

RubyRX

Marriott RTP

February 19 - 21, 2009

<http://www.nfjsone.com/conference/raleigh/2009/02/index.html>

Thu, Feb. 19, 2009

	Salon C	Salon D
7:00 - 8:00 PM	Registration	
8:00 - 9:00 PM	Lightning Talks	
9:00 - 9:30 PM	Mixer	

Fri, Feb. 20, 2009

	Salon C	Salon D
8:00 - 9:00 AM	Breakfast	
9:00 - 10:30 AM	Fear of Programming Nathaniel Talbott	Deployment with Capistrano David Bock
10:30 - 11:00 AM	Break	
11:00 - 12:30 PM	What's new in Ruby 1.9 Chad Fowler	Clojure Stuart Halloway
12:30 - 1:30 PM		
1:30 - 3:00 PM	ActiveRecord Deep Dive Chad Fowler	Homesteading: The New Entrepreneurial Model Matthew Bass
3:00 - 3:15 PM	Break	
3:15 - 4:45 PM	The Fall and Rise of Dynamic Programming Languages Rick DeNatale	Concurrent Programming with Clojure Stuart Halloway
4:45 - 5:00 PM	Break	
5:00 - 6:30 PM	Introduction to Erlang Kevin Smith	Writing Fast Ruby: Learn from Merb and Rails 3 Carl Lerche

Sat, Feb. 21, 2009

	Salon C	Salon D
8:00 - 9:00 AM	Breakfast	
9:00 - 10:30 AM	"Design Patterns" in Ruby Neal Ford	Five Skills Every Freelancer Must Have Nathaniel Talbott
10:30 - 11:00 AM	Break	
11:00 - 12:30 PM	Advanced DSLs in Ruby Neal Ford	Tools for your Ruby Toolbox - Web Publishing Edition David Bock
12:30 - 1:30 PM		
1:30 - 3:00 PM	Merb and Rails3: What's Next? Yehuda Katz	Fuzz-Testing Rails Apps with Tarantula Glenn Vanderburg
3:00 - 3:15 PM	Break	
3:15 - 4:45 PM	Erlang For The Practicing Programmer Kevin Smith	Sinatra: Essential Simplicity Glenn Vanderburg
4:45 - 5:00 PM	Break	
5:00 - 6:30 PM	Rails Application Tuning and Performance monitoring David Bock	Meta-programming Ruby for Fun & Profit Neal Ford

RubyRX

Marriott RTP

February 19 - 21, 2009

<http://www.nfjsone.com/conference/raleigh/2009/02/index.html>

Homesteading: The New Entrepreneurial Model by Matthew Bass

Many years ago a community would come together to help a neighbor get started. They'd all pitch in, knowing that when they needed help, those same friends would come to their aid as well. This model of community self-sufficiency has been recycled in the software world.

Deployment with Capistrano by David Bock

Capistrano is the de-facto tool used to deploy Rails applications, but it is useful for so much more. In this talk we will publish a real website from scratch, with consideration for real world stuff like safeguarding passwords, turning on and off application monitoring, multiple machine deployments, and general system administration tasks.

Tools for your Ruby Toolbox - Web Publishing Edition by David Bock

Rails may be the framework that turned many of us on to Ruby, but if you are using it for all of your server-related Ruby projects, you probably have a hammer and are seeing every problem as a nail. There are a number of smaller, tighter solutions to problem in this space, including GServer (built into the Ruby libraries), StaticMatic and Webby (tools for generating a static site, but with all the templating goodness), Sinatra, a small server with an awesome DSL for restful web services, and Rack, an easy way to get content published as a web service.

Rails Application Tuning and Performance monitoring by David Bock

Most developers who adopt Rails deploy their first application and see it get rushed under heavy load... thus the rumor that 'Rails doesn't scale'. Well, Rails does, you just have to know a few things

The Fall and Rise of Dynamic Programming Languages by Rick DeNatale

A war is being waged between the Empire and the Rebel Alliance. The imperial forces are laboring tirelessly to stamp out such dangerous ideas as late-binding, and automatic memory management. Based on long experience, such ideas are known to cause severe performance problems at best, and to bring down the Empire with errors at worst. The year, 2008 Ruby vs. Java*, or ca 1990 Smalltalk vs. C++, or ... The languages change, but the arguments remain. After a period in the shadows, dynamic languages have been on the rise again. Ruby has been at the forefront of this resurgence, with older dynamic languages tagging along. At the same time, hard-learned lessons from the implementation of these earlier languages are starting to have an influence on implementation of the newer ones. Witness the buzz in the Ruby community about "repurposing" projects such as MagLev.

"Design Patterns" in Ruby by Neal Ford

The Gang of Four book was actually 2 books: a nomenclature describing common software problems and a recipe book for solutions. The vocabulary they defined is still useful. The recipes are a disaster! Ruby has powerful meta-programming facilities far beyond the languages encompassed by the GoF. It turns out that many of the structural design patterns in the Gang of Four book and beyond are much easier to solve with meta-programming.

