



Java, JRuby och RSpec

Ola Bini



Agenda

- Om mig
- Vad är Ruby och JRuby
- Vad är BDD
- RSpec
- Demonstrationer
- Ett större Javasystem
- Frågor och svar



Om mig

- Ola Bini
- Utvecklare, Karolinska Institutet
- JRuby Core Developer
- Javautvecklare sedan 1998
- Rubyfantast sedan 2004
- Stort intresse för programmeringsspråk



Vad är Ruby?

- Rent OO, dynamist, starkt typat, interpreterat programmeringsspråk
- Open Source (GPLv2)
- C-implementation de-facto standard (MRI)
- Startades 1993 av Yukihiro ('Matz') Matsumoto
- Kraftfullare än Perl, mer OO än Python
- 'Principle of least surprise'



Rubykod

```
• 1.upto(10) { |n| puts n }

• class Foo
  def initialize(val)
    @internal = val
  end
  def bar(arg1, *rest)
    rest.each do |v|
      yield @internal * v * v
    end
  end
end
Foo.new(13).bar(:aSymbol, 17, 42, 25, 113) { |o|
  puts "We got the value #{o}"
}
```



Vad är JRuby?

- 100% Java impl av Ruby...Ruby på JVM:en
- Rent interpreterat, operativsystemstrådar
- Föddes 2001, många utvecklare inblandade
- Mestadels Ruby 1.8-kompatibel
- Långsammare än MRI... För tillfället



Ruby anropar Java

- `require 'java'`

```
JFrame = javax.swing.JFrame
```

```
JButton = javax.swing.JButton
```

```
frm = JFrame.new("Ett fönster")
```

```
btn = JButton.new("En knapp")
```

```
frm.set_size(300, 300)
```

```
frm.content_pane.add(btn)
```

```
frm.show
```



Ruby anropar Java

- `AL = java.awt.event.ActionListener`

```
class MyListener < AL
  def actionPerformed(event)
    event.source.text = "Ny text"
  end
end
```

```
btn.add_action_listener(MyListener.new)
```

- `btn.add_action_listener(AL.impl { |ev, _|
 ev.source.text = "Nyare text" })`



Vad är BDD?

- Behavior Driven Development
- Ett sätt att strukturera test
- Bör användas som TDD
- Testar förväntat beteende
- Black box
- Kan också testa prestanda och säkerhet



RSpec

- BDD-ramverk för Ruby
- “Executable specifications of program behavior”
- Fungerar väl i JRuby
- Kan användas för att testa Javakod



Demonstration av RSpec-test



Demonstration av RSpec-test av Javakod



Ett större Javasytem

- Samla data från flera datakällor
- Vara bas för all identitetsinformation på KI
- Administration via Javainterface
- IBM Tivoli Directory Integrator är spindeln i nätet

