



The Guidebook



9:30AM - 10:30AM

Opening Keynote

Yukihiro Matsumoto (Matz)
[Grand Ballroom A]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

10:40AM - 11:25AM

Learning and Teaching

Building a Better OpenStruct

Ariel Caplan

OpenStruct, part of Ruby's standard library, is prized for its beautiful API. It provides dynamic data objects with automatically generated getters and setters. Unfortunately, OpenStruct also carries a hefty performance penalty.

Luckily, Rubyists have recently improved OpenStruct performance and provided some alternatives. We'll study their approaches, learning to take advantage of the tools in our ecosystem while advancing the state our community.

Sometimes, we can have our cake and eat it too. But it takes creativity, hard work, and willingness to question why things are the way they are.

Bootcamp Grads Have Feelings, Too

Megan Tiu

You're a bootcamp student. You're so excited to become a developer! Amidst your excitement about this new industry, you hear everyone say that bootcamps are a blemish on the community, that they're a waste of time and money. "Maybe I've made a huge mistake," you think. "I don't know how I'll fit in here."

But you can make this community better! In this session, you'll learn about the varied experiences of bootcamp students and grads, how exclusionary behavior can end up stunting our community as a whole, and what you can do to make a more inclusive environment for everyone of all skill levels.

Attention Rubyists: you can write video games

Cory Chamblin

Ruby may seem like a humdrum language best suited for writing web apps, but underneath that cool exterior lies plenty of power -- power we can harness for making our own games!

In this talk, we'll introduce Gosu, a sweet little game library. We'll talk about how games are structured in pretty much every language and framework ever, where to get ideas for things to make, and ways you can share your game with your friends and the world.

Comparative Ruby

Building maintainable command-line tools with MRuby

Eric Hodel

mruby and mruby-cli makes it possible to ship single binary command-line tools that can be used without setup effort, but how difficult is development in MRuby?

In this talk we'll explore how to work with MRuby and mruby-cli. We'll cover the differences in development patterns when using MRuby, the maturity of the MRuby ecosystem, and development patterns when using mruby-cli to build a command-line tool.

11:35AM - 12:20PM

Tools and Toys

C Ruby? C Ruby Go! Go Ruby Go!

Loren Segal

Ever wanted to rewrite performance sensitive code as a native Ruby extension, but got stuck trying to navigate the depths of Ruby's C API before you could get anything done? Or maybe you're just not comfortable with C and want an easier path.

Do you know any Go? Well, if you do, you're in luck! Join us for this talk about a tool named gorb that will quickly and easily let you generate native Ruby extension wrappers for your Go programs. And if you don't know Go, come anyway, it's really easy to learn! We'll have you writing blazing fast code that you can use right from Ruby, in no time at all.

Learning and Teaching

Continuing Education at Work

Katherine Wu

The list of things we want to learn is infinite. How many of us have marked items to read/watch, yet never go back to them? Despite the best of intentions, I often only learn what I directly need. It wasn't until I started running a couple lightweight continuing education programs at work that I followed through on my goals. We'll discuss strategies for making these programs low maintenance and long-lived, as well as flexible enough to help both more and less experienced folks. If you've been looking for a more effective approach to learning, but still outside classrooms, this talk is for you!

Comparative Ruby

Deletion Driven Development: Code to delete code!

Chris Arcand

Good news! Ruby is a successful and mature programming language with a wealth of libraries and legacy applications that have been contributed to for many years. The bad news: Those projects might contain a large amount of useless, unused code which adds needless complexity and confuses new developers. In this talk I'll explain how to build a static analysis tool to help you find unused code and remove it - because there's no code that's easier to maintain than no code at all!

Composition

James Dabbs

Our work as programmers consists largely of problem decomposition and solution recomposition. This talk is interested in how we cobble small units together into cohesive solutions. We'll examine and compare both object and functional composition, using a Haskell-inspired, functional style of Ruby. Along the way, we'll see how good functional principles can improve our object-oriented design, and vice versa.

12:20PM - 1:30PM

Lunch

[Grand Ballroom B]

1:30PM - 2:00PM

Just a Ruby Minute

Andrew Faraday

[Grand Ballroom A]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

2:10PM - 2:55PM

Tools and Toys

Learning and Teaching

Ruby Deep Dive

Comparative Ruby

Evaluate Ruby Without Ruby

Takashi Kokubun

Have you ever wanted to implement a tool that can be configured by Ruby code? You may think such tools require MRI, JRuby or Rubinius to evaluate the configuration. But that's not true! Let's see how we can build a tool that evaluates Ruby without dependency on Ruby interpreter using mruby.

"Am I senior yet?" Grow your career by teaching your peers.

