

Up Front • Functional Programming and Gonzo Engineering

• Michael Swaine • If you're the kind of person who think refactoring code to make it conceptually cleaner is exciting, then consider this issue your guidebook to adventure.

Choice Bits • A Smattering of Nattering • Michael Swaine • We follow Twitter so you don't have to.

Functional Programming Basics • What's It All About?

Robert Martin • "Uncle Bob" Martin strips the paradigm down to its essentials, and explains why you can and must understand functional programming now. • functional

Web Programming in Haskell, Part II • Rethinking the

Structure of Web Applications • Paul Callaghan • Paul continues his deep dive into the Haskell language and functional programming with a look at Web frameworks for Haskell. • functional, frameworks, haskell, web

Using Memoization in Groovy • Dynamic Programming in a Dynamically Typed Language

• Venkat Subramaniam • Venkat explores the immense time-space tradeoffs of memoization, and explains how Groovy makes memoization easy. • groovy, functional

Mac Kung Fu: Breaking the Magician's Code • Uncovering "Default Write" Commands

• Keir Thomas • Keir breaks the magician's code and reveals useful secrets that Apple has hidden in its code. • cocoa, osx

Launching a Gonzo Engineering Project • I: What Is Gonzo Engineering?

• Steven K. Roberts • The legendary gonzo engineer shares his secrets for pursuing crazy dreams and succeeding in a new series starting in this issue. • hardware

The Quiz • Quotable Coders • Michael Swaine • A monthly diversion at least peripherally related to programming.

Shady Illuminations • Instagram's Image Problem • John Shade • By reading this column you agree to abide by incomprehensible terms of use that John reserves the right to change at any time. • humor



Up Front • Ruby Is Twenty • Michael Swaine • Ruby Is Twenty.

Choice Bits • A Smattering of Nattering • Michael Swaine • We follow Twitter so you don't have to.

RubyMotion • Expressive iOS Development • Clay Allsopp • Clay shows how Ruby and Objective-C can coexist in iOS app development. • ruby, iOS, objective-c

Deploying with JRuby in the Cloud • Adjusting the Settings on Your Java Machine • Joe Kutner • If you've been hesitant to switch to JRuby due to lack of familiarity with the JVM, Joe has some good news for you. • java, ruby, cloud

Estimation is Evil • Overcoming the Estimation Obsession • Ron Jeffries • What you don't know can hurt you, especially when you convince yourself that you do know it. • agile, estimation

The Five Cs of Offshore Communications • What You Don't Know Can Hurt You • Nick Krym • Offshore project success often comes down to communication. Chances are, you could be doing it better. • career

Launching a Gonzo Engineering Project • II: Just Do Something • Steven K. Roberts • The legendary gonzo engineer shares his secrets for pursuing crazy dreams and succeeding in this series. • hardware

Flashback • Bill Gates on Writing and Selling Basic to MITS • Michael Swaine • Mike was going through his files and found a 1983 interview with Bill Gates about writing and selling his first BASIC. • history

The Quiz • Code in Code • Michael Swaine • A monthly diversion at least peripherally related to programming. • puzzle

Shady Illuminations • Prediction • John Shade • Did the analysts who caused Apple's stock drop also cast Ashton Kutcher to play Jobs? • humor



Up Front • Passion and Pragmatism • Michael Swaine • This issue's about pursuing your dreams while earning a paycheck.

Choice Bits • What Twitter Is Good For • Michael Swaine • We follow Twitter so you don't have to.

What If You Don't Want to Pivot? • Sometimes It's Personal • Jonathan Rasmusson • Sometimes when the world tells you to pivot, you really need to stay the course. • agile

Launching a Gonzo Engineering Project • III: The Business Angle and Getting Educated • Steven K. Roberts • The legendary gonzo engineer shares his secrets for pursuing crazy dreams and succeeding in this series. • hardware

Finding the Geek Who Fits • Five Tips for Hiring as an Agile Team • Johanna Rothman • The team will have to work with the new hire. Shouldn't the team do the hiring? • career

Being the Geek Who Fits • Don't Forget That You're Interviewing Them, Too • Andy Lester • In a counterpoint to Johanna Rothman's article on finding the geek who fits your agile team, Andy looks at the situation from the prospective hire's point of view. • career

Uncle Bob and Functional Programming • A Reply to "Uncle Bob" Martin's "FP Basics Episode 3" • Paul Callaghan • Paul has been debating functional programming and test-driven development with "Uncle Bob" Martin. Here he shares the result of that conversation. • functional

Shady Illuminations • Clouds and Caves • John Shade • John is so distressed by the social nature of software today that he turns to poetry. • humor



Up Front • Strange Math and Gonzo Engineering • Michael Swaine • This issue's about strange math, gonzo engineering, and some ideas that may change your thinking.