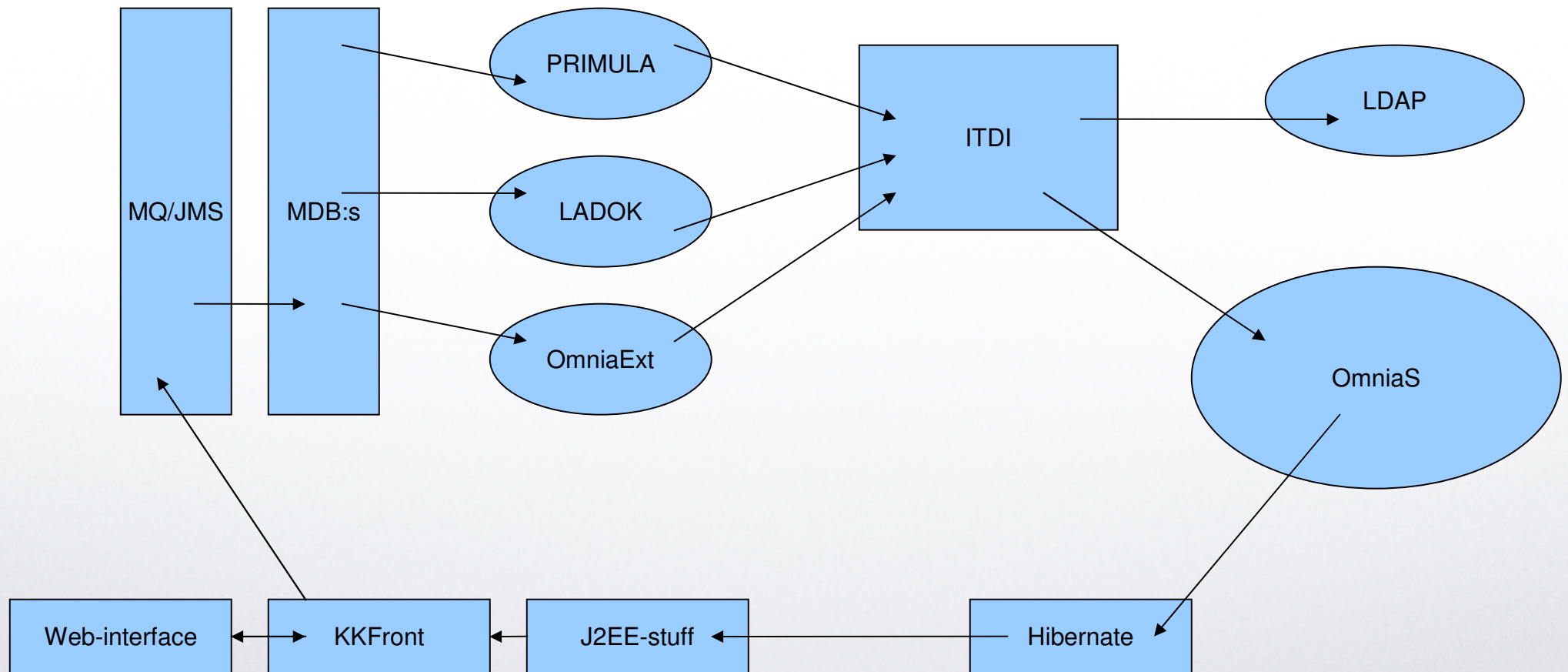


Acceptanstester - behov

- KIMKAT har massor av enhetstester
- Dock ej systemtester och integrationstester
- Tidsbrist!
- Beskrivningen av testfall skall vara så lika testimplementationerna som möjligt



En bild av KIMKAT





ITDI - MetaMerge

- Samlar information från datakällor
- Formaterar om data, implementerar affärsregler
- Skriver till databaser
- Använder Java 1.4.2 och JavaScript
- Vår svagaste punkt



KKFront

- Klientaccess till KIMKAT
- Hanterar behörighetskontroll
- Skrivning till mer än en datakälla är transparent
- Allt som behövs för ett Webgränssnitt till KIMKAT



Testning av Databaser + ITDI

- KITLib
- Standard Ruby
- Test::Unit
- ActiveRecord
- Mimer-adapter



Demo av KITLib



Testning av KKFront

- KITLib + extensions
- Kör i JRuby
- Ett extralib för att jobba mot KKFront



Demo av KKITLib



Sammanfattning

- BDD är ett sätt att skriva TDD i black box-manér
- Testning av innehåll i databaser med ActiveRecord är väldigt användbart
- Ruby kan användas för att skapa läsbara test-DSL:er
- JRuby är väldigt användbart för att testa Javakod
- Java+Ruby är en kombination som inte bör förringas



Information

- Email: ola.bini@ki.se
- Blog: ola-bini.blogspot.com
- Ruby: www.ruby-lang.org
- JRuby: www.jruby.org
- RSpec: rspec.rubyforge.org



Frågor?
Svar?
42!