Katlyn Parvin

"How do I become a senior engineer?" It's a question every bootcamp grad will ask. Most engineers look at advancement through a lens of increasing technical skill. More than ever, though, being "senior" means more than just parsing Ruby in your sleep. As our companies grow and as our industry grows, seniority means helping new teammates and colleagues advance their own skills. It means knowing how to teach.

You don't need Matz-level knowledge to be a great teacher. With social awareness, a dash of psychology, and some proven approaches, by helping others learn, you'll earn senior-level respect.

A Look at Hooks

Craig Buchek

Ruby has several methods that are invoked implicitly under certain circumstances. These methods are called "hooks", and they provide points to extend behavior. While hooks might seem like "spooky action at a distance", they can be really powerful. In fact, hooks are one of the primary ways that Ruby provides for meta-programming.

Unfortunately, Ruby's hooks are not all documented very well. We'll take a look at what hooks are available and how to use them. We'll also talk about when to avoid using hooks and provide some tips on how to troubleshoot when hooks are involved.

Why recursion matters

James Coglan

In modern programming, recursion is so common that we take it for granted. We work with recursive processes and structures every day, and it's easy to forget that recursion was once a highly contentious issue in programming language design.

But there's more to recursion than meets the eye. In this talk I'll discuss how it is fundamental to computing; how it can prove programs are correct, make them run faster, and even let you run programs backwards! Along the way we'll meet maths, Haskell, Bash, JavaScript, and our old friend Ruby doing some very surprising things.

3:05PM - 3:50PM

Tools and Toys

Ruby Deep Dive

Comparative Ruby

Improving Coverage Analysis

Ryan Davis

If you follow modern practices, test coverage analysis is a lie, plain and simple. What it reports is a false positive and leaves you with a false sense of security, vulnerable to regression, and unaware that this is even the case. Come figure out how this happens, why it isn't your fault, and how coverage analysis can be improved.

Dōmo arigatō, Mr. Roboto: Machine Learning with Ruby

Eric Weinstein

Machine learning is a popular and rapidly growing field these days, but you don't usually hear it mentioned in the same breath as Ruby. Many developers assume that if you want to do machine learning or computer vision, you've got to do it in Python or Java. Not so! In this talk, we'll walk through training a neural network written in Ruby on the MNIST dataset, then develop an application to use that network to classify handwritten numbers.

Keyword Args — the killer Ruby feature you aren't using

Guyren G Howe

Write clearer code more easily. Write functions that are more composable and flexible.

Ruby 2.0's KWArgs feature isn't just an easier way to write functions that take keyword arguments; KWArgs turn Ruby into a lean, mean Hash processing machine.

In this talk, you'll learn the details of how to write the next generation of Ruby code. Code that is easier to read, easier to write, more flexible and easier to change, all based on this one simple Ruby feature.

To Clojure and back: writing and rewriting in Ruby

Phill MV

Act 1 - State of Grace
After many years of Ruby, we built our app in Clojure, why not. Clojure is a breath of fresh air. Learn a lisp today!

Act 2 - The Fall
It's harder to succeed when you're in a new and different ecosystem. How Clojure contrasts with Ruby.

Act 3 - The Wilderness
Rewriting your app is a bad idea but we did it anyways. How we came to that decision.

Act 4 - Reconciliation
How I write Ruby has changed: a discussion.

3:50PM - 4:20PM

Break

[Sponsor Hall]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

4:20PM - 5:05PM

Ruby Deep Dive

Rainbows! Color Theory for Computers

Lito Nicolai

We often use color as a way to add information, whether in design, in UX, or for visualizations. When we visualize information, what's the best possible color scheme to use? How can we display the most possible information?

The only way to know is to explore the nature of color! We'll build up to the color-handling code that exists in 'graphics.rb', a Ruby-language visualizations library. For free, we'll end up with intuitive models of computer color spaces and tricks for how to think about common color concepts like gradients and paint mixing.

Problem Solved! Using Logic Programming to Find Answers.

Gavin McGimpsey

We love Ruby's object orientation, and you might have heard functional programming is the new hotness. But don't leave home without one more paradigm! Logic programs express relations and constraints, allowing us to work with incomplete information, build knowledge systems, and find outputs of a system given whatever inputs are available.

You'll see examples of logic programs using Ruby, learn about an algorithm that searches for a solution in a finite problem space, and explore potential applications for logic programming approaches, both in practical areas and for your mental models.

Lies, Damned Lies, and Substrings

Haseeb Qureshi

Generate all of the substrings of a string—a classic coding problem. But what's its time complexity? In many languages this is a pretty straightforward question, but in Ruby it turns out, it depends.

Follow me into the matrix as I explore copy-on-write optimization, how substrings are created in MRI, and eventually create a custom build of Ruby to try to speed up this classic coding problem. This talk will be a mix of computer science and a deep dive into how Ruby strings work in MRI.

