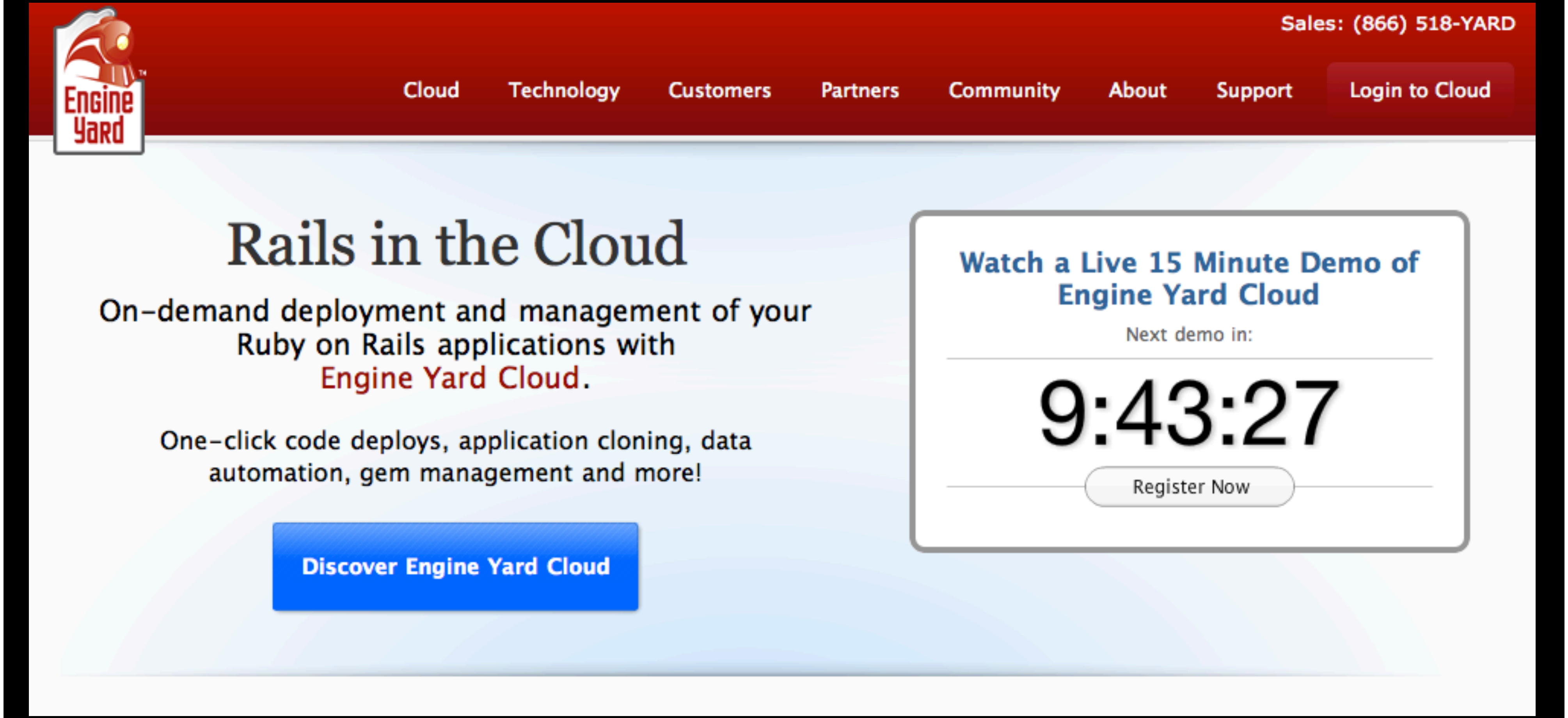
Your First Day with JRuby

Me

- Charles Oliver Nutter
- Java dev since 1997
- Rubyist, JRuby dev since 2004
- Full time JRuby dev since 2006
- headius@headius.com, headius on Twitter
- blog.headius.com

Engine Yard

- Engine Yard Cloud
  - C Ruby and JRuby (real soon)
  - Fire and forget
  - Slick management console
- JRuby developer support (real soon)
  - Inexpensive per-seat dev support



Sales: (866) 518-YARD

Cloud Technology Customers Partners Community About Support Login to Cloud

Rails in the Cloud

On-demand deployment and management of your Ruby on Rails applications with Engine Yard Cloud.

One-click code deploys, application cloning, data automation, gem management and more!

