

The
Pragmatic
Programmers

Pragmatic Project Automation

*How To Build, Deploy,
and Monitor Java Applications*

The Pragmatic Starter Kit - Volume III



Mike Clark

What readers are saying about *Pragmatic Project Automation*

Where has this book been all my life?! Mike Clark's clear, concise, and fun style has me on the edge of my seat eager to find out what trick is next. His CruiseControl RSS publisher is already in action on my projects, and even more of these gems are working their way into my routine. Lava Lamps and Groovy—the hippest software book ever!

► **Erik Hatcher**, Co-author of *Java Development with Ant* and *Lucene in Action*

From the minute I learned about the *Pragmatic Starter Kit*, this book was the one I was most eager to read. Now that I've read it, it's the one I'm most eager to recommend.... Mike's book shows clearly that automation is part and parcel of good software engineering: well-designed systems lend themselves to automation, and automation helps us build good systems.

► **Glenn Vanderburg**

This book—fun and interesting to read—is a wonderful collection of tips and tricks that will help you take simple everyday tools and do amazing things with them.

► **James Duncan Davidson**, Creator of Ant

We've all heard that time is money. So it follows naturally that saving time is saving money. This book shows you how to do both.... I'm going to staple this book to my desk, so it doesn't 'disappear.'

► **David Rupp**, Sr. Software Engineer, Great-West Life & Annuity

If your software project is not automated, you are wasting time and money every day, and I can't think of a better, more thorough, or more practical book to get you started on that path than Mike Clark's *Pragmatic Project Automation*.

► **Alberto Savoia**, CTO, Agitar Software Inc.

If you've ever hoped to find a technical book that gave solid, usable examples that you can apply in real life rather than just throwing lofty-sounding buzzwords about, then this book is it.... This book will be mandatory reading on all of my projects from this time forth.

► **Simon P. Chappell**, Technical Lead, Lands' End, Inc.

This book is both inspiring and informative. I now have no excuses not to automate my projects.

► **David Vydra**, www.testdriven.com

Finally! Mike Clark has captured in this book what has taken me years to learn—how to make the computers do the dull repetitive work so that the developers can spend more time solving the REAL problems.... By implementing the simple guidelines presented here, every software project can, in essence, gain another team member.

► **Jonathan Julian**, Java Consultant

Doing the things this book describes saves each member of our team hours of grief and frustration every time we do a release. Overall, I think this is an excellent addition to the lineup—it's valuable stuff, and the writing and examples are very clear.

► **Greg Wilson**, Hewlett-Packard

Pragmatic Project Automation explains plainly how and why to automate the building, integration, and release of your projects. This book is a must-read for anyone who wants to have their software project succeed.

► **Johannes Brodwall**, Senior Software Architect, BBS
Norway

In the tradition of the Pragmatic series, *Pragmatic Project Automation* introduces a wealth of techniques (and free tools) for improving both software quality and software developers' quality of life.

► **Darin Herr**, Web Developer

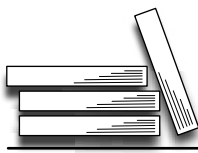
Pragmatic Project Automation

How to Build, Deploy, and Monitor
Java Applications

Mike Clark

The Pragmatic Bookshelf

Raleigh, North Carolina Dallas, Texas



Pragmatic Bookshelf

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 and the linking “g” device are trademarks of The Pragmatic Programmers, LLC. The configuration of the LAVA[®] brand motion lamp is a registered trademark of Haggerty Enterprises, Inc.

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 courses, workshops and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at:

<http://www.pragmaticprogrammer.com>

Copyright © 2004 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.

Printed in the United States of America.

ISBN 0-9745140-3-9

Printed on acid-free paper with 85% recycled, 30% post-consumer content.

First printing, June 2004

Version: 2004-6-30

Contents

About the Starter Kit	ix
Preface	xi
1 Introduction	1
1.1 Look Ma, No Hands!	1
1.2 Types of Automation	4
1.3 Questions About Automation	6
1.4 Road Map	9
2 One-Step Builds	10
2.1 Building Software Is Like Making Sausage	10
2.2 Choosing a Project Directory Structure	15
2.3 Making Your First Build	16
2.4 Building with Ant	19
2.5 Taste-Testing the Build	29
2.6 Cleaning Up	34
2.7 Scripting a Build	35
2.8 Getting an Early Start	39
3 Scheduled Builds	42
3.1 Scheduling Your First Build	43
3.2 Putting a Build on CruiseControl	46
3.3 Running CruiseControl	58
3.4 Publishing the Build Status	63
3.5 Scaling Up	68
4 Push-Button Releases	71
4.1 Releasing Early and Often	71
4.2 Preparing for Your First Release	72
4.3 Packaging the Release	78
4.4 Generating the Release	85
4.5 Tagging the Release	90