Diving into the Details with DTrace

Colin Jones

"Argh, why is this so slow?!" We've all asked questions like this without getting satisfying answers. And so we slog along, finding workarounds and architecting around the slow bits.

But what if we had a friend who could see things we can't? Someone who could say whether our guesses were right, and what to look at next?

DTrace is that friend, and it's here to help! In this talk, I'll show you how DTrace can answer incredibly general and specific questions about systems performance, and help us to generate new ones. Along the way, we'll see that it's easier to use than you might think!

5:15PM - 6:00PM

Computer Science: The Good Parts

Jeffrey Cohen

[Grand Ballroom A]

9:30AM - 10:15AM

You Have the Empathy of a Goat: Documenting for the User

Tara Scherner de la Fuente
[Grand Ballroom A]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

10:25AM - 11:10AM

Ruby and Hardware

(m)Ruby on small devices

Shashank Daté

mruby is the lightweight implementation of the Ruby language for linking and embedding within your applications. This talk will show how it can be used on small resource constrained devices like Raspberry Pi Zero and exploring some techniques of memory and run-time optimizations.

Better Code Through Boring(er) Tests

Betsy Haibel

Your tests are annoying. Whenever you update your application code, you need to change a lot of them. You've built yourself some test macros, and some nice FactoryGirl factories, and installed this really cool matcher gem... and your tests are still annoying. What if you changed your application code instead? In this talk, you'll learn what "listening to your tests" means in the real world: no mysticism, no arrogant TDD purism, just a few practical refactorings you can use tomorrow at work.

From no OSS experience to the core team in 15 minutes a day

André Arko

Using and contributing to open source has been a cornerstone of the Ruby community for many years. Despite this strong tradition, it's hard to find anything collecting the likely advantages and costs of working on open source. This talk will introduce open source work, the benefits and costs of doing that work, and then provide a straightforward list of activities that can be done by anyone, no matter their level of experience with programming. Pick a project, schedule at least 15 minutes per day, join the core team. It's your destiny!

Lessons Learned

Finding your edge through a culture of feedback

Paulette Luftig

Have you ever wished for more feedback from colleagues to help you get better at your job? When's the last time you offered helpful feedback to someone else? Imagine an entire company fluent in the daily practice of giving and receiving constructive feedback. Would your experience improve? What does a team lose when feedback doesn't flow?

Feedback conversations can be difficult. But giving and receiving feedback pushes us to the edge of our growth potential, where the biggest payoffs occur. Join this session to grow your career by learning how to get real.

11:20AM - 12:05PM

Ruby and Hardware

Building HAL: Running Ruby with Your Voice

Jonan Scheffler

We've been living in the future for a full 15 years already and developers are still using antiquated technology like "keyboards" and "mice" to run their applications.

We're going to learn to use voice recognition to run our Ruby code so we won't need to depend on archaic plastic input methods to live our megalomaniacal dreams.

I think we can all agree that the world needs more robots listening to our every word, let's build an army of them and arm them with Ruby!

Surgically Refactoring Ruby with Suture

Justin Searls

The next feature means changing legacy code. The estimate just says "PAIN" in red ink. Your hands tremble as you claim the card from the wall.

For all of Ruby's breakthroughs, refactoring is as painful as 2004, when Feathers' book released. It's time we Make Refactors Great Again.

That's why I wrote Suture, a gem to help at every step:

In development, new code is verified against recorded calls
In staging, old & new code is ensured side-by-side
In production, unexpected errors fall back to the old code
With renewed confidence and without fear, you grab the card. You've got this.

Implementing Virtual Machines in Ruby (and C)

Eleanor McHugh

Most of us who've played games or worked in any one of a number of popular programming languages will have used virtual machines but unless we've taken a course in programming language design we probably have only a loose idea of what these are and how they work.

In this talk I'll look at the various parts necessary to model a computer-like machine in code, borrowing ideas as I go from real-world hardware design. We'll use a mixture of C and Ruby as our modelling languages: C is the lingua franca of the VM world whilst Ruby is the language which brought us monkey-patching...

Lessons Learned

My Meta Moments

John DeWyze

Meta-programming is alluring. To write code that writes more code sounds like the peak of efficiency, but learning it is filled with infinite loops and confusing stack traces. It is a place with lots of hair pulling and endless puts debugging.

Meta-programming in ruby can be intimidating. Yet, even with all that, it is fun. It is learnable. It is quintessentially ruby. If you're interested in getting your feet wet, come share my journey of learning things like method dispatch, BasicObject, class ancestry, and the joy of things you should think twice about before doing in production code.

