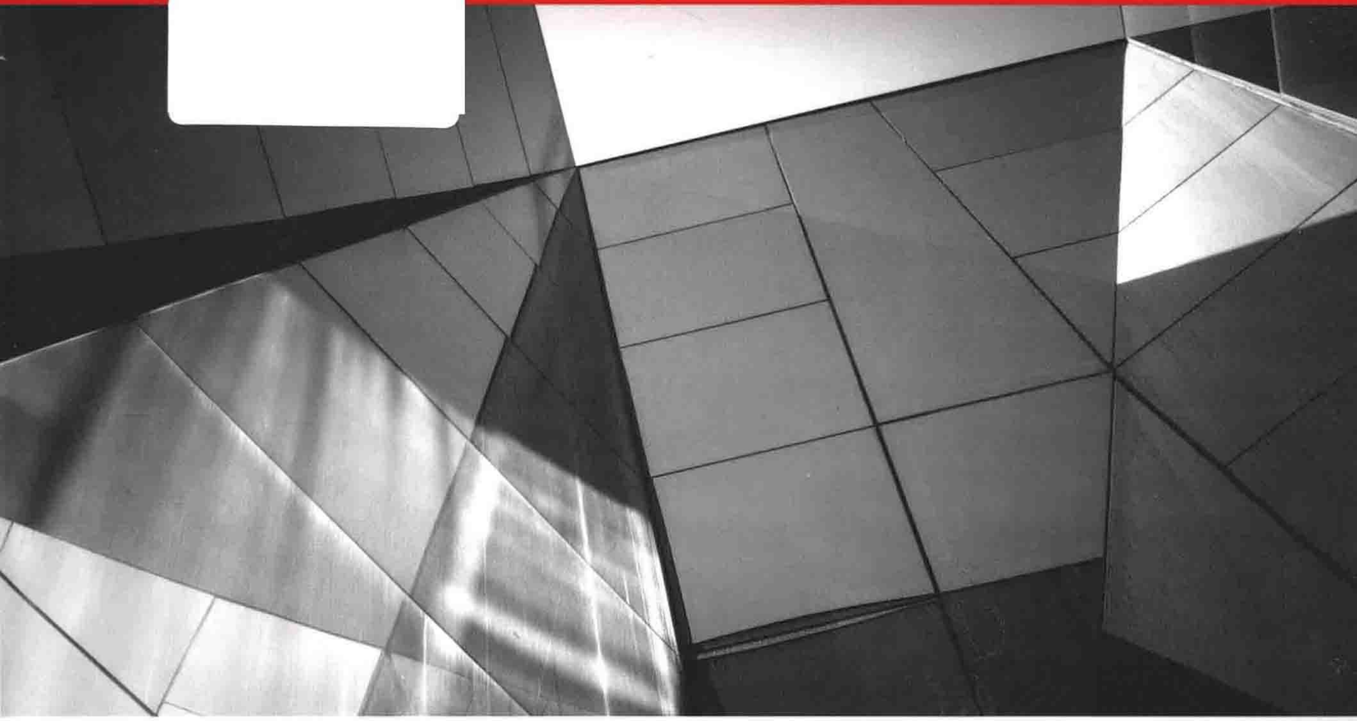
ORACLE



Hudson Continuous Integration in Practice

Maximize Quality and Minimize Software Development Time

Ed Burns
Winston Prakash

Foreword by Mike Milinkovich, Executive Director, Eclipse Foundation

Oracle Press

ORACLE®

Oracle Press™

Hudson Continuous Integration in Practice

Ed Burns and Winston Prakash

Mc
Graw
Hill
Education

New York Chicago San Francisco
Athens London Madrid Mexico City
Milan New Delhi Singapore Sydney Toronto

Cataloging-in-Publication Data is on file with the Library of Congress

McGraw-Hill Education books are available at special quantity discounts to use as premiums and sales promotions, or for use in corporate training programs. To contact a representative, please visit the Contact Us pages at www.mhprofessional.com.

Hudson Continuous Integration in Practice

1 2 3 4 5 6 7 8 9 0 DOC DOC 1 0 9 8 7 6 5 4 3

ISBN 978-0-07-180428-8
MHID 0-07-180428-5

Sponsoring Editor
Brandi Shailer

Editorial Supervisor
Janet Walden

Project Manager
Sheena Uprety,
Cenveo® Publisher Services

Acquisitions Coordinator
Amanda Russell

Technical Editors
Steve Christou and Bob Foster

Copy Editor
Margaret Berson

Proofreader
Bev Weiler

Indexer
Ted Laux

Production Supervisor
James Kussow

Composition
Cenveo Publisher Services

Illustration
Cenveo Publisher Services

Art Director, Cover
Jeff Weeks

Information has been obtained by Publisher from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Publisher, or others, Publisher does not guarantee to the accuracy, adequacy, or completeness of any information included in this work and is not responsible for any errors or omissions or the results obtained from the use of such information.