Choice Bits • What the Cool Kids Are Tweeting • Michael Swaine • We follow Twitter so you don't have to.

Estimation • The Best We Can Do • Ron Jeffries • Two months ago in these pages, Ron Jeffries told us estimation is evil. Now he's back to tell us it's a necessary evil—and that, done right, it isn't even evil. • estimation, agile

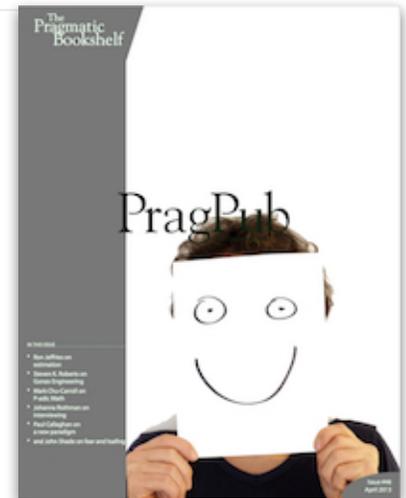
Launching a Gonzo Engineering Project • IV: The Media Dance • Steven K. Roberts • The legendary gonzo engineer shares his secrets for pursuing crazy dreams and succeeding in this series. • hardware

P-adic Math • Playing with Unreal Numbers • Mark Chu-Carroll • P-adic numbers are an alternative to the reals. They have real applications, but playing with p-adics is also math geek fun. • math

Questions Never to Ask a Candidate • Ask What You Want to Know • Johanna Rothman • If you're asking candidates any of these questions, you should stop now. Johanna tells you why. • career

Dependent Types • A New Paradigm? • Paul Callaghan • Paul digs deeper into dependent types, which can often replace tests in creating confidence in code. • functional, haskell, idris

Shady Illuminations • Fear Factor • John Shade • John faces his fear and hands it his car keys and wallet. • humor



Up Front • Functional Programming and Gonzo Engineering

• Michael Swaine

Choice Bits • Twitter Is About Sharing • Michael Swaine • We follow Twitter so you don't have to.

Rothman and Lester • Do You Need a Portfolio? • Johanna Rothman, Andy Lester • If applying for a job is a test, do you need to show your work? • career

A Functional Introduction to Lua • Creating Some Fireworks with Lua • Josh Chisholm • A lot of popular applications were written in Lua. Josh treats Lua as a functional language and makes some sparks fly. • functional, lua

Dependent Types, Part II • Tests and Proofs • Paul Callaghan • Paul continues last month's discussion of dependent types. • functional, haskell, idris

Java Active Objects • A Proposal • Allen Holub • Allen does an end run around the JCP to give Java capabilities it's missing. • functional, java

A Night at the Museum • ...Where Old Computers Refuse to Die • David Bunnell • Pioneers of personal computing gather for one amazing night at the Living Computer Museum. • history

Launching a Gonzo Engineering Project • V: Building the Team • Steven K. Roberts • The legendary gonzo engineer shares his secrets for pursuing crazy dreams and succeeding in this series. • hardware

Pub Quiz • He Says It's Simple • Michael Swaine • You only come into this pub for the games. Have at it. • puzzle

Shady Illuminations • The Death of Blinky • John Shade • John exposes the Blink Tag Conspiracy. • humor



Up Front • The Magic Elixir • Michael Swaine • Elixir, Rails, Javascript MVC frameworks, and the Unification algorithm.

Choice Bits • From the Twitterstream • Michael Swaine • We follow Twitter so you don't have to.