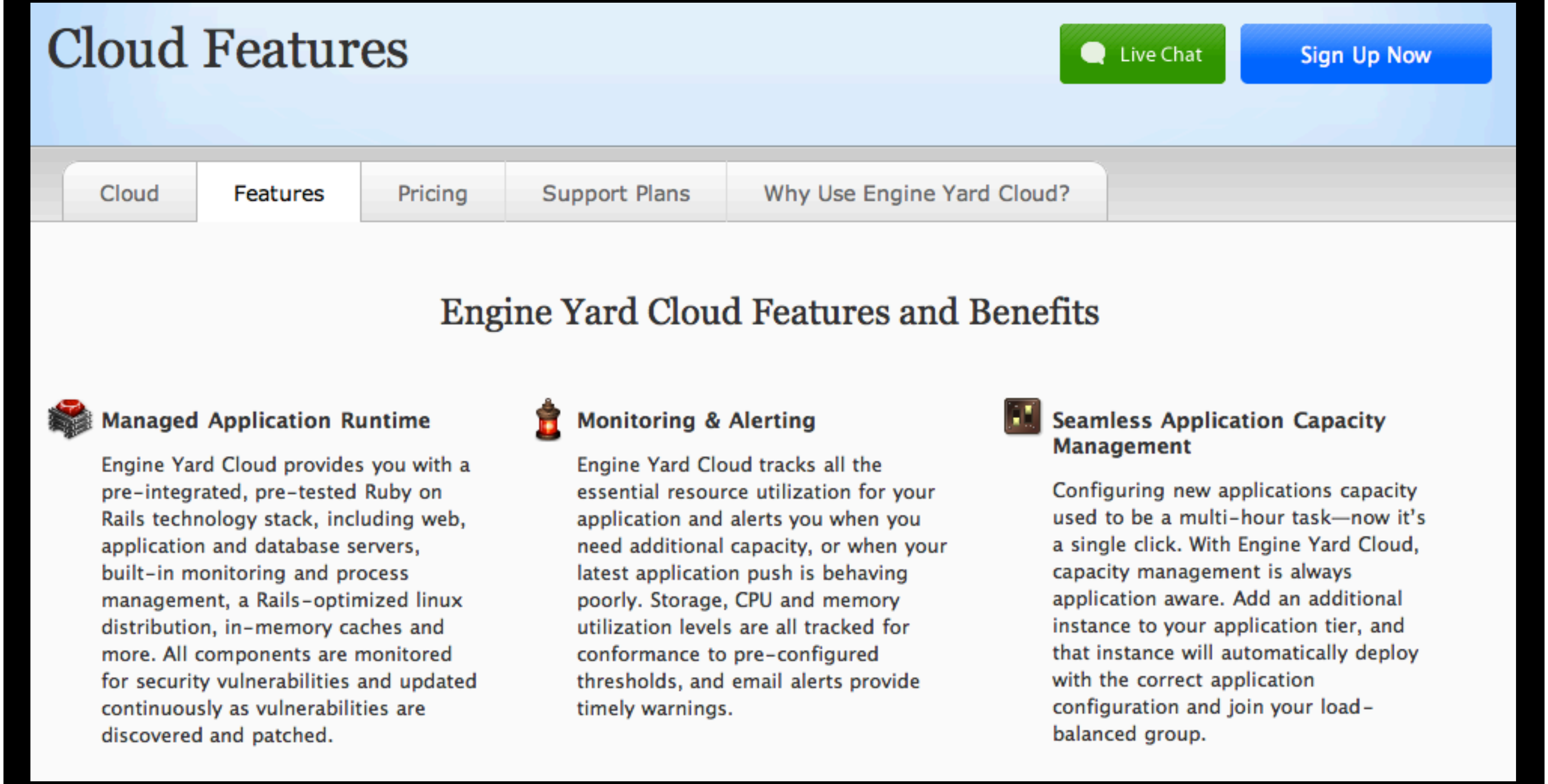
Discover Engine Yard Cloud

Watch a Live 15 Minute Demo of Engine Yard Cloud

Next demo in:

9:43:27

Register Now



Cloud Features

Engine Yard Cloud Features and Benefits

Managed Application Runtime

Engine Yard Cloud provides you with a pre-integrated, pre-tested Ruby on Rails technology stack, including web, application and database servers, built-in monitoring and process management, a Rails-optimized linux distribution, in-memory caches and more. All components are monitored for security vulnerabilities and updated continuously as vulnerabilities are discovered and patched.

Monitoring & Alerting

Engine Yard Cloud tracks all the essential resource utilization for your application and alerts you when you need additional capacity, or when your latest application push is behaving poorly. Storage, CPU and memory utilization levels are all tracked for conformance to pre-configured thresholds, and email alerts provide timely warnings.

Seamless Application Capacity Management

Configuring new applications capacity used to be a multi-hour task—now it's a single click. With Engine Yard Cloud, capacity management is always application aware. Add an additional instance to your application tier, and that instance will automatically deploy with the correct application configuration and join your load-balanced group.