Oracle Corporation does not make any representations or warranties as to the accuracy, adequacy, or completeness of any information contained in this Work, and is not responsible for any errors or omissions.

To my wife, Amy, whose boundless patience and dedication inspire me in everything I do.

—Ed Burns

I dedicate this book to my wife, Dora, who never stops encouraging me to take on new endeavors.

—Winston Prakash

About the Authors

Ed Burns is currently a consulting engineer at Oracle America, Inc., where he leads a team of Web experts from across the industry in developing JavaServer Faces Technology through the Java Community Process and in open source. He is the author of three other books for McGraw-Hill: *Secrets of the Rock Star Programmers* (2008), *JavaServer Faces: The Complete Reference* (coauthored with Chris Schalk, 2006), and *JavaServer Faces 2.0: The Complete Reference* (coauthored with Chris Schalk and Neil Griffin, 2009).

Dr. Winston Prakash is currently an architect at Oracle Corporation. He has extensive experience in object-oriented design and development of large-scale applications. At present, his main focus is on developing data-driven, enterprise Web applications using advanced Java EE technologies. He leads the open source project, Hudson CI Server, at Eclipse Foundation.

About the Technical Editors

Steve Christou is a software developer with a focus on Continuous Integration. His experience with programming started with QBasic in high school and moved to Java and C++ in college. Steven currently has six years of progressive computer programming experience. He is now the lead designer of Cobertura, an open-source code coverage tool for Java, and continues to provide support for Hudson.

Bob Foster is a Consulting Developer at Oracle and an active Hudson committer. He designed and led implementation of the CPU/Memory Profiler in JDeveloper 11*g*. Prior to joining Oracle, he was an Eclipse contributor and author of the XMLBuddy XML Editor. He worked on several Integrated Development Environments (IDEs) for C++ and Java that are no longer in use, beginning 20 years ago, when he wrote the Classy C++ visual editor, which was purchased by Symantec and released as Visual Architect for Macintosh.

Foreword

Years ago, Tim O'Reilly coined the phrase "architecture of participation." His insight was that successful open source projects all shared one common trait: the ability for developers outside of the core team to easily extend the code to meet their needs. Examples of this successful pattern include the Linux kernel, the Apache Web server, Firefox extensions, and Eclipse plugins. Hudson was one of the first open source Continuous Integration tools to embrace this notion. It provided an extensible platform and, in turn, created an ecosystem of plugins that allowed it to support a myriad of repository, build, and analysis tools in many different workflows. Kohsuke Kawaguchi, Hudson's original author, worked hard to create a vibrant and engaged community around the Hudson project, and that has been a big part of its success.

In many ways, Hudson changed the face of software development. By making it significantly easier to implement Continuous Integration for a development team—or even an entire software organization—it moved what was considered a best practice to common practice. By allowing developers to more easily automate their builds, it helped make software production a repeatable process. By lowering the developer cost of builds, it helped make Continuous Integration mainstream. It is a rare open source project that can claim to have advanced the state of the art in software development.

The Eclipse Foundation itself is a great example of a large, distributed organization that relies on Hudson. At the time of writing, we have over 150 separate open source projects using Hudson as their Continuous Integration server. Our 2013 simultaneous release, Eclipse Kepler, consisted of 71 projects and 58 million lines of code. It was built by 420 Eclipse committers from over 50 companies around the world, as well as hundreds of individual contributors. And it was entirely built with Hudson. In fact, it is hard to imagine how we could operate without the automation and Continuous Integration provided by Hudson.

Winston Prakash is the leader of the Eclipse Hudson project, and is intimately aware of its internal architecture and implementation. I know that he has worked tenaciously over the past couple of years to improve the reliability, code quality, and scalability of Hudson. Ed Burns is the leader of JavaServer Faces, and has used Hudson extensively in managing the complexity of that project. Together the coauthors bring a wealth of experience and insight to the topic. I am sure that you will find their book a valuable resource as you embrace the "Hudson Lifestyle."

