Programming Ruby

The Pragmatic Programmer's Guide

Previous < Contents ^ Next >

Preface

This book is a tutorial and reference for the Ruby programming language. Use Ruby, and you'll write better code, be more productive, and enjoy programming more.

These are bold claims, but we think that after reading this book you'll agree with them. And we have the experience to back up this belief.

As Pragmatic Programmers we've tried many, many languages in our search for tools to make our lives easier, for tools to help us do our jobs better. Until now, though, we'd always been frustrated by the languages we were using.

Our job is to solve problems, not spoonfeed compilers, so we like dynamic languages that adapt to us, without arbitrary, rigid rules. We need clarity so we can communicate using our code. We value conciseness and the ability to express a requirement in code accurately and efficiently. The less code we write, the less that can go wrong. (And our wrists and fingers are thankful, too.)

We want to be as productive as possible, so we want our code to run the first time; time spent in the debugger is time stolen from the development clock. It also helps if we can try out code as we edit it; if you have to wait for a 2-hour make cycle, you may as well be using punch cards and submitting your work for batch compilation.

We want a language that works at a high level of abstraction. The higher level the language, the less time we spend translating our requirements into code.

When we discovered Ruby, we realized that we'd found what we'd been looking for. More than any other language with which we have worked, Ruby *stays out of your way*. You can concentrate on solving the problem at hand, instead of struggling with compiler and language issues. That's how it can help you become a better