12:05PM - 1:15PM

Lunch

[Grand Ballroom B]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

1:15PM - 2:00PM

Ruby and Hardware

How I Taught My Dog To Text Selfies

Greg Baugues

This talk is for the Rubyist who wants to get into hardware hacking but feels intimidated or unsure of where to start. The Arduino Yun is wifi-enabled and runs a stripped-down version of Linux. That means that you can build hardware hacks that execute Ruby scripts and interact with web APIs. In this talk, we'll start from scratch and live code a hardware hack that my dog uses to text me selfies using a webcam, Twilio, and a big red button.

Test Doubles are Not To Be Mocked

Noel Rappin

Test doubles (which you may know under the alias "mock objects") are the most misunderstood and misused testing tools we've got. Starting with the fact that nobody can agree on what to call them. Contrary to what you may have heard, test doubles do not inherently lead to fragile tests. What they do is hold up a harsh mirror to the assumptions in our code. They are the light saber of testing tools: a more elegant weapon for a more civilized age. But be careful not to cut your code in half, so to speak. Herein: a guide to using test doubles without losing your sanity.

Performance

Ruby 3 Concurrency

Koichi Sasada

Learn from Koichi about the work he's doing to bring a new concurrency model to Ruby 3!

Lessons Learned

Why Is Open Source So Closed?

Ra'Shaun Stovall

Why is Open Source So Closed? With the rapidly increasing amount of students coming out of bootcamp schools we have now created a gap within our communities of the "haves", and the "Looking for job"s. Being the organizer of New York City's 4,500+ member Ruby community with NYC.rb I have discovered ways we can ensure the generations of rubyists after us have a path paved before them. "Cyclical Mentorship" is the answer. Best part is we will know individually how we can immediately begin the feedback loop of not computers, but people!

2:10PM - 2:55PM

Ruby and Hardware

It's More Fun to Compute

Julian Cheal

Come with us now on a journey through time and space. As we explore the world of analog/digital synthesis. From computer generated music to physical synthesisers and everything in between.

So you want to write music with code, but don't know the difference between an LFO, ADSR, LMFAO, etc. Or a Sine wave, Saw wave, Google wave. We'll explore what these mean, and how Ruby can be used to make awesome sounds. Ever wondered what Fizz Buzz sounds like, or which sounds better bubble sort or quick sort? So hey Ruby, let's make music!

Ruby For Home-Ec

Adam Forsyth

Come learn how to design your own algorithms and code to solve problems around the house. Trying to use your scrap wood efficiently? Want to sort your pantry to maximize variety? I'll talk about the problem-solving process, walk through code, and touch on the computer science behind it all.

Performance

Ruby's C Extension Problem and How We're Solving It

Chris Seaton

Ruby's C extensions have so far been the best way to improve the performance of Ruby code. Ironically, they are now holding performance back, because they expose the internals of Ruby and mean we aren't free to make major changes to how Ruby works.

In JRuby+Truffle we have a radical solution to this problem - we're going to interpret the source code of your C extensions, like how Ruby interprets Ruby code. Combined with a JIT this lets us optimise Ruby but keep support for C extensions.

Ruby, Red Pandas, and You

Sean Marcia

Red pandas are adorable, playful, curious, and super cute. Unfortunately, they are in serious trouble. Over 50% of red panda newborns born in captivity do not survive long enough to leave their den and no one knows why. Come find out why red pandas are so amazing, how I met a Smithsonian Zoo researcher studying this and how we're solving this important problem using Ruby and machine learning. You will also leave this talk knowing how you can get involved (no matter your skill level) with great projects like this.

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

3:05PM - 3:50PM

Ruby and Hardware

Programming in the Small: Kids, Chickens, and Ruby

Coraline Clark, Jason R. Clark

After several years of programming in Ruby using Shoes, my daughter and I were hunting for a new project. Something more useful than a game. Something with a real-world connection. Then it struck us: Chickens!

Join us as we show you how we built our coop monitoring system. It'll be a wild ride of hardware hacking, weather-proofing, and father-daughter bonding, with Ruby sprinkled throughout. You'll learn how to modernize your surroundings, and about engaging the young people in your life in technology along the way.

JRuby Everywhere! Server, Client, and Embedded

Thomas E Enebo, Charles Oliver Nutter

Ruby has seen its heaviest on servers; Client-side Ruby has been limited to experiments and toys, and Ruby C extensions complicate embedded use. But there's a better way: JRuby. Using frameworks like JRubyFX and Shoes 4, among many GUI and graphics libraries, JRuby makes client-side development fast and powerful. With an embedded JVM, JRuby runs your app on mid-level embedded devices without compiling a line of code. And JRuby is still the best way to power up your Rails 5 app. Come learn how to use JRuby everywhere!