JRuby

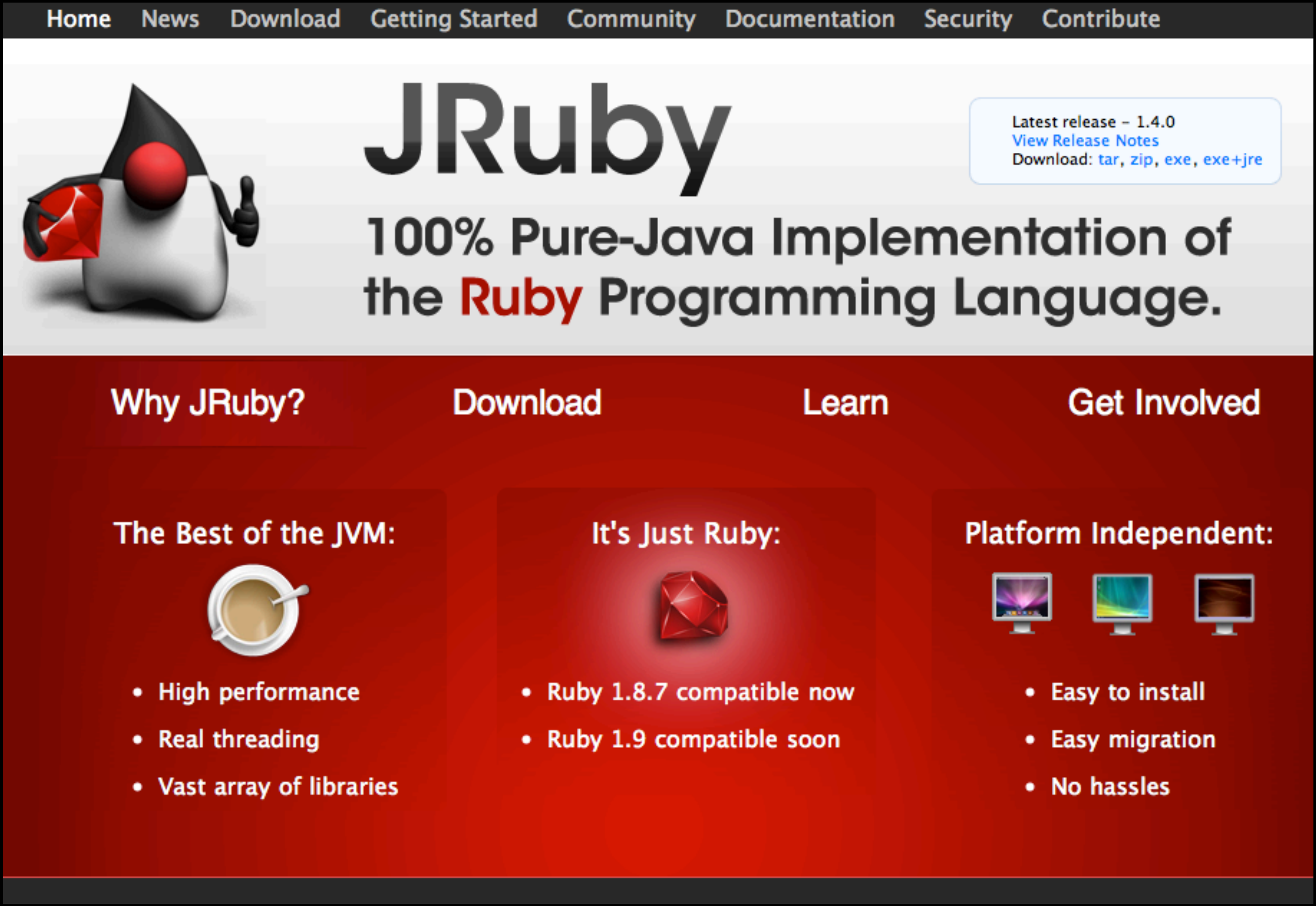
- Current version: 1.4.0
- Ruby for the JVM
  - A beautiful, simple, powerful language
- Rails for the Java platform
- “Just another JAR file”
- Interop with Java easily

The First Day

- Setup
- Learning Ruby and Rails
- Tools
- JRuby and Java
- Deployment

Getting JRuby

- Download from JRuby.org
    - Windows installer + tar/zip
- Packages for Linux, MacPorts
    - (Sometimes a little behind)
- Download an IDE (more on that later)
- OS X installer for 1.5.0



JRuby

Latest release - 1.4.0
View Release Notes
Download: tar, zip, exe, exe+jre

100% Pure-Java Implementation of the Ruby Programming Language.

The Best of the JVM:

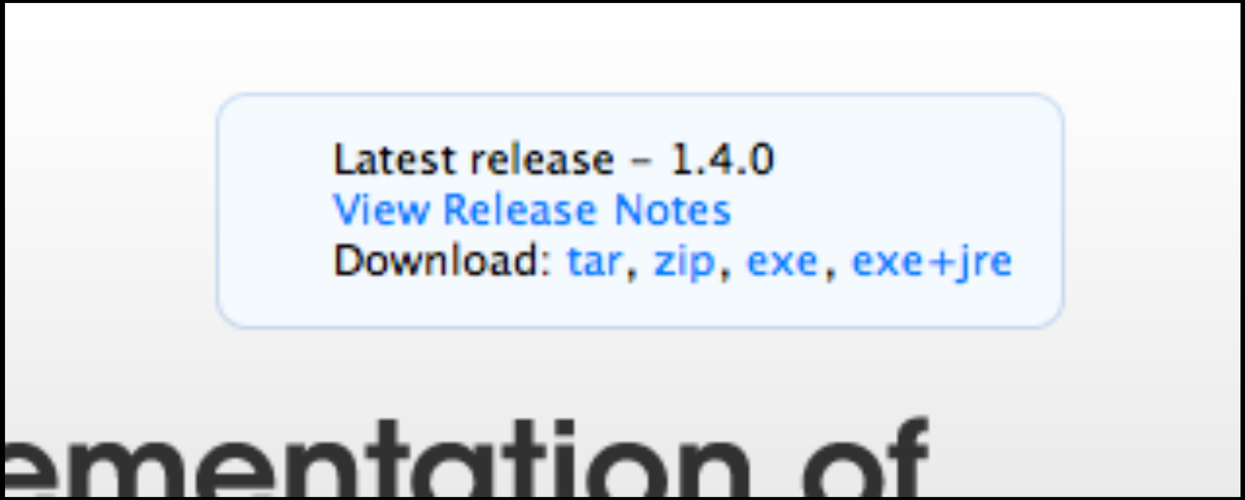
- High performance
- Real threading
- Vast array of libraries

