

Extracted from:

Ruby on Rails Background Jobs with Sidekiq

Run Code Later without Complicating Your App

This PDF file contains pages extracted from *Ruby on Rails Background Jobs with Sidekiq*, 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 © 2023 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

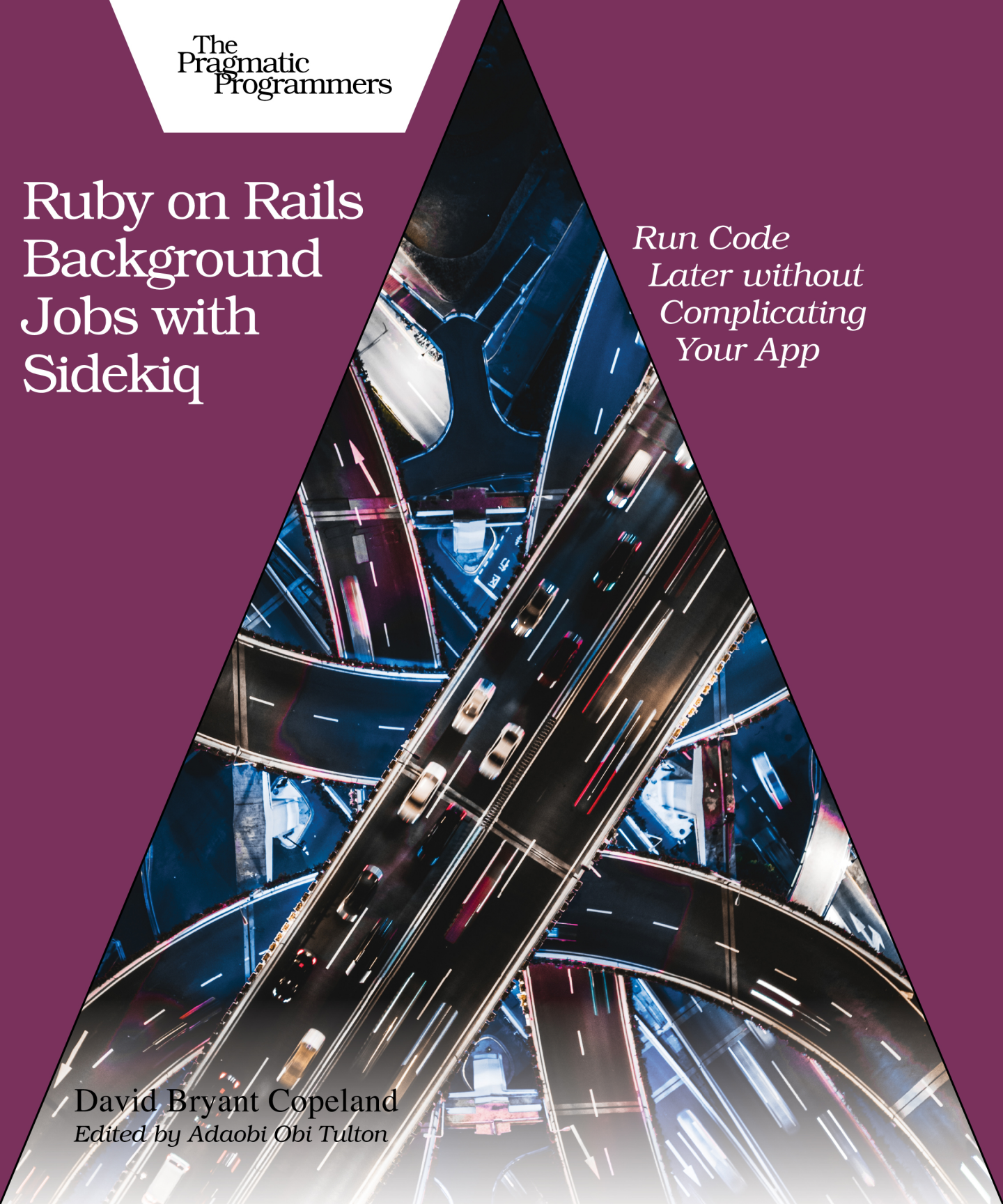
Dallas, Texas

The
Pragmatic
Programmers

Ruby on Rails Background Jobs with Sidekiq

*Run Code
Later without
Complicating
Your App*

David Bryant Copeland
Edited by Adaobi Obi Tulton



Ruby on Rails Background Jobs with Sidekiq

Run Code Later without Complicating Your App

David Bryant Copeland

The Pragmatic Bookshelf

Dallas, Texas



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.

For our complete catalog of hands-on, practical, and Pragmatic content for software developers, please visit <https://pragprog.com>.

The team that produced this book includes:

Publisher: Dave Thomas

COO: Janet Furlow

Managing Editor: Tammy Coron

Development Editor: Adaobi Obi Tulton

For sales, volume licensing, and support, please contact support@pragprog.com.

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

Copyright © 2023 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.

ISBN-13: 979-8-88865-036-3

Encoded using the finest acid-free high-entropy binary digits.

Book version: P1.0—October 2023

Preface

My first job with Ruby on Rails was at LivingSocial in 2011, about two months before the biggest shopping day in the United States called "Black Friday." I was the brand new owner of our payments system, which was built on background jobs using Resque. I had never heard of Resque and never worked on a system that had background jobs. I had never worked on a system that charged people money.

The only other person that knew anything about this code got so sick at one point that the CTO had to call his wife to make sure he was still alive (he was). During that time, thousands of background jobs were failing. They sat in a queue with error messages no one understood. The CTO, VP of Engineering, and I sat in front of a computer together and eventually figured out how to fix these jobs. They wouldn't be the last, but I became self-sufficient over the next year. It was a painful crash course in resiliency, fault tolerance, and background jobs.

Many years later, I was building a system that was almost entirely background jobs. I chose Sidekiq since it was clearly the de facto standard for background jobs in Rails. Although I had a lot of experience with background jobs at this point, I had none with Sidekiq. It was immediately obvious how much thought was put into the design. Everything I needed was there, and any strategy I wanted to employ to manage production issues was easily supported. It was a joy to work with.

Thinking back, if someone had explained and demonstrated what happens in the real world, and shown me some examples of how to deal with it, I could've avoided that harrowing trial by fire. If someone had then said, "Now that you see what happens in a production system, use Sidekiq," I could've saved myself quite a few deep dives into other systems' codebases.

This book, and the code that goes with it, are what I would've wanted. Not just theory, but a real demonstration of what can actually happen, and how to manage it.