Performance

Slo Mo

Richard Schneeman

No one wants to be stuck in the slow lane, especially Rubyists. In this talk we'll look at the slow process of writing fast code. We'll look at several real world performance optimizations that may surprise you. We'll then rewind to see how these slow spots were found and fixed. Come to this talk and we will "C" how fast your Ruby can "Go".

The Little Server That Could

Stella Cotton

Have you ever wondered what dark magic happens when you start up your Ruby server? Let's explore the mysteries of the web universe by writing a tiny web server in Ruby! Writing a web server lets you dig deeper into the Ruby Standard Library and the Rack interface. You'll get friendlier with I/O, signal trapping, file handles, and threading. You'll also explore dangers first hand that can be lurking inside your production code- like blocking web requests and shared state with concurrency.

3:50PM - 4:20PM

Break

[Sponsor Hall]

4:20PM - 5:05PM

Ruby and Hardware

Running Global Manufacturing on Ruby (among other things)

Lee Edwards

A few miles from this convention center, Teespring prints millions of short-run custom products every year from modern direct-to-garment printers and ships them all over the world. Like most companies, Teespring's architecture is polyglot, but the core business logic is in Ruby. We'll discuss how we use large at-scale manufacturing and production systems to help anyone, anywhere turn their ideas into reality.

The long strange trip of a senior developer

Brandon Hays

Have you ever felt like you are in the passenger seat of your career? By simply looking around and seeing how few 20+ year veterans you work with, you're actually staring our industry's sustainability problem right in the face. But the software industry needs you still contributing 20 years from now! Fret not, we can start fixing these problems, right now, in your own job.

Using in-depth research across companies of all sizes, you'll come away with a plan to start designing a sustainable career track to help you grow in skill, influence, and income now and long into the future.

Performance

Halve Your Memory Usage With These 12 Weird Tricks

Nate Berkopec

Heroku and AWS hate him! Want to reduce your server bills by running more processes per server but running out of RAM? Ever wonder why that poor Ruby application of yours is using hundreds of megabytes of memory anyway? Not sure if your application has a memory leak or just bloat? Don't even know the difference? We'll talk about what causes memory usage, how to debug bloated memory and memory leaks, and how to write faster Ruby programs by decreasing the amount of memory they need.

Grow Your Team In 90 Days

Cheryl Gore Schaefer

So you want to work with an awesome ruby developer? Great! But finding one is hard. It's much easier to hire that apprentice or bootcamp grad, but that person is not an awesome developer. How do you help them get there?

You will learn how to build a learning plan and become the best mentor to a budding software developer. You have the opportunity to mold your apprentice's skill set into exactly what you need, while benefiting from their diverse skills and experiences. We will build an example learning plan of a developer learning Ruby, showing clear and specific choices and likely outcomes.

5:15PM - 6:45PM

LIGHTNING TALKS

[Grand Ballroom A]

9:30AM - 10:15AM

Ruby versus the Titans of FP

Cassandra Cruz
[Grand Ballroom A]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

10:25AM - 11:10AM

Ruby Deep Dive

How I Corrupted Survey Results and (Maybe) Ruined a Business

Mike Calhoun

It was the perfect storm of events and circumstances: a first job, naïveté of experience, a fear of getting fired, and a loud boss prone to yelling. One morning, I realized that the first web “project” of my first job in my new career had gone horribly off track. What came to pass in the following weeks was the most involved and tense coverup I’ve undertaken in my life. This experience can tell us all about how we communicate with each other. How we can create environments where people can ask for help and how an atmosphere of pressure and tension can ruin a business.

The Truth About Mentoring Minorities

Byron Woodfork

In the tech industry, we currently lack the ability to produce mentors who are able to effectively teach and connect with their minority protégés. In this talk, we discuss what changes we can make in the way we mentor to give our minority protégés the best chance to succeed. We will take a look at some of the struggles that I faced throughout my software developer apprenticeship, and pinpoint the sources of my success. In conjunction, we will also look at various case studies of other minority professionals who were both successful and unsuccessful in their attempts to climb the corporate ladder.

Methods of Memory Management in MRI

Aaron Patterson

Let’s talk about MRI’s GC! In this talk we will cover memory management algorithms in MRI. We will cover how objects are allocated and how they are freed. We will start by looking at Ruby’s memory layout, including page allocation and object allocations within those pages. Next we’ll cover collection algorithms used by MRI starting with the mark and sweep algorithm, followed by generational collection, and the tri color abstraction. Finally we’ll cover experimental developments for the GC like heap splitting. Expect to leave this talk with heaps of pointers to add to your remembered set!