It's Just Ruby:

- Ruby 1.8.7 compatible now
- Ruby 1.9 compatible soon

Platform Independent:

- Easy to install
- Easy migration
- No hassles



Latest release - 1.4.0
View Release Notes
Download: tar, zip, exe, exe+jre

ementation of

Installation

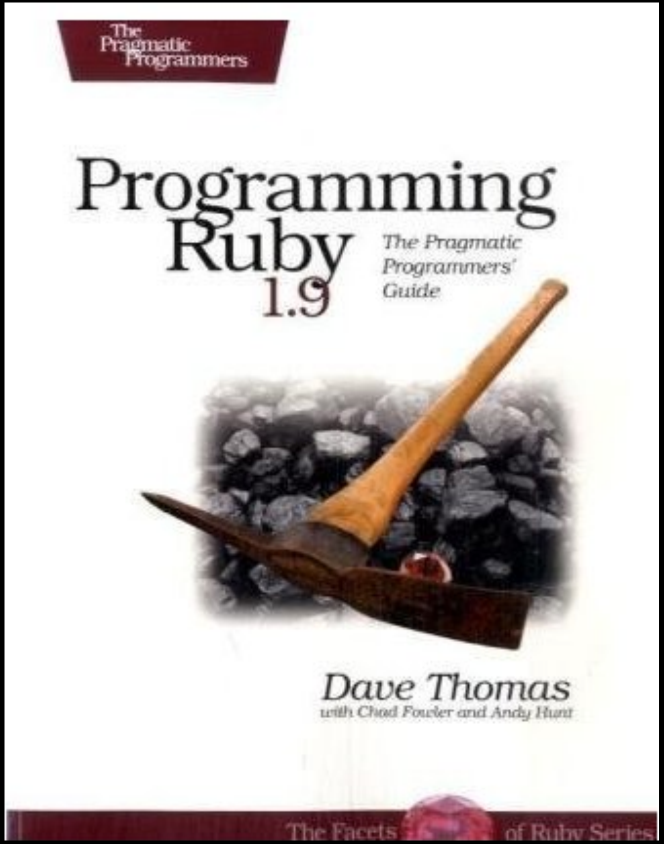
- tar/zip - Unpack, put in PATH
- exe/mpkg - Just run the installer
- Other packaging systems...

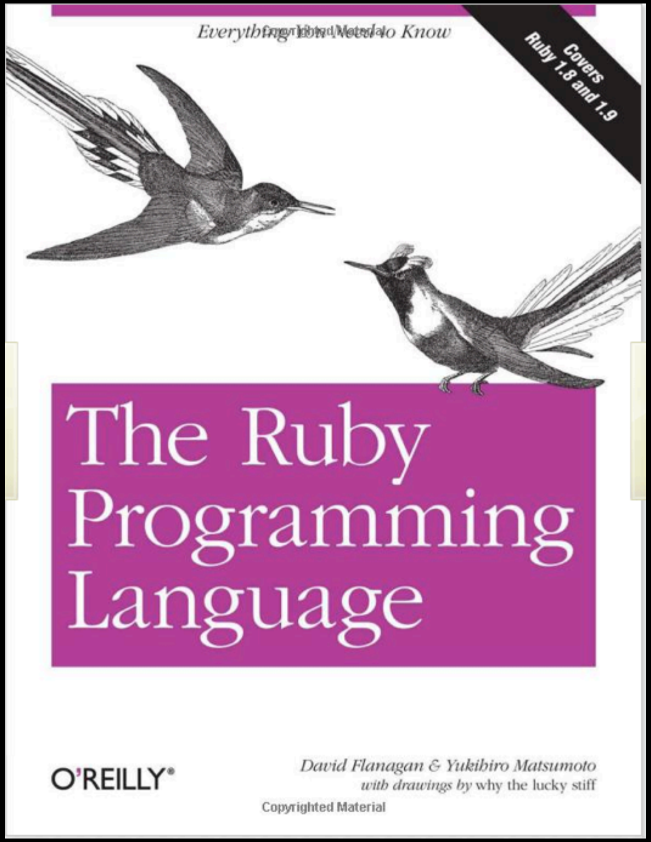
Try It Out!

