<http://pragmaticprogrammer.com/updates>

Be notified when updates and new books become available.

Join the Community

<http://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

<http://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/nfjs07.

Contact Us

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