

Extracted from:

Seven Concurrency Models in Seven Weeks
When Threads Unravel

This PDF file contains pages extracted from *Seven Concurrency Models in Seven Weeks*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.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 printed versions; the content is otherwise identical.

Copyright © 2014 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.

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina

Seven Concurrency Models in Seven Weeks

When Threads Unravel



Paul Butcher

Series editor: *Bruce Tate*

Development editor: *Jacquelyn Carter*

Seven Concurrency Models in Seven Weeks

When Threads Unravel

Paul Butcher

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina



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, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

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://pragprog.com>.

The team that produced this book includes:

Bruce A. Tate (series editor)
Jacquelyn Carter (editor)
Potomac Indexing, LLC (indexer)
Molly McBeath (copyeditor)
David J Kelly (typesetter)
Janet Furlow (producer)
Ellie Callahan (support)

For international rights, please contact rights@pragprog.com.

Copyright © 2014 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-13: 978-1-937785-65-9
Encoded using the finest acid-free high-entropy binary digits.
Book version: P1.0—July 2014

Preface

In 1989 I started a PhD in languages for parallel and distributed computing—I was convinced that concurrent programming was about to turn mainstream. A belated two decades later, I’ve finally been proven correct—the world is buzzing with talk of multiple cores and how to take advantage of them.

But there’s more to concurrency than achieving better performance by exploiting multiple cores. Used correctly, concurrency is the key that unlocks responsiveness, fault tolerance, efficiency, and simplicity.

About This Book

This book follows the structure of The Pragmatic Bookshelf’s existing *Seven in Seven* books, [Seven Languages in Seven Weeks \[Tat10\]](#), [Seven Databases in Seven Weeks \[RW12\]](#), and [Seven Web Frameworks in Seven Weeks \[MD14\]](#). The seven approaches here have been chosen to give a broad overview of the concurrency landscape. We’ll cover some approaches that are already mainstream, some that are rapidly becoming mainstream, and others that are unlikely to ever be mainstream but are fantastically powerful in their particular niches. It’s my hope that, after reading this book, you’ll know exactly which tool(s) to reach for when faced with a concurrency problem.

Each chapter is designed to be read over a long weekend, split up into three days. Each day ends with exercises that expand on that day’s subject matter, and each chapter concludes with a wrap-up that summarizes the strengths and weaknesses of the approach under consideration.

Although a little philosophical hand-waving occurs along the way, the focus of the book is on practical working examples. I encourage you to work through these examples as you’re reading—nothing is more convincing than real, working code.

What This Book Is Not

This book is not a reference manual. I'm going to be using languages that might be new to you, such as Elixir and Clojure. Because this is a book about concurrency, not languages, there are going to be some aspects of these languages that I'll use without explaining in detail. Hopefully everything will be clear from context, but I'm relying on you to persevere if you need to explore some language feature further to understand fully. You might want to read along with a web browser handy so you can consult the language's documentation if you need to.

Nor is this an installation manual. To run the example code, you're going to need to install and run various tools—the README files included in the example code contain hints, but broadly speaking you're on your own here. I've used mainstream toolchains for all the examples, so there's plenty of help available on the Internet if you find yourself stuck.

Finally, this book is not comprehensive—there isn't space to cover every topic in detail. I mention some aspects only in passing or don't discuss them at all. On occasion, I've deliberately used slightly nonidiomatic code because doing so makes it easier for someone new to the language to follow along. If you decide to explore one or more of the technologies used here in more depth, check out one of the more definitive books referenced in the text.

Example Code

All the code discussed in the book can be downloaded from the book's website.¹ Each example includes not only source but also a build system. For each language, I've chosen the most popular build system for that language (Maven for Java, Leiningen for Clojure, Mix for Elixir, sbt for Scala, and GNU Make for C).

In most cases, these build systems will not only build the example but also automatically download any additional dependencies. In the case of sbt and Leiningen, they will even download the appropriate version of the Scala or Clojure compiler, so all you need to do is successfully install the relevant build tool, instructions for which are readily available on the Internet.

The primary exception to this is the C code used in [Chapter 7, Data Parallelism, on page ?](#), for which you will need to install the relevant OpenCL toolkit for your particular operating system and graphics card (unless you're on a Mac, that is, for which Xcode comes with everything built in).

1. <http://pragprog.com/book/pb7con/>

A Note to IDE Users

The build systems have all been tested from the command line. If you're a hardcore IDE user, you should be able to import the build system into your IDE—most IDEs are Maven-aware already, and plugins for sbt and Leiningen can create projects for most mainstream IDEs. But this isn't something I've tested, so you might find it easier to stick to using the command line.

A Note to Windows Users

All the examples have been tested on both OS X and Linux. They should all run just fine on Windows, but they haven't been tested there.

The exception is the C code used in [Chapter 7, *Data Parallelism*, on page ?](#), which uses GNU Make and GCC. It should be relatively easy to move the code over to Visual C++, but again this isn't something I've tested.

Online Resources

The apps and examples shown in this book can be found at the Pragmatic Programmers website for this book.² You'll also find the community forum and the errata-submission form, where you can report problems with the text or make suggestions for future versions.

Paul Butcher

Ten Tenths Consulting

paul@tententhsconsulting.com

Cambridge, UK, June 2014

2. <http://pragprog.com/book/pb7con>