Learning Ruby and Rails

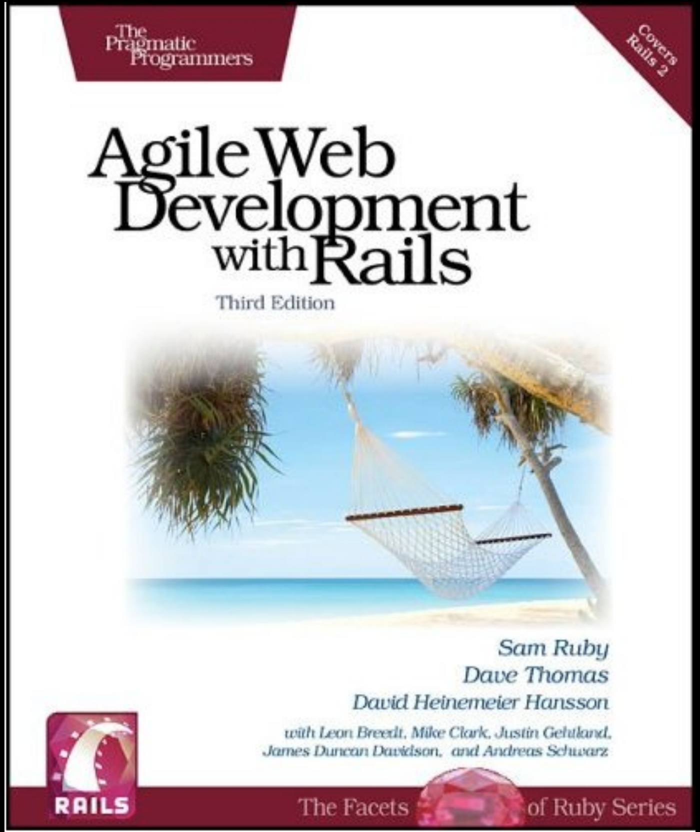
- Dozens of books out there
- User groups all over the place
- Conferences all over the place
- Active, friendly online community

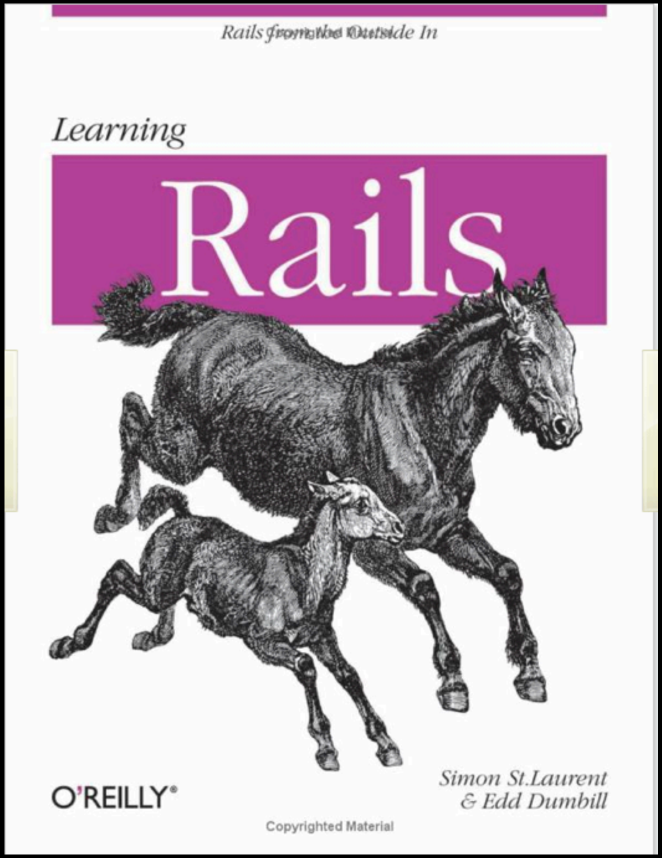
Learning Ruby


The Pragmatic Programmers
Programming Ruby 1.9
The Pragmatic Programmers' Guide
Dave Thomas
with Chad Fowler and Andy Hunt
The Facets of Ruby Series

