

Sun Microsystems, Ruby, and JRuby

Charles Oliver Nutter
Senior Staff Engineer
Sun Microsystems



Except where otherwise noted, the content of this presentation is licensed under

the Creative Commons Attribution-Share Alike 3.0 United States License (<http://creativecommons.org/licenses/by-sa/3.0/us/>).

Agenda

- Introduction
- Open source and Sun Microsystems
- Ruby in the world
- JRuby
 - > What is JRuby
 - > Why JRuby?
 - > JRuby use cases

Introduction

- Charles Oliver Nutter
- Java developer since 1996
- Engineer at Sun Microsystems since 2006
- JRuby developer since 2004
- Also working on JVM dynlang support
- Wide range of past experience
 - > C, C++, C#, Python, Delphi, Lisp, Scheme
 - > Java EE and ME, WS-*

Why Work at Sun?

- History of openness
- Get to work on my favorite project
- Side-by-side with many software experts
- Work from home
- Travel around the world to speak

Open Source at Sun Microsystems

- OpenSolaris
- OpenOffice
- OpenJDK
- DTrace
- GlassFish
- ZFS
- MySQL
- NetBeans IDE
- Jython
- JRuby

Ruby in the World

- Ruby has appeared out of nowhere
- Rapidly becoming standard for new web apps
- Amazing adoption rate
- Big companies taking interest
- Many concerns about performance, scaling
- Shocked at weak Unicode support (1.8)
- Many new Rubies: JRuby, XRuby, YARV/1.9, Rubinius, Ruby.NET, IronRuby, MacRuby, MagLev

Sun's Commitment to Ruby

- Sun wants to help Ruby
 - > Funding JRuby development
 - > Engineers submitting patches to Ruby
 - > Donated SVN server to Matz
 - > Building out full support for Ruby in Solaris
 - > Still improving NetBeans Ruby support
 - > Sponsored a Rubinius sprint last year

Why Ruby?

- Designed for programmers
- Fluid, malleable...make what you want
- Growing and excited community
- Momentum from new Rubies
- Nothing else is quite like Ruby

JRuby

- www.jruby.org
- Java implementation of Ruby language
 - > “It's just Ruby!”
- Started in 2002, open source, many contributors
 - > Tom Enebo, Ola Bini, Marcin Mielzynsky, Nick Sieger, Vladimir Sizikov, MenTaLguY
- Aiming for compatibility with current Ruby version
 - > Ruby 1.8.6 patchlevel 111 (114 was just released)

Why JRuby?

- World-class VM that runs everywhere
 - > Faster performance for Ruby
 - > Better memory, thread scaling
- Vast collection of Java libraries
 - > Cross-platform, battle-tested
 - > Java libraries for everything!
- Java platform growing beyond Java
 - > Java is NOT the answer for everything
 - > The platform is greater than the Java language

Where is JRuby today?

- Tooling for IDEs
 - > JRuby enables NetBeans, Eclipse IDEs
- Swing GUI development
 - > Makes Swing much nicer to use, easier to handle
- Graphics
 - > Ruby makes graphics programming fun
- JRuby on Rails
 - > Better deployment options, better performance

