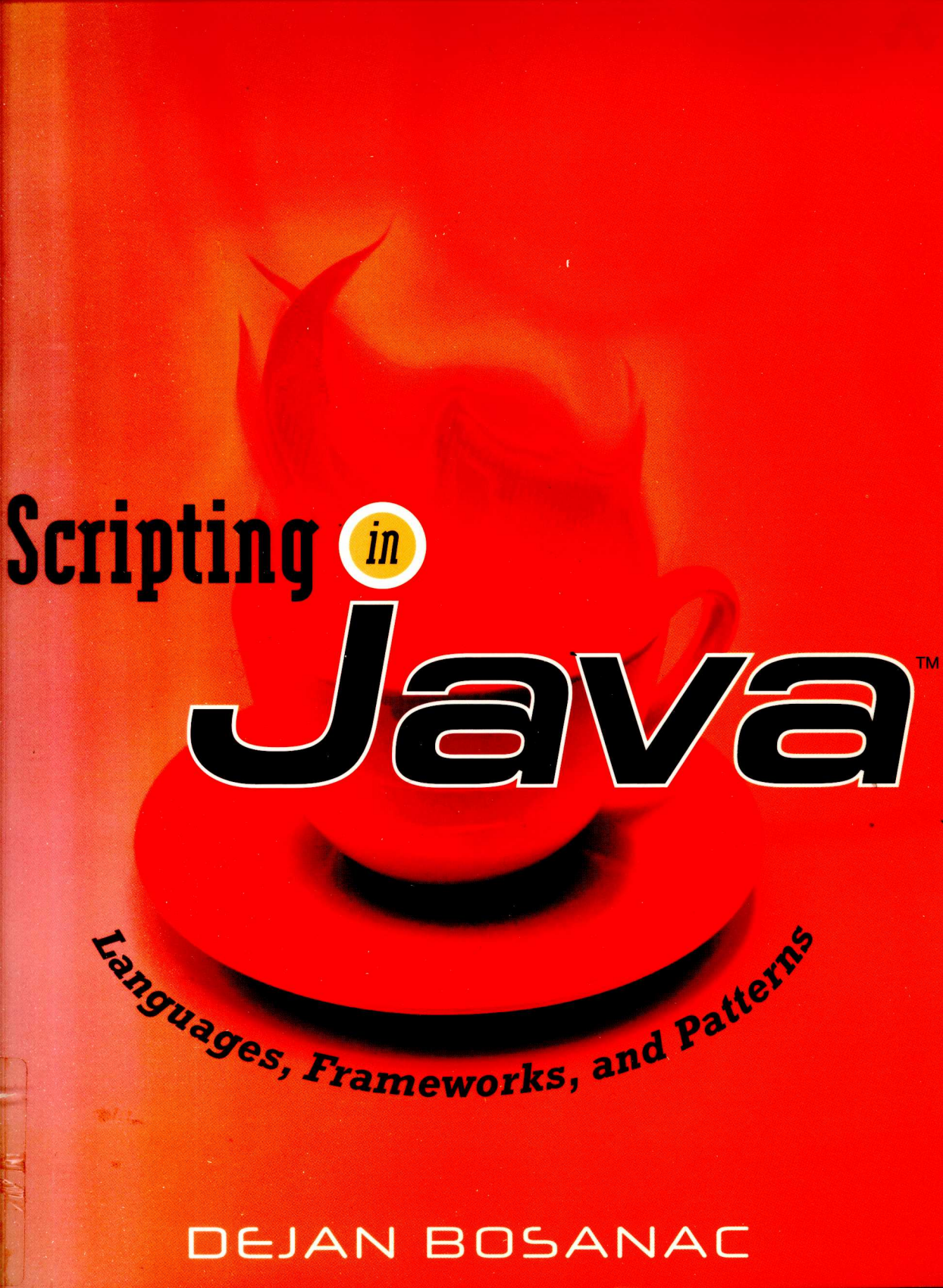
Scripting in Java™
Languages, Frameworks, and Patterns
DEJAN BOSANAC

Scripting in Java™

Languages, Frameworks, and Patterns

Dejan Bosanac

Addison-Wesley

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Cape Town • Sydney • Tokyo • Singapore • Mexico City

Library of Congress Cataloging-in-Publication Data

Bosanac, Dejan.
Scripting in java : languages, frameworks, and patterns / Dejan Bosanac.
p. cm.
ISBN 0-321-32193-6 (pbk. : alk. paper) 1. Java (Computer program language)
2. Programming languages (Electronic computers) I. Title.

QA76.73.J38B6715 2007
005.13'3—dc22

2007017654

ISBN-13: 978-0-321-32193-0
ISBN-10: 0-321-32193-6
Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, IN.
First printing August 2007

EDITOR-IN-CHIEF
Mark Taub

ACQUISITIONS EDITOR
Greg Doench

DEVELOPMENT EDITOR
Audrey Doyle

MANAGING EDITOR
Gina Kanouse

PROJECT EDITOR
Anne Goebel

COPY EDITOR
Geneil Breeze

INDEXER
Brad Herriman

PROOFREADER
Water Crest Publishing, Inc.

PUBLISHING COORDINATOR
Michelle Housley

COVER DESIGNER
Chuti Prasertsith

COMPOSITION
Bumpy Design

Dedication

To Ivana, for being so lovely

PREFACE

Java is an excellent object-oriented programming language. It has provided many benefits to software developers, including a good object-oriented approach, implicit memory management, and dynamic linking, among others. These language characteristics are one of the main reasons for Java's popularity and wide acceptance.