Rothman and Lester • Depth or Breadth? • Johanna Rothman, Andy Lester • Johanna and Andy discuss managing your career over time. • career

Programming Elixir • A Gentle Introduction • Dave Thomas • Elixir is a modern, functional programming language designed for high availability and concurrency. It has Ruby-like syntax married to the power and reliability of the Erlang VM. If you wanted to get into functional programming but were put off by the academic feel, now's the time to jump in. • functional language, elixir

Keeping Rails on the Rails • System Testing Rails Itself • Sam Ruby • How the system that keeps the Rails book on track also aids in the development of Rails itself. • testing, books, frameworks, rails

Beyond the DOM • Getting the Most out of JavaScript MVC Frameworks • Stephen Rylander • JavaScript MVC frameworks are here to stay. Here Stephen introduces Knockout, one of the best. • frameworks, javascript, web

Unification • It's Not Just for Prolog • Paul Callaghan • A technique developed for automatic theorem proving back in the 1960s proves to be a gateway to many other applications built around matching. • functional, haskell, language

Puzzle • Puzzle In Puzzle • Michael Swaine • A short session in mental calisthenics.

Shady Illuminations • The Phish Cheer • John Shade • Phishing has John feeling like he's living in a dream world—only it's not his dream. • humor



Up Front • From the Secrets of GitHub to Lambdas in C++ • Michael Swaine • No, the guy on the back page isn't David Crosby. Although we can see why you might think so.

Choice Bits • From the Twitterstream • Michael Swaine • We follow Twitter so you don't have to.

Rothman and Lester • Responsible Networking • Johanna Rothman, Andy Lester • Johanna and Andy share their networking secrets. • career

Programming Elixir • Part 2: Functions and Pipelines • Dave Thomas • Elixir is a modern, functional programming language designed for high availability and concurrency. It has Ruby-like syntax married to the power and reliability of the Erlang VM. If you wanted to get into functional programming but were put off by the academic feel, now's the time to jump in. • elixir, functional, language

Why Lisp Still Matters • Getting Clojure • Michael Bevilacqua-Linn • This issue begins a series on the Clojure language by Michael Bevilacqua-Linn. • clojure, functional, language, lisp

An Interview with Tom Preston-Werner • Chatting with the CEO of GitHub • Jack Kaufman • The co-founder of GitHub chats about being an entrepreneur. • startups

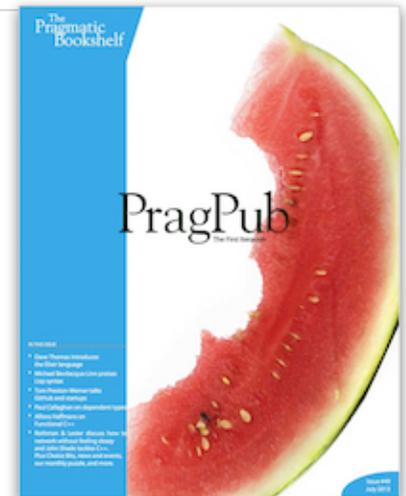
Dependent Types III • From Properties into Types • Paul Callaghan • In this final installment in his series on functional programming, Paul shows how much power can be packed into types. • haskell, functional, language

Living with Lambdas • Functional Programming in C++ • Alfons Haffmans • You can work in the functional programming paradigm in C++. And you may be surprised at how complete C++'s support for functional programming is. • c++, functional, language

Puzzle • A Sudoku-Anagram Mashup • Michael Swaine • A short session in mental calisthenics. • puzzle

Shady Illuminations • Call It C++++ • John Shade • John doesn't exactly review a book. • humor

Rear Window • A Parting Shot • Michael Swaine • The personal computer revolution had its roots in 1970s counterculture. • history



Up Front • The Second Iteration • Michael Swaine • It's our 50th issue and the beginning of a new era.

Choice Bits • From the Twitterstream • Michael Swaine • We follow Twitter so you don't have to.

