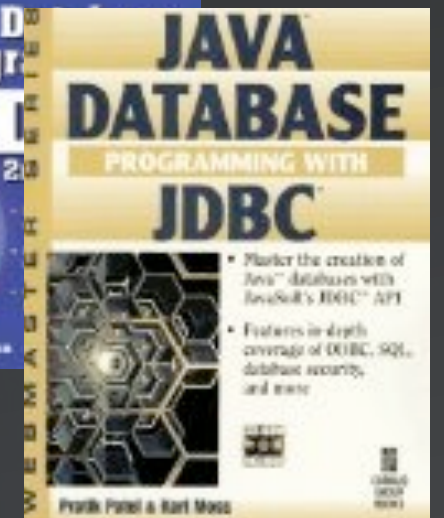




Groovy – Grails & Apache Camel

pratik@mypatelspace.com

About the presenter



About the presenter

Pratik Patel (pratik@mypatelspace.com)

- ▶ Corporate “architect” by day
- ▶ Code hacker by night
 - ▶ www.TripLingo.com



- ▶ Code repo:
 - ▶ <http://github.com/prpatel>
 - ▶ Twitter: prpatel

Publications



Topics

1 Rapid Prototyping with Groovy & Grails

2 Apache Camel

3 Grails - Camel plugin

4 Camel Groovy DSL

Dynamic Languages on the JVM

- 1 JVM – reads and executes bytecode
- 2 Bytecode can be generated from any language
- 3 Groovy, JRuby, Jython, Javascript (Rhino)
- 4 Varying levels of interop with Java code & libs

Scripting Langs in the Enterprise

- 1 Typically used for webapps
- 2 What about Perl, Python, Shell scripts?
- 3 We've always been polyglot programming!
- 4 Why not part of "real" development?

Choosing a dyn language

- 1 Groovy - Seamless Java integration, powerful
- 2 JRuby - based on Ruby
- 3 Jython - based on Python
- 4 JavaScript - based on... umm.. JavaScript

Groovy - best on the JVM

- 1 Groovy - Seamless Java integration, powerful
- 2 Easy Java-interop
- 3 Java code IS Groovy code
- 4 Tools we know and love

Grails - Rapid Web Dev

- 1 Grails - Rails-like framework built on Groovy
- 2 Hibernate and Spring under the hood
- 3 Convention over configuration
- 4 Plugins galore

Grails as a platform

- 1 Easy to integrate different technologies w/ plugins
- 2 ActiveMQ and Camel plugins!
- 3 Awesome for prototyping
- 4 Great for small to large projects

Apache Camel

1 The little Camel that Could



Grails

Code Examples

Camel Groovy

Camel Groovy

Camel

- ▶ A bit more terse
- ▶ Can do string interpolation

```
class ExampleRoute {  
    def configure = {  
        from('seda:input').to  
        ('stream:out')  
    }  
}
```

Closures in Route

Closures in Route

Camel

- ▶ Closure in a Filter

```
from("direct:start").filter {  
    e -> e.in.headers.foo ==  
    "bar"}.to("mock:result")
```


Creating a Route in Grails

Creating a Route in Grails

- ▶ Run this command
- ▶ `grails-app/routes`

```
grails create-route MyMessage
```

Grails Magic

Grails Magic

Runtime injection

- ▶ Grails routing plugin provides handy methods in svc and controllers

```
def myMessage =  
[name: "foo", data: "bar"]  
  
sendMessage("seda:myQueue",  
myMessage)
```

Adding Camel Components



Adding Camel Components

- ▶ in:
- ▶ BuildConfig.groovy

```
runtime  
( "org.apache.camel:camel-  
ftp:2.5.0" )
```



Live Demo
Hang on tight!

Grails as integration platform

- 1 XML processing is super easy
- 2 Runtime adjustment is easy
- 3 Grails plugins + Camel components = every tech
- 4 Easily extensible

Working with MQ's

- 1 Use the ActiveMQ Grails plugin
- 2 Embeds an instance of ActiveMQ
- 3 Use the bean builder or Spring applicationContext for others
- 4 Weblogic, Websphere, etc

Thanks for attending!

Samples:

<https://github.com/prpatel/>

Blog:

<http://www.mypatelspace.com/>

Links:

<http://camel.apache.org/>

Feedback Welcome
Fill out the Session Eval