Advanced DSLs in Ruby by Neal Ford

Building Domain Specific Languages in Ruby shows the power and flexibility of the language.

Meta-programming Ruby for Fun & Profit by Neal Ford

Ruby is the revenge of the Smalltalkers. Not since Smalltalk has a language had such powerful meta-programming facilities. While this may seem like a minor feature, it turns out that surgical meta-programming allows solutions to problems that are clearer, more concise, more maintainable, and take orders of magnitudes fewer lines of code.

What's new in Ruby 1.9 by Chad Fowler

Ruby 1.9 the first major upgrade to the Ruby language in many years. On top of a slew of new language features, it offers an exponentially faster, byte-code compiled runtime environment.

ActiveRecord Deep Dive by Chad Fowler

ActiveRecord is one of the leakiest abstractions you'll ever use as a software developer. Becoming a true master of ActiveRecord requires you to embrace that fact and dig into its guts. Knowing how ActiveRecord really works makes you a much better ActiveRecord programmer.

Clojure by Stuart Halloway

In recent years, the Java community has embraced a variety of new languages that target the JVM, but also offer productivity advantages over traditional Java coding.

Concurrent Programming with Clojure by Stuart Halloway

Clojure is dynamic language for the Java Virtual Machine with several powerful features for building concurrent applications.

Merb and Rails3: What's Next? by Yehuda Katz

With the announcement that Merb will be merging into Rails3, you might have some questions about what that means for the future of Rails. Yehuda, the lead developer of the Merb project (now a Rails core team member), will talk about the work he's doing to bring modularity, performance, and a public API to Rails. The first Rails3 beta is scheduled for May of this year, so you can be one of the first to get a peek into the work the new Rails team is doing.

Writing Fast Ruby: Learn from Merb and Rails 3 by Carl Lerche

It has been said that Ruby is a slow language, but that is not true. Numerous Ruby projects have shown that it is possible to write fast, scalable software using Ruby. Merb, for instance, is faster than any major PHP web framework.

Introduction to Erlang by Kevin Smith

Spend an hour learn about Erlang -- a functional programming language designed for demanding multi-threaded environments. You'll get an introduction to the language including syntax and major concepts. You'll also learn which companies and projects are using Erlang today and the kinds of problems they're solving with Erlang.

Erlang For The Practicing Programmer by Kevin Smith

Learn how to use Erlang in a practical setting. This talk will cover building and deploying a simple, but complete, web application using Erlang. The talk will illustrate how to use the OTP framework to write robust servers and how to leverage third party libraries, like Mochiweb, to make developing for the web easier.

Fear of Programming by Nathaniel Talbott

We developers spend a lot of time talking about how to improve technically at our craft, how to write better code, how to be more productive when we're writing code. But what about when we don't feel like coding? What about the emotions that often keep us from putting our fingers on the keyboard and working on that cool library we want to write or finishing up that cool side project we were so fired up about a few weeks ago? Or to put it another way, why is it that some developers turn out so much more code than the rest of us? How do they maintain five open source projects (or more!) while we might be doing good to make progress on one? It certainly might be that they're smarter, or that they spend more time at it. But maybe not... Maybe the reason we don't create more is that we're afraid. Afraid of not finishing, afraid of what others will say, afraid we won't know how to solve a problem, afraid that we're not working on the right thing. If creating code is even partly an act of artistry, then it's worth examining our emotional connection with what we're making. It might be time to start realizing that coding productivity is affected as much by our emotional outlook as it is by our technique.

Five Skills Every Freelancer Must Have by Nathaniel Talbott

Lots of developers decide to try going it on their own - perhaps they're tired of being a cube rat, having technical decisions made by uninformed management, or having to service the bureaucracy before the client. Rubyists seem to have an especially high likelihood of going independent, since it gives them much more flexibility to use the language that makes them happy. But as alluring as it might be to "be your own boss", everyone who makes the jump from the cube to the room over the garage finds out real quick that

there's a whole new set of skills necessary to be successful. Even worse, freelancing doesn't come with a training manual that outlines all these important skills, and most folks just have to learn them the hard way: by trial and error.

Fuzz-Testing Rails Apps with Tarantula by Glenn Vanderburg

Tarantula is a Rails plugin for doing easy, repeatable fuzz testing of Rails applications. Tarantula crawls your application, supplying random inputs at every turn and looking for signs of trouble. It can be used to regularly check for many kinds of error handling issues, cross-site scripting and SQL injection vulnerabilities, and more mundane things like invalid HTML. This talk will discuss how Tarantula works, how to use it, and how to make it a regular part of your testing strategy.

Sinatra: Essential Simplicity by Glenn Vanderburg

Sinatra is a Ruby-based framework that makes it trivial to write simple web apps very quickly. It's much lighter weight than Rails and fills a different need, but a lot of very smart people are using this exciting new tool.