Rothman and Lester • Articulating Value • Johanna Rothman, Andy Lester • Johanna and Andy talk about how to articulate your own value in a résumé. • career

Programming Elixir • Part 3: Processes and Messages • Dave Thomas • Elixir is a modern, functional programming language designed for high availability and concurrency. It has Ruby-like syntax married to the power and reliability of the Erlang VM. If you wanted to get into functional programming but were put off by the academic feel, now's the time to jump in. • elixir, functional, language

Identity, Value, and State in Clojure • Clojure's Answer to Unconstrained Mutability • Michael Bevilacqua-Linn • Michael looks at Clojure's novel philosophy on identity, value, and state. • clojure, functional, language

Finding the Joy in Legacy Code • It's Out There • Jeff Foster • The people who wrote that code knew more than you do about the problem they were solving. • testing

Big Problems • ...and Data Wasn't the Only One • Jesse Anderson • A cautionary tale. • big data

3D Graphics with SceneKit • Scaling Complexity • David Rönnqvist • The ramp from 2D to 3D is steep, but a really good framework helps a lot. • cocoa, graphics, objective-c

Puzzle • A Sudoku-Anagram Mashup • Michael Swaine • A short session in mental calisthenics. • puzzle

Shady Illuminations • Implementing Alistair Cockburn • John Shade • John embraces the metaphor of the software developer as a downhill skier, and encourages developers to embrace the trees they'll encounter on their downhill run. • humor

Rear Window • A Parting Shot • Michael Swaine • What Ed Roberts, founder of MITS, did after creating the personal computer. Photo by Paul Freiburger. • history



On Tap • The Play Issue • Michael Swaine • Putting the fun back in Web development.

Choice Bits • From the Twitterstream • Michael Swaine • We follow Twitter so you don't have to.

Rothman and Lester • Preparing for an Interview • Johanna Rothman, Andy Lester • The key to a successful interview is preparation. Johanna and Andy discuss how to walk in the room prepared to nail it. • career

Playing with the Play Framework • Bring the Fun Back into Web Development • Nilanjan Raychaudhuri • Time to play? Get the scoop on Play Framework, which has been getting a lot of attention for how easy it makes it to build web applications with Java and Scala. • frameworks, scala, java, web

Concurrent Programming in Clojure • Running in Parallel • Michael Bevilacqua-Linn • Michael continues his exploration of Clojure with a look at its advantages for concurrent programming. • clojure, functional, language

The Automated Tester • Write. Code. Automate. • Jonathan Rasmusson • Don't let yourself be limited by yesterday's job definitions. • testing

Hadoop • A Few Good Use Cases • Jesse Anderson • Solving problems for traveling salesmen and other overburdened folks. • big data

Guest Column • A Brief, Incomplete, and Mostly Wrong History of Programming Languages • James Iry • John Shade is on vacation and James Iry is filling in. • humor, history, language

Puzzle • A Sudoku-Anagram Mashup • Michael Swaine • A short session in mental calisthenics. • puzzle

Rear Window • A Parting Shot • Michael Swaine • The front and back covers of Ted Nelson's Computer Lib, the inimitable book that served as the manifesto for a revolution. • history



On Tap • Polymorphism • Michael Swaine • What's in this issue.

Choice Bits • Antonio on Books and Drips from the Twitterstream • Michael Swaine, Antonio Cangiano • Antonio Cangiano joins the Choice Bits lineup with all the new tech books. • books

Rothman and Lester • The Nuts and Bolts of Career Development • Johanna Rothman, Andy Lester • Johanna and Andy share tips on career development and becoming a T-shaped person. • career

Clojure and Polymorphism • Taking on Different Forms • Michael Bevilacqua-Linn • Java and Clojure have different approaches to polymorphism. Understanding how Clojure uses polymorphism is critical to Clojure mastery. • clojure, functional, language

The Luminus Framework • This Clojure-based Micro-framework Cuts to the Chase • Dmitri Sotnikov • How to build a web application quickly using Clojure and the Luminus micro-framework. • clojure, frameworks, functional, language, web

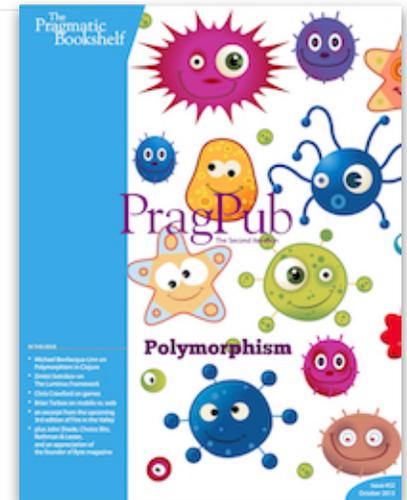
Game Plan • How to Discover the Natural Dimensions of Game Appreciation • Chris Crawford • A legend in computer game design offers a project that could make games better. • games