The Neuroscience and Psychology of Open Source Communities

Elizabeth Barron

Because people are complex, diverse creatures, ensuring that your open source community is healthy, vibrant, and welcoming can be challenging. The good news is, science can help you understand human thoughts and behaviors, and the complexities of interacting in a collaborative way online. We’ll discuss cognitive biases, the SCARF collaboration model, the online community life cycle, and how these things interact. You’ll come away from this talk with a deeper understanding of yourself, your fellow humans, & the knowledge to improve personal interactions and the open source communities you belong to.

11:20AM - 12:05PM

Life Beyond Bootcamps

Ruby Deep Dive

Weird Ruby

Datacenter Fires and Other "Minor" Disasters

Aja Hammerly

Most of us have a “that day I broke the internet” story. Some are amusing and some are disastrous but all of these stories change how we operate going forward. I’ll share the amusing stories behind why I always take a database backup, why feature flags are important, the importance of automation, and how having a team with varied backgrounds can save the day. Along the way I’ll talk about a data center fire, deleting a production database, and accidentally setting up a DDOS attack against our own site. I hope that by learning from my mistakes you won’t have to make them yourself.

Becoming a Mid: Two Perspectives on Leveling Up

Kimberly D. Barnes, Kinsey Ann Durham

What does becoming a mid-level developer mean? How can a junior set goals and make steps to achieve this? It’s difficult to shed the title of ‘junior’, not only in your own mind, but in the minds of others. It is important to keep progressing in your career as a developer to be able to level up and no longer be seen as the junior on the team. Kim and Kinsey will provide two perspectives to enable you to leave this talk with tangible action items. They will also dive into what employers can do to help build frameworks to allow for this transition.

Optimizing ruby core

Shyouhei Urabe

I made ruby interpreter 10 times faster. Let me show you how.

Metaprogramming? Not good enough!

Justin Weiss

If you know how to metaprogram in Ruby, you can create methods and objects on the fly, build Domain Specific Languages, or just save yourself a lot of typing. But can you change how methods are dispatched? Can you decide that the normal inheritance rules don’t apply to some object?

In order to change those core parts of the language, there can’t be much difference between how a language is implemented and how it’s used. In this talk, you’ll make that difference smaller, building a totally extensible object model on top of Ruby, using less than a dozen new classes and methods.

12:05PM - 1:15PM

Lunch

[Grand Ballroom B]

[Grand Ballroom A]

[Junior Ballroom B]

[Junior Ballroom C]

[Junior Ballroom D]

1:15PM - 2:00PM

Life Beyond Bootcamps

Ruby Deep Dive

Weird Ruby

The Building Built on Stilts

Nickolas Means

In the summer of 1978, structural engineer William LeMessurier got a phone call that terrified him. An undergraduate student claimed that LeMessurier's acclaimed 59-story Citicorp Center in Manhattan, just completed the year prior, was dangerously unstable under certain wind conditions. The student was right, and it was almost hurricane season.

The key to building a culture of innovation in your team is learning how to respond when mistakes inevitably happen. Let's let Bill LeMessurier teach us how to respond when it all goes wrong so that our creations can thrive despite our mistakes.

Learning Fluency

Sara Simon

All languages work in formulaic ways. Cracking these formulas takes discipline, time, creativity, trial and error. But is there an overarching formula to crack these formulas? Is there a designated set of steps we can take to guarantee fluency?

In this talk, you will learn about the methods people use to learn both foreign languages and programming languages. As developers, we often just jump in and start building. Why is this? Does full immersion work best always and for everyone? What is fluency, and is it ever something we can achieve in Ruby? Let's explore.

Seeing Metaprogramming and Lambda Function Patterns in Ruby

Lukas Nimmo

Metaprogramming and lambda functions in Ruby should be explained within their rightful place: living examples. You may have read tutorials on what these concepts are, but still do not understand when or why to use them. I dredged through over 50 prominent Open Source Ruby projects to bring you ten successful patterns that are used time and time again to create some of the most expressive and popular Ruby DSLs. Join me as we cover these patterns so that you can immediately begin using them in your own code to implement powerful DSLs. No vacuum-living, esoteric concepts here.

Rhythmic Recursion

Celeen Rusk

One of the best things about multi-disciplinary work is recognizing familiar ideas in a different setting. It's like running into an old friend while you're on vacation halfway around the world-- "I had no idea you'd be here! It's so great to see you!"

In this talk we'll run into our old friend recursion in the faraway land of minimalist music, by rewriting a piece of rhythmic music in Ruby.

2:10PM - 2:55PM

Life Beyond Bootcamps

Weird Ruby

Lies

Ernie Miller

All abstractions are lies, if they are abstractions at all, and as developers, we live our lives surrounded by them. What makes some abstractions better than others? This will be an opinionated and empowering look at the value and nature of abstractions, with a jaunt through quantum mechanics and the nature of reality.

You know, just your average, light discussion.

You graduated from bootcamp, now what?

