

JRuby on Rails

Charles Oliver Nutter

JRuby Core Developer

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

- JRuby Basics
- Demo: Setting up
- Why JRuby: Deployment
- Deployment Demo
- Why JRuby: Libraries
- Library Demo
- Closing Notes and Q/A

Me

- Charles Oliver Nutter (charles.nutter@sun.com)
- Longtime Java developers (12+ yrs)
- Ruby developer (3 yrs or so)
- Engineer at Sun Microsystems for 1.5 yr
 - > Full-time JRuby developer
 - > Also working on general JVM language support
- Lots of other crap
 - > C, C++, C#, Perl, Python, Delphi, Lisp, Scheme
 - > EE, ME, JINI, WS-*, BPM, ERP, STFU

JRuby

- “It's just Ruby” for the JVM
- Compatible with 1.8.6
 - > No continuations, no C-based extensions
 - > Native threading, better performance
- JRuby 1.0.3 current stable-branch release
 - > Rapidly falling behind, 1.0.4 probably EOL
- JRuby 1.1RC1 latest leading-edge release
 - > Compiler, compatibility, all sorts of awesome
 - > RC2 within the next week

JRuby Users

- mix.oracle.com
 - > Idea sharing, networking, Q/A
 - > Production staff would not run MRI
- mediacast.sun.com
 - > Media/marketing distribution channel
 - > Eating the dogfood
- Mingle by Thoughtworks
 - > Collaborative project management tool
 - > MRI was too difficult to productize, “shrink-wrap”

JRuby Setup

- Download
- Unpack

JRuby Setup (long version)

- Java installation (if necessary)
- Download JRuby
 - > Includes JRuby, Ruby stdlib, rake and rspec
- Unpack JRuby
 - > Multiple copies on same system work just fine
- Add `<jruby dir>/bin` to PATH
- Install gems (`gem install` or `jruby -S gem install`)

So, Why Bother?

- Easier setup
- Better performance
- Easier deployment
- More libraries
- Wider database support
- Platform independence
 - > JRuby on Rails on AS/400 calling MS SQL? No prob!
- Less political resistance

So, Why Bother?

- Easier setup
- **Better performance**
- **Easier deployment**
- **More libraries**
- Wider database support
- Platform independence
 - > JRuby on Rails on AS/400 calling MS SQL? No prob!
- Less political resistance

Performance

- No, it's not all that important
 - > Until it is!
- JRuby 1.0 was about 2x slower than Ruby 1.8.6
- JRuby 1.1 Beta 1 was about 2x faster
- JRuby trunk is 5x faster, often faster than 1.9
 - > As a result, we've stopped working on perf for now
 - > ...but targeting Java performance next

DEMO Performance

Deployment: The Old Way

- Mongrel as server process
 - > Times number of apps
 - Times number of concurrent requests
 - With monitoring to keep alive
 - Hopefully no zombies or memory leaks
 - Better have a big slice
- Sure, it can work...but why?
 - > These problems have been solved before

DEMO

JRuby on Mongrel

Deployment: A Classic Revisited

- Java app server deployment
 - > N apps
 - > N concurrent requests
 - > N database connections
 - > 1 process, to limits of machine
- Designed to scale up and out
 - > ...but at a cost of some agility
- Political standard
 - > Meet them halfway!

DEMO

JRuby on GlassFish

Deployment: Back to Basics

- We want both worlds
 - > Agile: CLI support, dev time server, no build/deploy
 - > Simplicity: one-shot execution, both prod and dev
 - > Scaling: single process for all apps and requests
 - > Ruby friendly: “gem install” and run

DEMO

JRuby GlassFish Gem

Libraries

- JRuby can't support native extensions
 - > Designed around single-threaded execution
 - (i.e. one app, one request per process at a time)
 - > Stability, security problems
 - > Too-permissive Ruby extension API
- But who cares?
 - > If you want to do it, there's a Java library
 - > If not, we support native access through JNA
 - > Porting is also not hard, and appreciated

DEMO

ImageVoodoo!

What Does It All Mean?

- JRuby is raising the bar for Rails all the time
 - > It works today!
 - > Performance improving constantly
 - > Massive collection of high-perf libraries
- GlassFish is not your daddy's app server
 - > An enterprise server in a 3MB gem!
 - > Far easier deployment than other options
- Looking for users, use cases to support
 - > What do you need? Work with us!

Links and Q/A...Thank you!

- www.jruby.org
- glassfish.dev.java.net
 - > Gem is at <http://rubyforge.org/projects/glassfishgem/>
 - > Looking for bug reports, feature requests!
- www.netbeans.org
 - > By far the best IDE/editor for Ruby
- Stop by the Sun booth
 - > Try out NB and GlassFish
 - > Dude, it's all Free.

JRuby on Rails

Charles Oliver Nutter

JRuby Core Developer

Sun Microsystems