Mobile Applications vs. Web Pages • Competing Interests vs. User Experience • Brian Tarbox • Are there service for which you have both the mobile app and a link to the web site? Is there a hint there that something is not right? • mobile, web

Creating the Apple][• A Story from Personal Computing's Early Days • Michael Swaine • Jobs in tears, Woz playing pranks, and the ad in Playboy: the crazy days leading up to the release of the Apple][. • history

Shady Illuminations • Career-View Mirror • John Shade • John gives career advice, which you are well advised to ignore. • humor

Rear Window • A Parting Shot • Michael Swaine • Wayne Green died this summer, and with his death closes a chapter in personal computer history. Here, in brief, is the story of the birth of Byte magazine. • history



On Tap • Getting Clojure, but Feeling a Little AWKward • Michael Swaine • Languages old and new, career and productivity wisdom from experts.

Choice Bits • Chosen just for You • Michael Swaine • Flotsam on the Twitterside, tech books of note, and other goodies we snared in our net. •

Rothman and Lester • Unhappy at Work? • Johanna Rothman, Andy Lester • Should I stay or should I go? • career

Namespaces in Clojure • Organizing Code into Simpler Building Blocks • Michael Bevilacqua-Linn • Michael gives you “just enough about namespaces to get you going.” • clojure, functional, language

Rediscovering Awk • Sometimes Older Programs are the Right Tool for the Job • Derrick Schneider • A tedious task sends Derrick digging through the musty man pages of an archaic language, where he unearths some gems. • language, command-line, awk

Good and Bad Technical Debt • And How TDD Helps • Henrik Kniberg • Sometimes technical debt is good. The trick is knowing how much and when. • testing

Seven Coaching Patterns • Tapping Human Potential • Portia Tung • Talk less, listen more, and other habits to cultivate. • agile, coaching, patterns

Nostalgia for the Future • A Story from Personal Computing’s Early Days • Michael Swaine, Paul Freiberger • An excerpt from the upcoming *Fire in the Valley* 3rd edition. • history

Shady Illuminations • The Post-Post-PC Era • John Shade • John reflects on the Post-Post-PC Era, Jony Ive’s fingerprints, and just-in-time deconstructionist iconography. • humor

Rear Window • A Parting Shot • Michael Swaine • The first computer mouse, invented by Doug Engelbart at SRI in 1964, was carved from a block of wood. • history



On Tap • Wrapping Up the Year with Clojure • Michael Swaine • This issue has a Clojure theme, but there's lots more in this final offering for 2013.

Choice Bits • Chosen just for You • Michael Swaine • A month recaptured in tweets and news bits.

Rothman and Lester • How Do I Get My Boss to... • Johanna Rothman, Andy Lester • Want a raise? A new chair? Here's how to get it. • career

HealthCare.Gov • Rallying for a Software Hippocratic Oath • Robert Martin • Uncle Bob names the culprit for the botched rollout of the government healthcare site: it's you. • career

Exploring the Java Virtual Machine Version of Clojure • Dancing with Java • Michael Bevilacqua-Linn • Michael takes you on a whirlwind tour of Clojure's Java interop. • clojure, functional, language, java

Tracking Down Culprit Code in Clojure • An Exercise in Profiling • Dmitri Sotnikov • Learning Clojure? Here's a project that will give you hands-on experience with writing Clojure code. • clojure, functional, language

Making Music with Clojure • Live Coding is all the Rage at Raves • Sam Aaron • Admit it: your real mission in life is to modify code in real time and have it projected on screens in a nightclub. Welcome to the wild world of Livecoding. • clojure, functional, language, music

The Basics • A Story from Personal Computing's Early Days • Michael Swaine, Paul Freiburger • In the beginning was Basic. But even getting the Basic programming language onto those first microcomputers was a challenge. • history, basic

Antonio on Books • Antonio Cangiano • All the new tech books of note. • books

Shady Illuminations • A Few Wild Shots from the T-Shirt Canon • John Shade • John casts a jaundiced eye at the latest announcements from Wolfram Research and IBM. • humor, functional

Rear Window • A Parting Shot • Michael Swaine • The first programmer turns 198 this month. • history