Melanie Gilman

Throughout bootcamp, your biggest worry was finding a job. Now that you've found one and you've started your career as a developer, what comes next? In this talk, we'll explore what the career of a bootcamp graduate looks like a few years after the program. We'll talk about the good and not-so-good parts of being a newly-minted developer. We'll come away with actionable steps we can take to continue to grow as developers post-bootcamp and be happy and successful, even when we don't have the mythical perfect job.

Even the Justice League Works Remotely

Allison McMillan

"Remote welcome for senior level". This appears in countless job descriptions. Remote work is largely accepted in the developer community, but often only for very experienced developers. This fear is not without cause. Sometimes hiring remote developers at any level doesn't turn out well but there's generally a reason for that. Newer developers can also be very successful remote workers. In this talk, you'll learn what to look for when hiring remote developers at any level and non-senior developers will learn what characteristics to keep in mind in order to have a successful remote experience.

That Works?! Quines and Other Delightfully Useless Programs

Colin Fulton

Performance, readability and correctness are fine and dandy, but what happens when we start optimizing for whimsy, illegibility and outright silliness? Self-rewriting programs that also function as ASCII art? Yup. Rewrite any JavaScript program only using six characters? Why not?

Let's take a break from the practical and laugh at some of the most unbelievable code you've ever seen. Then let's pull out the magnifying glass to figure out how it actually works.

Learn how to read the unreadable and how to write code that—to borrow a phrase from the Ig Nobel Awards—makes people laugh, then think.

3:05PM - 3:50PM

Matz Q&A

Yukihiro Matsumoto (Matz)
[Grand Ballroom A]

3:50PM - 4:30PM

CLOSING SOCIAL
[Sponsor Hall]



Evan Phoenix
@evanphx

Evan is a Director at Ruby Central and long time Ruby user. Additionally, he works for Hashicorp on infrastructure and helps maintain Rubygems. Husband to Abby and father to Zoe and Kira, he loves spending time with his girls.



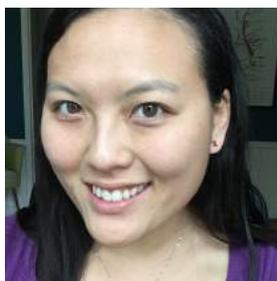
Marty Haught
@mghaught

Marty is a software architect/engineer building software for clients through his Colorado-based company, Haught Codeworks. He is active in supporting the Ruby community through Ruby Central and Boulder Ruby. A proud husband and father of two, Marty enjoys taking advantage of the Colorado outdoors year round.



Sarah Mei
@sarahmei

Sarah is a developer at DevMynd, focused on teaching clients the code and architecture changes that support team growth. She's a pair programming fan, a director of Ruby Central, a founder of RailsBridge, and a resident of foggy San Francisco.



Abigail Phoenix
@aphoenix

Abigail (Abby) Phoenix is the Executive Administrator of Ruby Central, which means she gets to reply to all of your lovely emails and write the checks, which are her two favorite things in life... besides tacos, obviously. Prior to her work with Ruby Central, Abby worked in event planning and arts management at the Annenberg Space for Photography, the Annenberg Foundation, and the Gates Foundation.



Heather Johnson

Heather is the Event Producer at Ruby Central. After years of planning and producing trade shows for several years, she decided to shift her focus to planning conferences and events. She loves working on DIY projects and spending time with her husband, daughter, and fur babies.



Shirley Bailes
@shirleybailes

Shirley heads up event marketing for NGINX, where she executes anywhere from 15-30 events annually. She spent a number of years cat-herding for a variety of O'Reilly conferences, and enjoyed a brief stint with the Ruby Central team as its Event Producer. Whenever she's not running around putting out (hopefully metaphoric) fires, she loves chatting with Ruby friends both new and old, so if you see her, please come up and say hi!