—Mike Milinkovich
Executive Director, Eclipse Foundation
June 2013

Acknowledgments

Hudson is a software tool that can help increase team productivity, but productivity lives or dies at the hands of the team wielding that tool. The same can be said for authoring a book. Neither Winston nor I could have been productive in writing this book without the team from McGraw-Hill Education. Sponsoring editor Brandi Shailer provided firm and experienced guidance through all phases of the authoring process. I especially appreciated her understanding as Winston and I juggled our day jobs with the task of writing this book. Acquisitions coordinator Amanda Russell weathered several episodes of chapter renumbering and reallocation and managed to keep track of all the moving parts of this book, entirely using e-mail attachments. Thanks to the illustrators at Cenveo Publisher Services for turning the rough art from Winston and me into professional images. Steve Christou and Bob Foster provided timely, in-depth, and above all, frank technical editing that served both as an advocate for the reader, and an oversight on cohesiveness. In a world where more and more information is available only online, and much of that information is coming from self-published individuals, I continue to believe that curated information from a traditional publishing team is the best way to deliver high-quality technical content in a portable and easily digestible way.

My personal support team also deserves a huge thank you. My wife, Amy, has supported me through four books now. With two small kids growing up, each one has been a different kind of sacrifice for her to pick up the slack I leave due to the spare-time nature of my technical book authoring arrangement. Thanks to my sons Owen and Logan as well, for understanding that it doesn't really take forever to finish a book; it just seems that way.

Thanks to Dad and Mom, Edward M. and Kathy Burns. Your solid support of my family has made it much easier to get this book done. Thanks also to my brother, Brendan Burns; as an avid reader, I appreciate your contribution to the book publishing ecosystem and, more importantly, I appreciate your personal support.

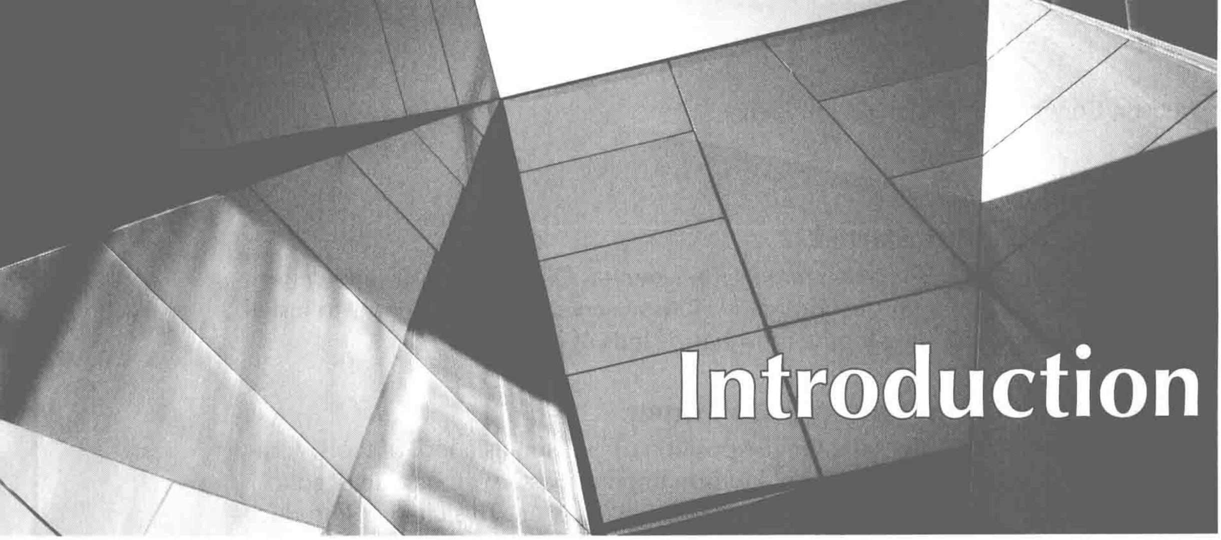
I'd like to give a special thanks to my long-time friends from Vienna, Austria: Erich Ruetz and Regina Preslmair, their two sons Christoph and Jakob, and Eric Sr. and Hedwig Ruetz. I look forward to the annual JSF conference in Vienna that I have been blessed to attend since 2008, and our traditional visits after that conference.

Finally, and above and under all, I give thanks to my Lord and Savior Jesus Christ, whose blessings make all things possible, including this book.

—Ed Burns
Florida, U.S.A.