4.6	Handing Off the Release	92
4.7	Automating the Release Procedure	92
4.8	Generating Daily Distributions	94
5	Installation and Deployment	97
5.1	Delivering the Goods	97
5.2	Installing the Standard Distribution File	98
5.3	Troubleshooting by Phone	99
5.4	Troubleshooting with Diagnostic Tests	101
5.5	Enhancing Your Installed Image	107
5.6	Deploying Hosted Applications	115
5.7	Auto-Updating Installed Applications	120
6	Monitoring	125
6.1	Monitoring Scheduled Builds	125
6.2	Getting Feedback from Visual Devices	128
6.3	Monitoring Your Java Process	133
6.4	Checking Up on Your Web Application	134
6.5	Watching Log Files	136
6.6	Monitoring with log4j	138
6.7	Building Trip Wires with RSS	141
6.8	Monitoring Health with a Debug Command	143
6.9	Creating a Crash Report	144
6.10	3-2-1...	145
6.11	Automate!	147
A	Resources	148
A.1	On the Web	148
A.2	Bibliography	149
B	Pragmatic Project Automation: Summary	150

About the Starter Kit

Our first book, *The Pragmatic Programmer: From Journeyman to Master*, is a widely acclaimed overview of practical topics in modern software development. Since it was first published in 1999, many people have asked us about follow-on books, or sequels. We'll get around to that. But first, we thought we'd go back and offer a *prequel* of sorts.

Over the years, we've found that many of our pragmatic readers who are just starting out need a helping hand to get their development infrastructure in place, so they can begin forming good habits early. Many of our more advanced pragmatic readers understand these topics thoroughly, but they need help convincing and educating the rest of their organization or team. We think we've got something that can help.

The *Pragmatic Starter Kit* is a three-volume set that covers the essential basics for modern software development. These volumes include the practices, tools, and philosophies that you need to get a team up and running and superproductive. Armed with this knowledge, you and your team can adopt good habits easily and enjoy the safety and comfort of a well-established "safety net" for your project.

The first volume, *Pragmatic Version Control*, describes how to use version control as the cornerstone of a project. A project without version control is like a word processor without an Undo button: The more text you enter, the greater the risk, as mistakes will be more and more expensive to fix. *Pragmatic Version Control* shows you how to use version control systems effectively, with all the benefits and safety but without crippling bureaucracy or lengthy, tedious procedures.

Volume II, *Pragmatic Unit Testing*, discusses how to do effective unit testing. Unit testing is an essential technique as it

provides real-world, real-time feedback for developers as they write code. Many developers misunderstand unit testing and don't realize that it makes *our* jobs as developers easier. There are two versions of this volume: one based on JUnit (for Java), the other based on NUnit (for C#).

This book, *Pragmatic Project Automation*, is the third volume of the series. It covers the essential practices and technologies needed to automate your code's build, test, and release procedures. Few projects suffer from having too much time on their hands, so *Pragmatic Project Automation* will show you how to get the computer to do more of the mundane tasks by itself, freeing you to concentrate on the more interesting—and difficult—challenges.

These books were created in the same approachable style as our first book, and they address specific needs and problems that you face in the trenches every day. But these aren't dummy-level books that give you only part of the picture; they'll give you enough understanding that you'll be able to invent your own solutions to the novel problems you face that we *haven't* addressed specifically.

For up-to-date information on these and other books, as well as related pragmatic resources for developers and managers, visit us on the web at www.pragmaticprogrammer.com.

Thanks, and remember to make it fun!

Dave Thomas and Andy Hunt

June 2004

`pragprog@pragmaticprogrammer.com`

Pragmatic Project Automation shows you how to improve the consistency and repeatability of your project's procedures using automation to reduce risk and errors. Simply put, we're going to put this thing called a computer to work for you doing the mundane (but important) project stuff. That means you'll have more time and energy to do the really exciting and difficult stuff, like writing quality code. Customer Reviews. Once again, the Pragmatic team has written exactly what working programmers need. Doing the things this book describes saves each member of our team hours of grief and frustratio