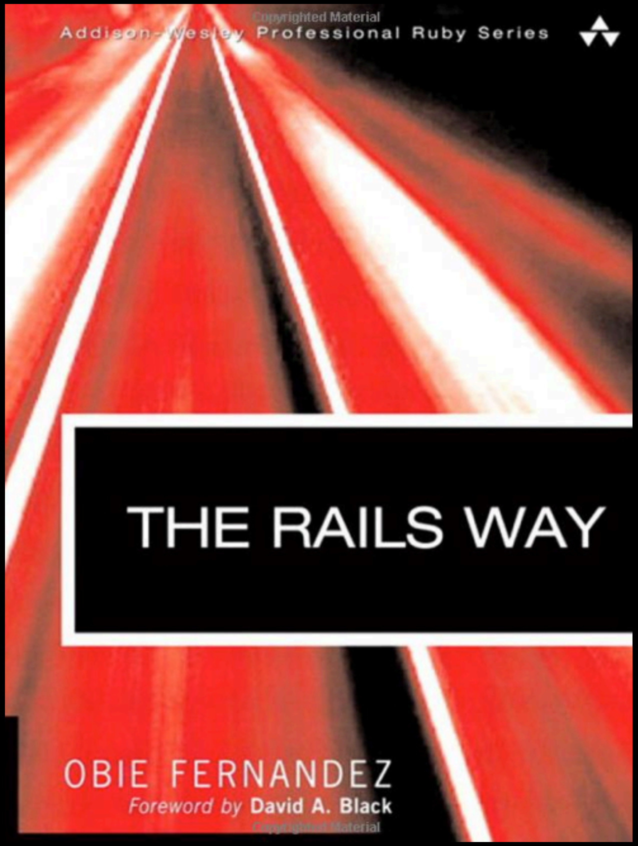

Everything You Need to Know
Covers Ruby 1.8 and 1.9
The Ruby Programming Language
O'REILLY®
David Flanagan & Yukihiro Matsumoto
with drawings by why the lucky stiff
Copyrighted Material