First I want to thank Ed Burns for inviting me to write this book with him and giving me an opportunity to experience, first hand, writing a book. Writing the initial manuscripts may be easy, but taking them through the editing process and making them worthy chapters is an involved process. Many thanks go to Brandi Shailer and Amanda Russell for making the process easy for us. Special thanks go to Bob Foster and Steven Christou for taking time from their busy schedule to do a thorough technical review of the chapters. I also want to thank Jim Clark for the many helpful brainstorming sessions we had about Hudson. Finally, thanks to my wife and son for letting me work during weekends, sacrificing fun outdoor activities.

—Winston Prakash
California, U.S.A.



Introduction

In the ever-evolving practice of software development, the only constant is that complexity keeps increasing. In his work on JavaServer Faces at Oracle, coauthor Ed Burns has found that Hudson is the single most valuable software tool to enable keeping a lid on that complexity, though it must be carefully applied, lest even more complexity is created. This book places the Hudson Continuous Integration product in the context of the larger software development lifecycle as a means to achieve higher productivity and contain complexity. The reader will learn how to install Hudson in a way that suits their particular environment. Hands-on examples, in Java, will be used to explain the depth and breadth of using Hudson. This approach will lead the reader from novice, to apprentice, to mastery. The Hudson plugin ecosystem will be examined in detail, and the software architecture that enables that ecosystem will be fully explained, with code examples leading the reader to writing their own plugin. The book closes with a thorough treatment of how to effectively live the Hudson lifestyle, which can be summed up as "automate everything."

This book is divided into four parts and contains ten chapters and three appendixes.

Part I: Essential Knowledge

Reading Part I will give you everything you need to know to be completely effective using Hudson for the tasks for which it is most commonly used.

Chapter 1: Getting Started

This chapter introduces Hudson and its fundamental concept of "job" and defines the characteristics of a production-ready Hudson server. You'll learn how to install Hudson and get it running for the most basic kinds of jobs.

Chapter 2: Hudson Precondition Primer

This chapter acknowledges the unique position of Hudson as a tool of tools. As a consequence of this fact, mastery of Hudson implies mastery of, or at least solid familiarity with, all of the tools being managed by Hudson. This chapter introduces the reader to the main classes of tools that you will encounter as you learn to use Hudson in practice.

Chapter 3: Basic Concepts

This chapter rounds out the essential knowledge for using Hudson in most enterprise environments. After reading the first three chapters, you will have everything you need to know to be effective using Hudson for most common tasks. More importantly, you will have the foundation for learning more.

This chapter introduces Software Configuration Management (SCM) and shows how to add it to a job. The reader is walked through the process of creating jobs for a Java application and Java Servlet. The concept of Hudson plugins is introduced via the Plugin Center and the JobConfig plugin. The sample Hudson instance is configured for security. Finally, you are introduced to software quality assurance tools and the most common build notifiers.

Part II: Applying Hudson

Part II builds on the knowledge from Part I and covers usage appropriate for those who are committed to moving toward Continuous Delivery as their primary approach for developing and delivering software.

Chapter 4: Hudson as Continuous Integration Server

Continuous Integration is the main task performed by most Hudson instances. This chapter defines the term with examples and shows how to achieve practical continuous integration with Hudson. It also introduces the concept of Continuous Delivery and how to approach it with Hudson.

Chapter 5: Hudson and Automated Testing

Automated testing is essential to Continuous Integration. This chapter provides an overview of the kinds of automated testing one can perform with Hudson, with code examples of current testing technologies such as HtmlUnit and Arquillian.

Chapter 6: Hudson as Part of Your Tool Suite

Because Hudson is a tool of tools, it's important to examine how Hudson fits into the wider tool suite used by software developers. This chapter surveys IDE integration and issue tracker integration, and closes with some popular ways to stay informed on Hudson job status in browsers and on mobile devices.

Chapter 7: Distributed Building Using Hudson

Even though significant value can be achieved using a single Hudson instance, many production environments need more than what a single instance can provide. This chapter explains when you would need a distributed Hudson system, and how to set one up. The Matrix Build feature is explained in detail.

Chapter 8: Basic Plugin Development

You can't get very far in using Hudson without having to install plugins. Similarly, you can't get very far in mastering Hudson without learning how to write your own Hudson plugin. This chapter is tutorial content on only the essential aspects of writing Hudson plugins.

Part III: The Hudson Lifestyle

Part III stands alone from the preceding two parts but is essential to maximizing the benefit of using Hudson in a software development team.

Chapter 9: Advanced Plugin Development

This chapter expands on the knowledge in the previous chapter to cover less common, but still important, aspects of plugin development such as integrating with various dashboards, accessing SCM information from your plugin, custom notifiers, and the Jelly UI technology.