The Program Committee

Evan Phoenix
@evanphx

Marty Haught
@mghaught

Sarah Mei
@sarahmei

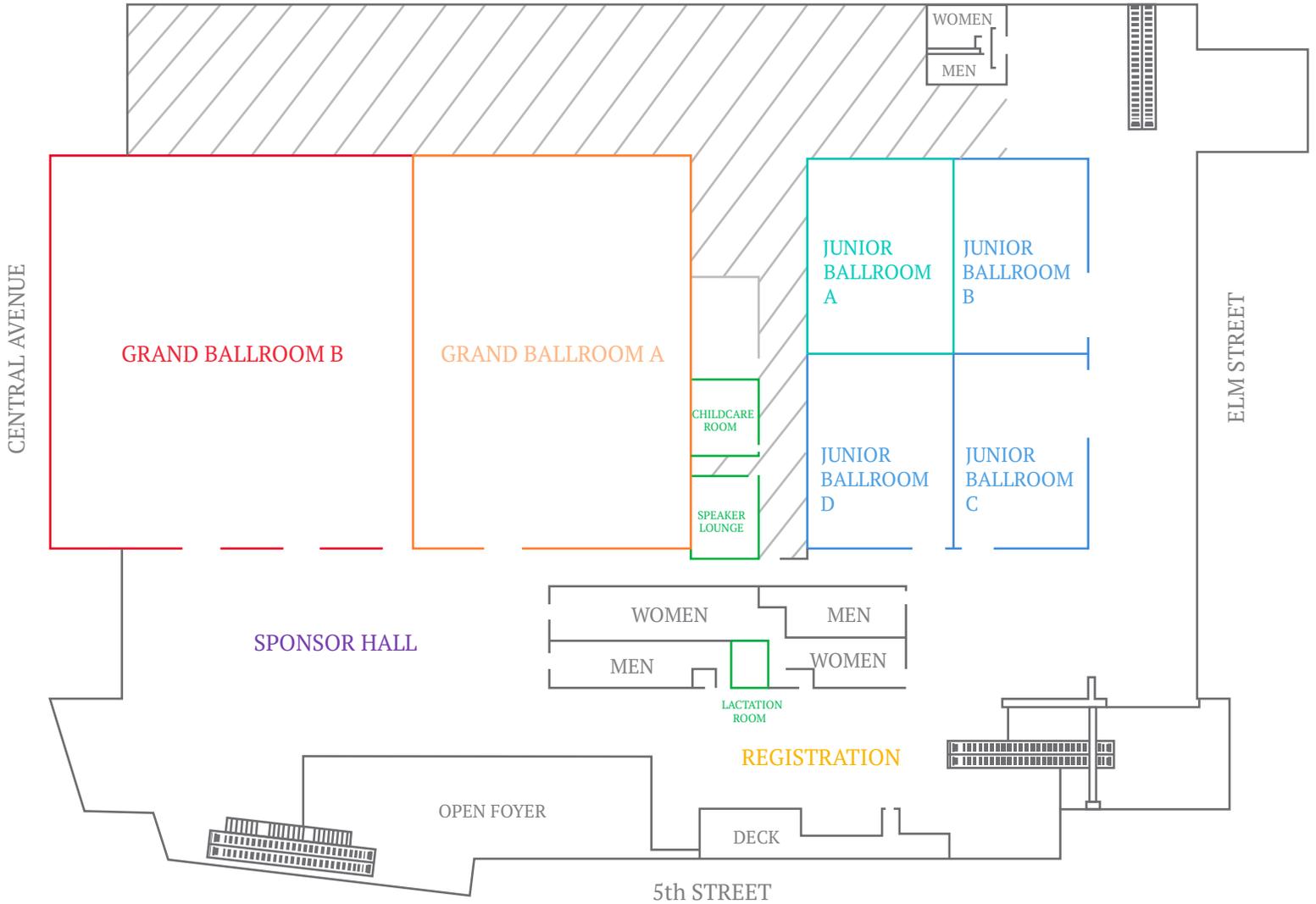
Aja Hammerly
@the_thagomizer

Hsing-Hui Hsu
@SoManyHs

Jeremy Hinegardner
@copiousfreetime

Matt Polito
@mattpolito

Sam Phippen
@samhippen



Keynote & Breakout

GRAND BALLROOM A

Lunch

GRAND BALLROOM B

Breakouts

JUNIOR BALLROOM B
JUNIOR BALLROOM C
JUNIOR BALLROOM D

Staff Room

JUNIOR BALLROOM A

Utility Rooms

CHILD CARE ROOM
LACTATION ROOM
SPEAKER LOUNGE



Points of Interest

- 1 Conference Venue: Duke Energy Convention Center
- 2 Conference Hotel: Hyatt Regency Cincinnati
- 3 RubyConf 5K start: Ohio River Trail under Taylor Southgate Bridge
- 4 Nearest Streetcar stop: E. 4th Street & Main Street
- 5 12th/Vine Streetcar Stop: Over-the-Rhine, many restaurants/bars nearby.
- 6 14th/Elm Streetcar Stop, Over-the-Rhine, many restaurants/bars nearby.
- 7 Elder/Elm Streetcar Stop, Findlay Market, classic Cincinnati public food market.

Walkable Restaurants

- A Maplewood Kitchen & Bar, known for brunch
- B Fountain Square: outdoor public square & ice rink, also features a Graeter's Ice Cream (local specialty!).
- C Taste of Belgium: great place to try goetta (local specialty!)
- D Moerlein Lager House: gastropub with tons of local beers
- E Cheapside Café: light cafe food, excellent coffee, including nitro cold brew



Thank you!