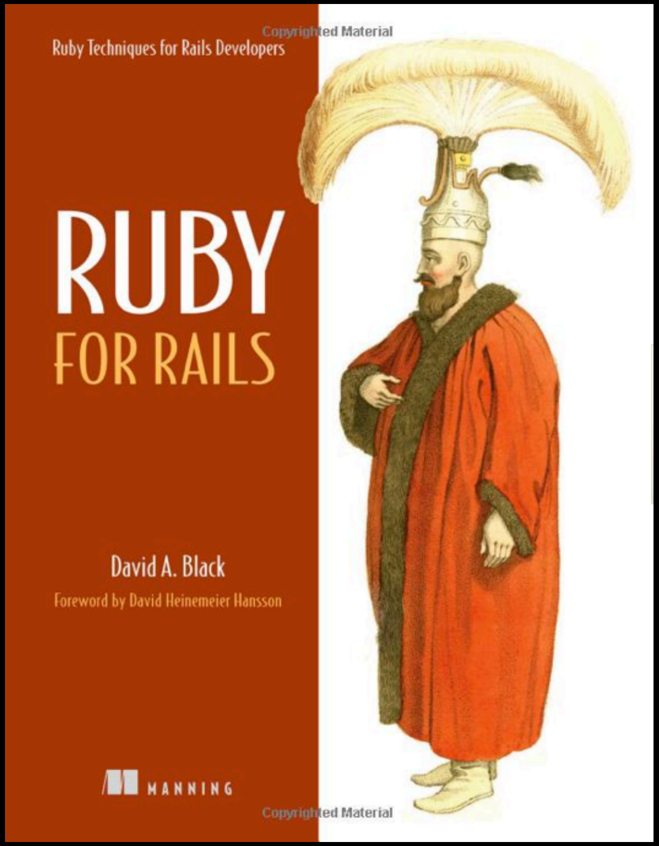
Learning Rails

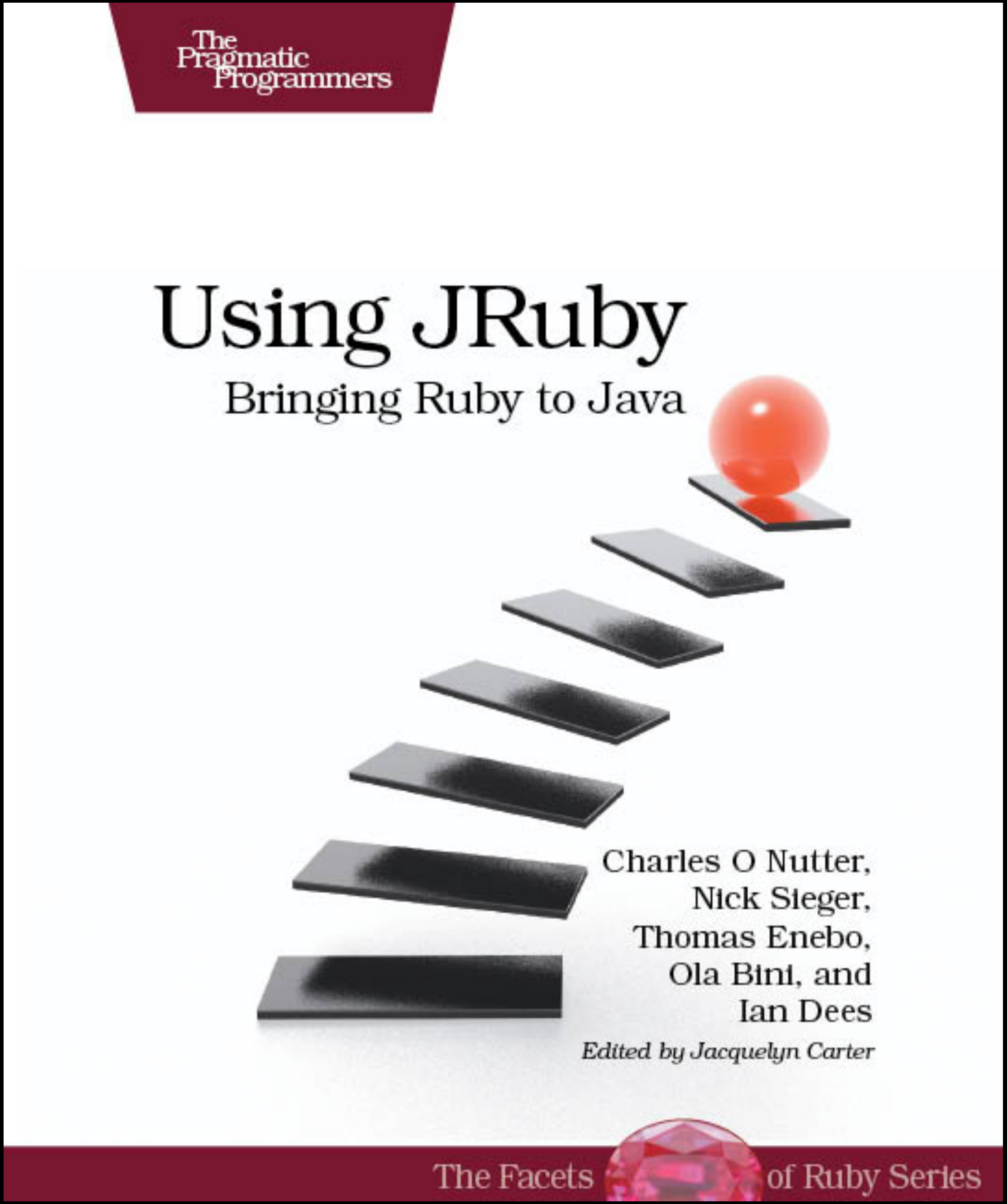

The Pragmatic Programmers
Covers Rails 2
Agile Web Development with Rails
Third Edition
Sam Ruby
Dave Thomas
David Heinemeier Hansson
with Leon Breedt, Mike Clark, Justin Gehtland, James Duncan Davidson, and Andreas Schwarz
RAILS
The Facets of Ruby Series



Getting Good







The Pragmatic Programmers

Using JRuby

Bringing Ruby to Java

Charles O Nutter,
Nick Sieger,
Thomas Enebo,
Ola Bini, and
Ian Dees

Edited by Jacquelyn Carter

The Facets of Ruby Series

Books Books Books!

NetBeans™ Ruby and Rails IDE with JRuby, Learning Rails, Rails for .NET Developers, Wicked Cool Ruby Scripts, JRuby Cookbook, Enterprise Recipes with Ruby and Rails, Developing Facebook Platform Applications with Rails, Foundation Rails 2, Enterprise Rails, Ruby On Rails Bible, Rails: Up and Running, Rails Pocket Reference, Ruby Phrasebook, Scripted GUI Testing with Ruby, Aptana RadRails, Advanced Rails Recipes, Deploying Rails Applications, The Art of Rails, Simply Rails 2, Practical REST on Rails 2 Projects, Ruby on Rails Web Mashup Projects, FXRuby: Create Lean and Mean GUIs with Ruby, RailsSpace, Ferret, Professional Ruby on Rails, Ruby: The Programming Language, Rails for PHP Developers, Pulling Strings with Puppet, Practical Reporting with Ruby and Rails, The Ruby Programming Language, Design Patterns in Ruby, Advanced Rails...

User Groups

- Swedish Ruby User Group
  - Meet-up Thursday night
  - http://rails.se/rails/show/HomePage
- Find one near you
- Start your own!

Conferences

- Nordic Ruby (http://nordicruby.org/)
  - 21-23 May, Gothenburg
- EuRuKo 2010 (http://euruko2010.org/)
  - 22-23 May, Krakow
- Scottish Ruby Conference (http://scottishrubyconference.com/)

Even more!

RubyConf, RailsConf, RailsConf EU, acts_as_conference, Euruko, Ruby Kaigi, Mountain West RubyConf, eRubyCon, Ruby Hoedown, Amsterdam Ruby en Rails, Scotland on Rails, RubyFools Copenhagen, RubyFools Oslo, Voices that Matter, South Carolina Ruby Conference, Lone Star RubyConf, RuPy, Gotham Ruby Conference, Silicon Valley Ruby Conference, RubyCamp, Conferencia Rails, Rails Summit Latin America, Ruby Manor, ...

Java into Ruby

```
public class MyClass {
  private String name;
  public MyClass(String name) {
    this.name = name
  }

  public void hello() {
    System.out.println(helloString(name));
  }

  public static String helloString(String str) {
    return “Hello, “ + str;
  }
}
```

Java into Ruby

```
public class MyClass
  private String name
  public MyClass(String name)
    this.name = name
  end

  public void hello()
    System.out.println(helloString(name))
  end

  public static String helloString(String str)
    return “Hello, “ + str
  end
end
```

Java into Ruby

```
public class MyClass

  public MyClass(String name)
    @name = name
  end

  public void hello()
    System.out.println(helloString(@name))
  end

  public static String helloString(String str)
    return “Hello, “ + str
  end
end
```

Java into Ruby

```
public class MyClass

  public MyClass(name)
    @name = name
  end

  public hello()
    System.out.println(helloString(@name))
  end

  public static helloString(str)
    return “Hello, “ + str
  end
end
```

