



# The End Of Java?

Spring 2006

# The Premise: Change is afoot

- Simple web programming with Java is more difficult and more annoying
- New frameworks, in 'new' languages, are emerging
- The 'Big Names' are leaving
- The Ruby On Rails Buzz

# Some New Frameworks

- Ruby On Rails (Ruby)
- Django (Python)
- Seaside (Smalltalk)
- Cake (PHP)

# The Features

- Quick feedback
- ORM
- Full stack
- Convention over configuration

# The Big Names are Leaving

- Bruce Tate ("Better, Faster, Lighter Java", "Bitter Java", "Bitter EJB") - Ruby On Rails
- James Duncan Davidson (Ant, Tomcat) - Python, Ruby, Objective C
- Steve Yegge (GeoWorks, Amazon, Wyvern MMOPG) - Dynamic Scripting Languages
- Martin Fowler ("Refactoring", ThoughtWorks) - Ruby

- “The Hottest Hacker on Earth”
- RailsConf
- Blog Buzz
- Java Community Reaction



WEB SITES

## The Hottest Hacker on Earth

David Heinemeier Hansson could be mistaken for a Gap model. The Danish 26-year-old's blog is filled with glam photos of himself, his friends at parties, and his hot girlfriend. But Hansson is actually the first Google-O'Reilly hacker of the year (2005), an über-geeky nod for producing Ruby on Rails.

RoR establishes a simple, open source framework for using the equally simple programming language Ruby. In RoR, commands look like sentences, and limited choices make it tough for Web developers to get lost or tangled up in code. As a partner at 37signals in Chicago, Hansson uses RoR to write software for the popular project-management tools Basecamp and Backpack.

RoR also makes it easy for developers to work with AJAX (short for Asynchronous JavaScript and XML), the technology that lets them do cool stuff like move folders and scroll through maps on a Web page. "It's like a color-by-numbers system," says Derek Sivers, founder of CD Baby, who's rebuilding his entire site with RoR. And that's been Hansson's goal all along – to fundamentally remake the way Web sites are built. As he puts it, "We've already slaughtered a whole herd of holy cows." – Nicholas Thompson

# The First 'Perfect Storm'

- Client / Server, Deployment & Microsoft
- C++ Problems
  - Pointers
  - Nested Includes
  - DLL hell
  - Inheritance problems
  - Portability

# Along comes Java...

- The Language
  - Easy to learn
  - Easy to program
  - Well designed, familiar
  - Transitional OO
  - Well designed APIs

# Along comes Java... (cont'd)

- The JVM
  - Platform independence
  - Security
  - Garbage Collection
  - Published byte code specification

- The Internet
  - Applets
  - Built in networking
  - Servlet
  - The Community
- IBM
  - Open Source
  - Community Process

# What's wrong with Java?

- Web Programming with Java is hard
  - Huge frameworks
    - more productive
    - long learning curve
  - Many frameworks
    - persistence frameworks
    - middle tier frameworks
    - web frameworks
  - Versatility - the gift and the curse?

# Features of 'The Next Big Thing'

- Dynamic Typing for better productivity
- Rapid feedback loop
- User interface focus
- Dynamic class model - discover and change parts of class at runtime
- True OOP - no primitives, single root object
- Consistent - encourages neat and maintainable code

# Features of 'The Next Big Thing' (cont'd)

- Portability
- Security
- 'Internet Enabled'
- Strong community
- Embrace Open Source

# Puts 'Hello World'

(Introducing Ruby)

- Fully OO
  - no primitives, single root object, operators are methods
- 'Scripting' Language
- Dynamically typed
  - can add new classes and add / redefine methods at runtime
- Strongly typed

# Puts 'Hello World' (cont'd)

(Introducing Ruby)

- 'Natural' syntax
  - `1.day.from_now`
  - `for user in users`
- Built-in regular expressions
- Mixins
- Interceptors
- Emphasis on simplicity and design elegance
- Garbage Collected
- Multithreaded (OS independent)
- Portable

- “Ruby on Rails is an open-source web framework that’s optimized for programmer happiness and sustainable productivity. It lets you write beautiful code by favoring convention over configuration.

# What they're saying...

“Rails is the most well thought-out web development framework I’ve ever used. And that’s in a decade of doing web applications for a living. I’ve built my own frameworks, helped develop the Servlet API, and have created more than a few web servers from scratch. Nobody has done it like this before.”

*-James Duncan Davidson, Creator of Tomcat and Ant*

“It is impossible not to notice Ruby on Rails. It has had a huge effect both in and outside the Ruby community... Rails has become a standard to which even well-established tools are comparing themselves to.”

*-Martin Fowler, Author of Refactoring, PoEAA, XP Explained*

“After researching the market, Ruby on Rails stood out as the best choice. We have been very happy with that decision. We will continue building on Rails and consider it a key business advantage.”

*-Evan Williams, Creator of Blogger and ODEO*

“Ruby on Rails is a breakthrough in lowering the barriers of entry to programming. Powerful web applications that formerly might have taken weeks or months to develop can be produced in a matter of days.”

*-Tim O'Reilly, Founder of O'Reilly Media*

# Ruby on Rails Features

- MVC Framework
- "Convention over configuration"
- "DRY" – Don't Repeat Yourself
- Generators for creating fully featured
- "scaffolding"
- Easily testable, agile friendly

- Encapsulates Persistence
- Contains business methods
- Coupled to and derives attributes from database
- Declares relationships to other models

# The View

- HTML generation with `.rhtml` using eRB
- XML generation with `.rxml`
- Sophisticated component rendering (DRY)
- ActionView provides large number of common rendering functions

# ActionController

(Controller)

- Processes requests - maps methods to urls
- Interacts with models, forwards to views
- Meta-programming handles form (including nested forms) data easily

- AJAX
- ActionMailer
- ActionWebService

# Ruby On Rails - The Good

- Productivity
- Less Coding
- You **HAVE** to unit test
- Easy
- Web 2.0 friendly

# Ruby On Rails - The Bad

- IDE Support
- Transition from static typing
- I18N
- Templates?
- It's a struggle to get acceptance

# Things to consider before diving in

- Am I ready?
- Is my team ready?
- Is the company ready?
- Is the project the right size?
- Do I have the right type of database?
- Do I have the right culture?

# Further Reading

- Agile Web Development With Rails - Dave Thomas and David Heinemeier Hansson
- <http://www.rubyonrails.org>
- <http://www.rubyonrails.org/docs>
- <http://www.rubyonrails.org/applications>
- <http://www.loudthinking.com/arc/000479.html>  
(scalability)
- <http://www.virtuas.com/files/JavaWebFrameworkSweetSpots.pdf>

- Thomas, Dave & Heinemeier Hansson, David. Agile Web Development with Rails. The Pragmatic Bookshelf, Raleigh, 2005. ISBN 0-9766940-0-X
- Tate, Bruce. Beyond Java. O'Reilly, Beijing, 2005. ISBN 0-596-10094-9
- Fernandez, Obie. Workin' on the Rails Road. ThoughtWorks. 2005



# Contact Us

## Telephone

(888) 594-6260 – Sales  
(416) 304-0860 – Main

## Fax

(416) 946-1929

## Address

345 Adelaide Street W, 6th Floor  
Toronto, Ontario  
Canada  
M5V 1R5

## Email Contacts

General Info  
[jonahinfo@jonahgroup.com](mailto:jonahinfo@jonahgroup.com)

Sales  
[jonahsales@jonahgroup.com](mailto:jonahsales@jonahgroup.com)