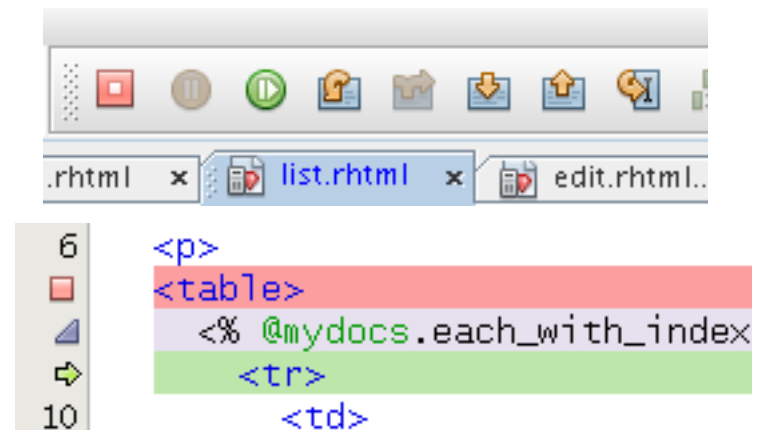
JRuby Enables Tooling

- JRuby's parser used by most Ruby IDEs
 - > NetBeans Ruby Support
 - > Eclipse RDT/RadRails/Aptana, DLTK, 3rd Rail
 - > IntelliJ (not parser, but other areas)
 - > Jedit
- NetBeans is the best right now (www.netbeans.org)

```
class Post < ActiveRecord::Migration
  def self.up
    create_table(name, options) ActiveRecord
    create_
  end
end
```

SchemaStatements.create_table(name, opt:

Creates a new table There are two ways to work with it: use the block form or the regular form, like this:



The screenshot shows a web browser window with three tabs: ".rhtml", "list.rhtml", and "edit.rhtml..". The "list.rhtml" tab is active, displaying a table with one row. The table has a header row with a red background and a data row with a green background. The data row contains a link labeled "@mydocs" and a link labeled "each_with_index".

```
6 <p>
7 <table>
8   <% @mydocs.each_with_index
9   <tr>
10    <td>
```

Swing GUI Programming

- Swing API is very large, complex
 - > Ruby magic simplifies most of the tricky bits
- Java is a very verbose language
 - > Ruby makes Swing actually fun
- No consistent cross-platform GUI library for Ruby
 - > Swing works everywhere Java does (everywhere)
- No fire-and-forget execution
 - > No dependencies: any script works on any JRuby install

Swing Options

- Cheri - cheri.rubyforge.org
 - > Builder-like DSL syntax
- Profligacy - ihate.rubyforge.org/profligacy
 - > Rubified layout expression language
 - > Trivial event binding without listeners
- **MonkeyBars - monkeybars.rubyforge.org**
 - > **Leverage GUI builders**
 - > **MVC structure**

DEMO

Swing in Ruby

Pretty Graphics!

- “Processing is an open source programming language and environment for people who want to program images, animation, and interactions.”
 - > Basically a cool Java library for 2D graphics
- Ruby-Processing wraps Processing with JRuby
 - > Cool, rubified 2D graphics environment for you
 - > Eye-candy demos for us
 - > Thanks to Jeremy Ashkenas for putting these together

DEMO
A Face for
Stephen Hawking

Web applications

- Classic Java web dev is too complicated
 - > Modern frameworks follow Rails' lead
- Over-flexible, over-configured
 - > Conventions trump repetition and configuration
- Rails deployment is still a pain
 - > You shouldn't need N processes!
- Rails performance should be better
 - > JRuby has potential to be much faster

DEMO

JRuby on Rails

Production JRuby on Rails

- CSI's Disease Surveillance System
 - > www.govtech.com/gt/323349
- Oracle's Mix – digg-like social customer site
 - > mix.oracle.com
- Sun's MediaCast – file distribution portal
 - > mediacast.sun.com
- ThoughtWorks' Mingle – collaborative project mgmt
 - > mingle.thoughtworks.com
- More on the way!

JRuby Future: Ruby 1.9

- JRuby will support both 1.8 and 1.9
 - > Flag to switch versions
 - > Maybe flags to switch individual features
- “Wait and see” approach
 - > Waiting for 1.9 adoption, demand
 - > 1.9 is still a work in progress
- 'jruby --1.9' turns on 1.9 features
 - > Only a few: Fiber, proc -> Proc.new, etc
 - > Eventually full 1.9 support

DEMO

JRuby 1.9 Support

JRuby Future: Performance

- JRuby competes on two fronts
 - > Against other Ruby impls
 - > Against other JVM languages
- JVM is a perfect host for languages
 - > Strong static-language backing for Java
 - > Strong dynamic-language backing from Smalltalk
- We can be fast now, faster in the future
 - > Every JVM release is 15-30% faster
 - > Want to run faster? Upgrade!

Thank you!

- JRuby - www.jruby.org
> wiki.jruby.org
- NetBeans - www.netbeans.org
> Try it out, send feedback!
- Ruby-processing (google it)
- My blog: headius.blogspot.com