Java into Ruby

```
class MyClass

  def MyClass(name)
    @name = name
  end

  def hello()
    System.out.println(helloString(@name))
  end

  def self.helloString(str)
    return “Hello, “ + str
  end
end
```

Java into Ruby

```
class MyClass

  def initialize(name)
    @name = name
  end

  def hello()
    puts(MyClass.helloString(@name))
  end

  def self.helloString(str)
    return “Hello, “ + str
  end
end
```

Java into Ruby

```
class MyClass

  def initialize(name)
    @name = name
  end

  def hello
    puts MyClass.helloString(@name)
  end

  def self.helloString(str)
    “Hello, #{str}”
  end
end
```

Tools

- RubyGems
- Rake
- Testing tools
- IDE support

RubyGems

- Ruby’s package manager
  - gem install <package>
- Remote fetch
- Dependency tracking
- Make your own!
  - gem push <mygem>

Example

```
~/projects/jruby ➔ gem install rails
Successfully installed activesupport-2.3.5
Successfully installed activerecord-2.3.5
Successfully installed actionpack-2.3.5
Successfully installed actionmailer-2.3.5
Successfully installed rails-2.3.5
5 gems installed
Installing ri documentation for activesupport-2.3.5...
Installing ri documentation for activerecord-2.3.5...
...
```

RubyGems + Maven

- gem install <maven artifact>
  - Fetches library + dependencies
  - Installs as a gem
- New for JRuby 1.5.0!

Rake

- Ruby’s build tool
- Clean, simple DSL
- Lots of standard and plugin tasks

Rakefile

```
require 'rake'
require 'ant'

task :init do
  ant.mkdir :dir => 'build'
end

task :compile => :init do
  ant.javac :destdir => 'build' do
    src {
      pathelement :location => 'src'
    }
  end
end

task :clean do
  ant.delete :dir => 'build'
end
```

Rake + Ant

- Use Ant tasks, targets from Rake
- Call Rake tasks from Ant
- Full two-way interop
- New for JRuby 1.5.0!

Testing Tools

- test/unit
- rspec

test/unit

```
require ‘test/unit’
require ‘java’

class MyUnitTest < Test::Unit::TestCase
  def setup
    @list = java.util.ArrayList.new
  end

  def test_new_list_is_empty
    assert @list.empty?
  end

  def test_list_with_one_element_size_is_one
    @list << ‘foo’
    assert_equal @list.size, 1
  end
end
```

test/unit Results

```
~/projects/jruby ➔ jruby test_list.rb
Loaded suite test_list
Started
..
Finished in 0.019 seconds.

2 tests, 2 assertions, 0 failures, 0 errors
```

rspec

```
require 'spec'
require 'java'

describe "An empty list" do
  before :each do
    @list = java.util.ArrayList.new
  end

  it "is empty" do
    @list.should be_empty
  end

  it "has size == 1 with one element" do
    @list << 'foo'
    @list.size.should == 1
  end
end
```

rspec Results

```
~/projects/jruby ➔ spec list_spec.rb
..

Finished in 0.025 seconds

2 examples, 0 failures
```

rspec Specdoc

```
~/projects/jruby ➔ spec -f s list_spec.rb

An empty list
- is empty
- has size == 1 with one element

Finished in 0.052 seconds

2 examples, 0 failures
```

Other Frameworks

- Expectations
- Dust
- test/spec
- Mocking libraries
- ...

IDE Support

- IDEA/IntelliJ
  - RubyMine for pure Ruby
  - IntelliJ for JRuby
- NetBeans
  - Ships with JRuby
- Aptana RadRails (Eclipse)

JRuby and Java

- Using Java libraries from Ruby
- Using Ruby from Java
- Deploying Rails to Java servers

Java from Ruby

- Java libraries "just work"
- Ruby or Java conventions
  - obj.setMyFoo() becomes obj.my_foo =
- Interactive Ruby
  - Great way to explore a library

Using Java libraries from Ruby

More Java from Ruby

- GUI frameworks
  - MonkeyBars, Glimmer, LimeLight, ...
- Android Scripting Extensions

Using Ruby from Java

- Scripting APIs
  - javax.script (Java 6)
  - RedBridge (JRuby embedding API)
- Compile to Java classes
  - Ruby classes usable from Java
  - New in 1.5.0!

Ruby from Java

Deploying Rails

- The Ruby Way: command-line server
  - `gem install glassfish`
  - GlassFish V3 embedded deploy
- The Java Way: WAR files
  - `gem install warbler`
  - `warble` (from app dir)

Deploying Rails

More Rails Deployment

- Engine Yard Cloud
  - GlassFish gem-based
- Google App Engine
  - Java webapp/WAR based

More Information

- www.jruby.org
  - Download, connect, contribute!
- wiki.jruby.org
  - Lots of walkthroughs and examples
- www.ruby-lang.org
  - Ruby docs, community links

Thank You!

- Charles Oliver Nutter
  - headius on Twitter
  - headius@headius.com
  - blog.headius.com