

*The enemy of the truth is very often not the lie—deliberate, contrived, and dishonest—but the myth—persistent, persuasive, and unrealistic.*

► John F. Kennedy

# Preface

---

## The Explosion

When I started writing *From Java to Ruby*, I had a good idea that Java was not the perfect solution for every problem, though some in the industry may wish it were so. I knew Java visionaries were nervously abandoning Java as they began to recognize real productivity problems and unrelenting complexity. I saw the Java community suffer through an unreal proliferation of frameworks, providing new choices but also introducing an integration quagmire and a near-paralyzing uncertainty. I heard from Java customers flailing against the tide of increasing complexity, desperate to keep their heads above water. In truth, some of those customers needed Java, and many still do. Still, others used Java because it was the popular platform at the time, even though more suitable alternatives existed. With slowly building momentum behind Ruby on Rails, I saw Ruby as a good candidate for growth. If Ruby kept growing, *From Java to Ruby* might have a chance.

But I had no idea how violent the explosion could be. Since I started writing this book, I've seen Ruby on Rails downloads grow by nearly an order of magnitude. Where I used to get dozens of emails a day, I now sometimes get hundreds of emails in a few hours from the Rails support forums. You can get books, hosting, training, consulting, and informed opinions from many reputable sources. Make no mistake: Ruby is exploding, and developers are driving the revolution. Developers can see the simplicity and power of Ruby, and developers first experience the amazing productivity improvements when using Rails.

The problem is this: developers don't usually pick technologies or sign checks. If Ruby is to continue the rapid growth, we developers need to understand how to make an effective case for our technology—but not by using technical buzzwords. We need to communicate in the language our managers understand.

*Those deciding between Java and Ruby must understand how Ruby can save them money and help them better satisfy the needs of their customers.*

I now believe that the ideas expressed in this book fill a vacuum. If this programmer-led revolution is to advance into the enterprise where it can do the most good, we must learn to express how the technical advantages of Ruby help solve business problems in ways that Java can't. After interviewing customers, visionaries, and Ruby programmers for this book I am more convinced than ever that Ruby represents a fundamental advancement over Java for many of the most important problems we need to solve. In *From Java to Ruby*, you'll learn:

- Why the Ruby risk profile is decreasing, even as Java's rapidly accelerates.
- Where Ruby can help, where it can't, and where it simply has more growing to do.
- Pilot strategies others have successfully used across many industries and circumstances.
- What industry visionaries say about Ruby.

If you think you might want to consider Ruby, other books may concentrate on helping you express your Java programs in Ruby. This book is about moving minds. If you are a manager, *From Java to Ruby* will help you articulate why Ruby is so important to your project, developers, and your customers. If you are a developer, you can buy this book for your manager or use the ideas to convince him yourself. These ideas work. I've used them successfully to jump-start my Ruby practice, and two reviewers of the book have already used them to help their management teams make the right decisions. They can work for you, too.

## Acknowledgments

Writing a book is a challenging endeavor that tests each author with every word. Writing a book worthy of my name, and that of my colleagues at the Pragmatic Bookshelf, takes passion and plenty of help. *From Java to Ruby* would not have been possible without many people who stirred my passion, provided technical support, answered my questions, and provided frank criticism.

I'm profoundly grateful to all who helped. If I fail to mention you, please accept my heartfelt apologies. Please let me know so I can list you here.

In particular, I would like to thank the people who use and drive this emerging language. This book would not be the same without the practical experience each one of you provided. I thank my good friend Stuart Halloway for an excellent discussion about the merits of Ruby. Thanks also to Neal Ford for shaping my thinking, and sharing your network.

Thanks to Martin Fowler for the phone conversations that helped shape my thinking and the subsequent interview that advances the ideas in this book. I'm a huge admirer, and I was more than a little in awe as I interviewed you. Thanks to Joshua Haberman at Amazon.com and Heri ter Steeg for telling your stories about real production applications in Ruby. Your ideas are compelling. I hope they will motivate others to succeed with Ruby as you have.

As an open source community, many developers contribute excellent frameworks and ideas to this community without any compensation. Several of the interviews in this book are by such people. Thanks to David Heinemeier Hansson for your astounding energy leading to the Rails framework. Thanks also for your note setting me straight about how things are accomplished in the Rails world—the note that ultimately led to the interview in this book. I'm thrilled with your creation and learning more about the little things you got right every day. Thanks to Jamis Buck, again, for your contributions to migrations, Capistrano, and base Rails. And thanks for being willing to tell my readers about your experience. You've always been willing to help.

The JRuby project is so exciting to me that I dedicated space for two interviews on the topic. Thanks to Thomas E. Enebo and Charles O. Nutter for your Herculean efforts in actually making Rails run on the JVM. I can't believe I'm actually typing that. Also, thanks for spending so much time with me so I could tell my readers what you've done and for helping me get the details right. I've got high hopes for JRuby from many different perspectives.

Thanks also to all of those who reviewed *From Java to Ruby*: Jeffrey Blessing, Miguel Serrano, Nate Schutta, Robert Brown, Steve Yegge, Venkat Subramaniam, and Wesley Reisz. Your comments were often frank. If your ego can survive the experience, these are the types of comments that help a book. I was blown away by the quality of your comments. I built a document with each comment you made, and considered every single one.

This is the first review process I've been through with such good feedback from each and every reviewer. Thanks also to Kim Wimpsett for a thorough copy edit.

I have a special desire to thank Dave and Andy. You both stepped in to do jobs that I'd never expect a typical publisher to do. Then again, nothing about the Pragmatic Bookshelf is the least bit typical. Dave, we've worked the same conference for three years now, and I'm finally getting around to doing a book with you. I should have done it before. The experience has been particularly rewarding to me. You opened doors for me that might still otherwise be closed. It must have been maddening working with some of my early stuff. Thanks for the hours formatting my text and graphics before I got comfortable with my new tool set. Andy, thanks for stepping out of your role and into mine to fill in details and make sure each pixel was in the right place. You've got a fantastic sense for what a book needs. I've not worked with this end of the Pragmatic dynamic duo, and now feel shorted. Individually, you're top notch. As a team, you're amazing to work with. Thanks to both of you for giving me this opportunity and helping me to make the most of it. And thanks for bringing a much needed jolt of sanity to the publishing process. It's nice to see the good guys do well.

More than anyone else, I'd like to thank my family. Julia, you will never know how beautiful you are to me. When you laugh, the whole room laughs with you. Kayla, your smile works magic on my soul. I've been tremendously proud of you both for all you've done at school as dyslexics. Keep working hard. You both know that daddy is dyslexic too, but if I can write a book or two, you can overcome your problems to make words do anything you want. Maggie, through it all, you have been the love of my life and my inspiration. You support me even when you are afraid. You give me joy and laughter even when pressing publishing deadlines make it seem like there's not much to smile or laugh about. You do the little things like proofreading some early disasters, helping me work through copy edits, and anything else that needs doing. More than anything else, you love me. That love feeds my energy and passion for writing. I'd be a worthless author without you. I love you, always and ever.

*Bruce Tate*

*June, 2006*

bruce@rapidred.com