Chapter 10: Hudson Best Practices

System administration tasks are an important aspect of Hudson management. This chapter treats such topics as memory and disk requirements, JVM options, Web proxy concerns, server redundancy, and upgrading Hudson.

Part IV: Appendixes

The appendixes supplement content in Part III for important areas that do not warrant an independent chapter.

Appendix A: Widely Used Hudson Plugins

This appendix gives insight into the process used by Oracle for curating the plugins in the Hudson Plugin Manager, as well as providing an overview of the most important plugins available there.

Appendix B: Personal Hudson Instance

This appendix delivers on the promise of showing how to live the Hudson lifestyle by examining several approaches to using a personal Hudson instance (Hudson-as-valet) to achieve higher individual developer productivity.

Appendix C: Hudson for Windows Developers

In a nod to the fact that Hudson is used in non-Java development as well as Java development, this appendix introduces several plugins that make it possible to use Hudson effectively for Windows-based software development projects.

Intended Audience

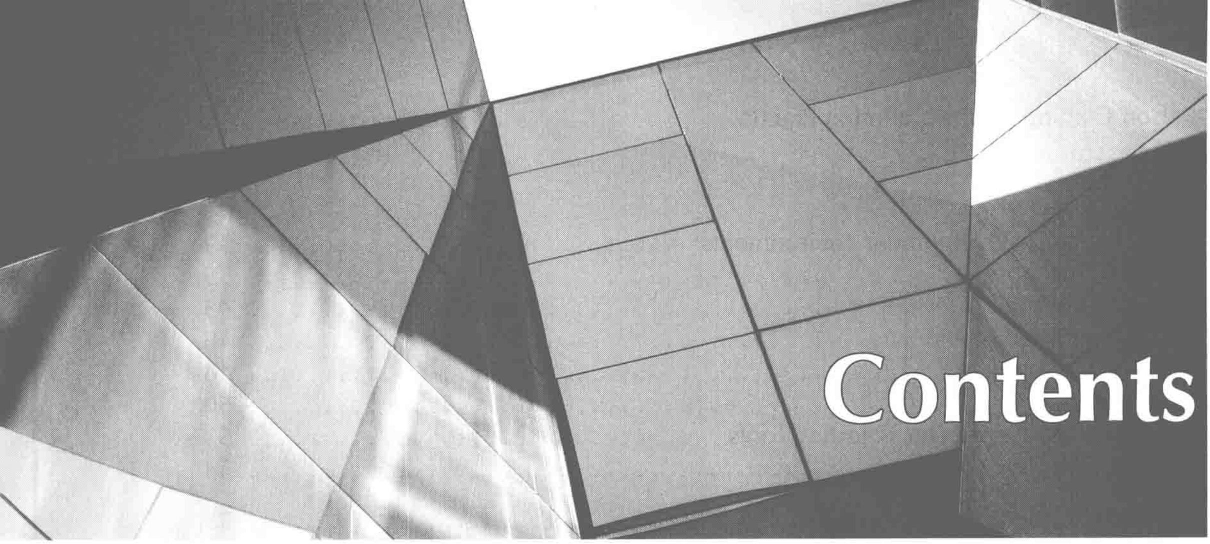
Code quality is something that everyone passionately admits is essential, but when it comes time to pay the cost to achieve it, the passion often dries up, leaving only the best efforts of the software developers writing the code. As such, software quality books tend to be seen as a longer-term investment than books on practical programming skills. In the face of this business reality, the saving grace of Hudson is that it admits this tendency of people to skimp on quality. Hudson endeavors to lower the cost of quality to the point where people will actually pay for it (mostly in terms of developer effort).

The major audience for this book is anyone working in software development. The Hudson product is clearly a system administration tool, but it is important to note that most developers *are not* sysadmins. In fact, developers often see sysadmin work as taking time away from their primary responsibility of programming. This book is aimed squarely at developers who want to spend less time on operations tasks and more time programming, and maintain higher code quality while doing so.

The secondary audience for this book is system administrators who work closely with programmers. These highly valued team members are often referred to as "buildmeisters." In practice, one of the developers often assumes this role, in addition to their normal programming responsibilities, but if the team is fortunate enough to have a dedicated buildmeister, this book is for them too.

Retrieving the Examples

Some of the source files used in this book can be downloaded from the Oracle Press website at www.OraclePressBooks.com. The files are contained in a ZIP file.



Contents

PART I
Essential Knowledge

PART II
Applying Hudson

PART III
The Hudson Lifestyle