But Java is much more than a programming language; it's a whole development platform. This means that it comes with a runtime environment (JRE), which provides the virtual machine, and the standardized application programming interfaces (APIs) that help developers accomplish most of their desired tasks. The main advantages of this integrated runtime environment are its true platform independence and simplification of software development.

On the other hand, scripting languages have played an important role in the information technology infrastructure for many years. They have been used for all kinds of tasks, ranging from job automation to prototyping and implementation of complex software projects.

Therefore, we can conclude that the Java development platform can also benefit from scripting concepts and languages. Java developers can use scripting languages in areas proven to be most suitable for this technology. This synergy of the Java platform and scripting languages, as we will see, adds an extra quality to the overall software development process.

In this book, I describe the concepts behind scripting languages, summarize solutions available to Java developers, and explore use cases and design patterns for applying scripting languages in Java applications.

How This Book Is Organized

This book consists of five logical parts.

Part I

The first part of the book comprises two chapters that describe scripting languages in general:

- **Chapter 1, "Introduction to Scripting"**—Here I define the basic characteristics of scripting languages and compare them to system programming languages.

- Chapter 2, "Appropriate Applications for Scripting Languages"—In this chapter, I explain the role of traditional (native) scripting languages in the overall information technology infrastructure. I also discuss tasks for which scripting languages have been used in various systems over time.

Part II

After discussing the basic concepts and uses of scripting languages, we are ready to focus on real technologies and solutions for the Java platform. This part of the book contains the following chapters:

- Chapter 3, "Scripting Languages Inside the JVM"—I begin this chapter by covering the basic elements of the Java platform and explaining where scripting languages fit into it. After that, I describe the main features of three popular scripting languages available for the Java Virtual Machine (JVM)—BeanShell, JavaScript, and Python—and how they can be used to interact with Java applications. At the end of this chapter, I describe other solutions available for Java developers.
- Chapter 4, "Groovy"—Here I discuss the Groovy scripting language in detail. I cover its Java-like syntax and all the scripting concepts built into this language, and I discuss Groovy's integration with Java, as well as some security-related issues.
- Chapter 5, "Advanced Groovy Programming"—In this chapter, I cover some of the Groovy extensions that can aid in day-to-day programming tasks. I also explain how Java programmers can access databases, create and process XML documents, and easily create simple Web applications and swing user interfaces, using the scripting-specific features in Groovy covered in Chapter 4.
- Chapter 6, "Bean Scripting Framework"—In this chapter, I describe the general Java scripting framework. In addition to explaining how to implement general support in your project for any compliant scripting language, I also discuss some basic abstractions implemented in the Bean Scripting Framework (BSF) and show some examples of successful uses.

Part III

This part of the book focuses primarily on the use of scripting languages in real Java projects:

- **Chapter 7, "Practical Scripting in Java"**—Here I cover topics related to the use of scripting for everyday programming tasks, such as unit testing, interactive debugging, and project building, among others.
- **Chapter 8, "Scripting Patterns"**—In this chapter, I discuss Java application design patterns that involve scripting languages. I show how you can use scripts to implement some parts of traditional design patterns and introduce some new design patterns specific only to the scripting environment. I also discuss the pros and cons of these design patterns, as well as their purpose.

Part IV

In the final part of this book, I cover the "Scripting for the Java Platform" specification, which was created according to the Java Specification Request (JSR) 223. Specifically, I cover two APIs defined by the specification:

- **Chapter 9, "Scripting API"**—Here I cover the Scripting API, the standardized general scripting framework for the Java platform. The purpose of this framework is the same as that of the Bean Scripting Framework, but the Scripting API brings many new features that the modern scripting framework needs. The Scripting API is a standard part of the Java platform with the release of Mustang (Java SE 6).
- **Chapter 10, "Web Scripting Framework"**—In this chapter, I discuss the Web Scripting Framework, a framework built on top of the Scripting API and created to enable scripting languages to generate Web content inside a servlet container. I explain how native scripting languages, such as PHP, can be synergized with the Java platform to bring more flexibility in Web application development.

Part V

At the end of the book, you can find a section comprising three appendixes. The main purpose of these appendixes is to provide the technical details about installation and use of certain technologies described in the book:

- **Appendix A, "Groovy Installation"**—In this appendix, I describe how to install, build, and configure the Groovy scripting language. A working installation of the Groovy interpreter is needed to run the code samples from the text.

- **Appendix B, "Groovy IDE Support"**—In this appendix, I provide instructions on how to install general Groovy support for Integrated Development Environments (IDEs).
- **Appendix C, "Installing JSR 223"**—Here I describe how to install the reference implementation (RI) of the JSR 223, which is needed to run examples from Chapter 10.

I hope you'll enjoy reading the book.

About the Web Site

This book is extended with a Web site at www.scriptinginjava.net where you can find the following:

- Source codes of all examples shown in the book available for download
- Book news, updates, and additions
- News and information related to this field of software development

Acknowledgments

I would like to thank my family, friends, and colleagues for endless patience and support during the writing of this book. I'm also grateful to the people from Addison-Wesley for believing in this material and making an excellent atmosphere to work in, especially my editors, Greg Doench and Ann Sellers. I would also like to thank all technical reviewers, especially George Jempty, Kevin Davis, and Rich Rosen. They have provided valuable feedback and helped me keep my focus when I was stranded. Without Audrey Doyle, this material would be much harder to read. Thank you for helping me shape the manuscript.

Finally, this book wouldn't be possible without all developers contributing their time to projects covered by this material.

About the Author

Dejan Bosanac is a professional software developer and technology consultant. He is focused on the integration and interoperability of different technologies, especially ones related to Java and the Web. Dejan spent a number of years in development of complex software projects, ranging from highly trafficked Web sites to enterprise applications, and was a member of the JSR 223 Expert Group.

CONTENTS