Groovy And Grails

Harshad Oak

Rightrix Solutions

ApacheCon

ASIA 2006

Groovy Basics

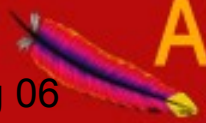
- Scripting / Agile / Dynamic ...Language
- Syntax very close to Java
- Meant for Java developers.
- A powerful high level language for the Java "platform"
- Groovy code Compiles to Java bytecode.
- You can get productive quickly

Groovy Basics

- Open source using a BSD / Apache style licence
- Backed by JSR 241
- groovy.codehaus.org

Groovy Installation

- Download binary
- Set GROOVY_HOME
- Add GROOVY_HOME/bin to your PATH
- Set JAVA_HOME

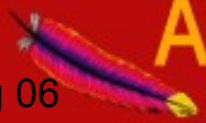


Groovy with Java

- Use groovyc just like javac to produce bytecode files
- Groovy integrates with the Bean Scripting Framework, which allows you to embed any scripting engine into your Java code

Closures

- Closures are reusable blocks of code
- One or more program statements enclosed in curly brackets
- Closures do not require a class or a method name

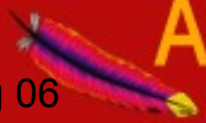


Closures

- Can be passed around like a variable
- The statements within a closure are not executed until the call() is made
- Return is using an explicit return statement or the value of the last executed statement

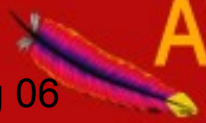
Dynamic Methods

- Adding new methods to an object at runtime
- Intercept non-existent method calls
- Used extensively in Grails
- Meta-Object Protocol



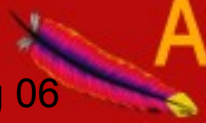
Groovy Features

- Language Constructs
- List & Maps support
- XML Capabilities
- Swing Capabilities



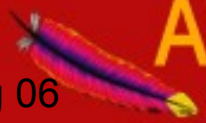
Groovlets and GSP

- Groovy alternatives for Servlets and JSP
- Will work on any J2EE container.
- Add JAR files
- Tweak web.xml



Groovlets and GSP

- Implicit Variables
  - request ServletRequest
  - response ServletResponse -
  - context ServletContext
  - application ServletContext
  - session getSession(false)
  - out response.getWriter()
  - sout response.getOutputStream()
  - html new MarkupBuilder(out)

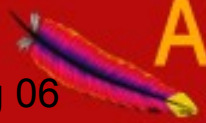


Grails

- Web Framework inspired by Ruby On Rails
- Latest version is 0.2.1
- Coding by convention paradigm
- Reduce the need for configuration files and other boilerplate code
- File names are important

Grails

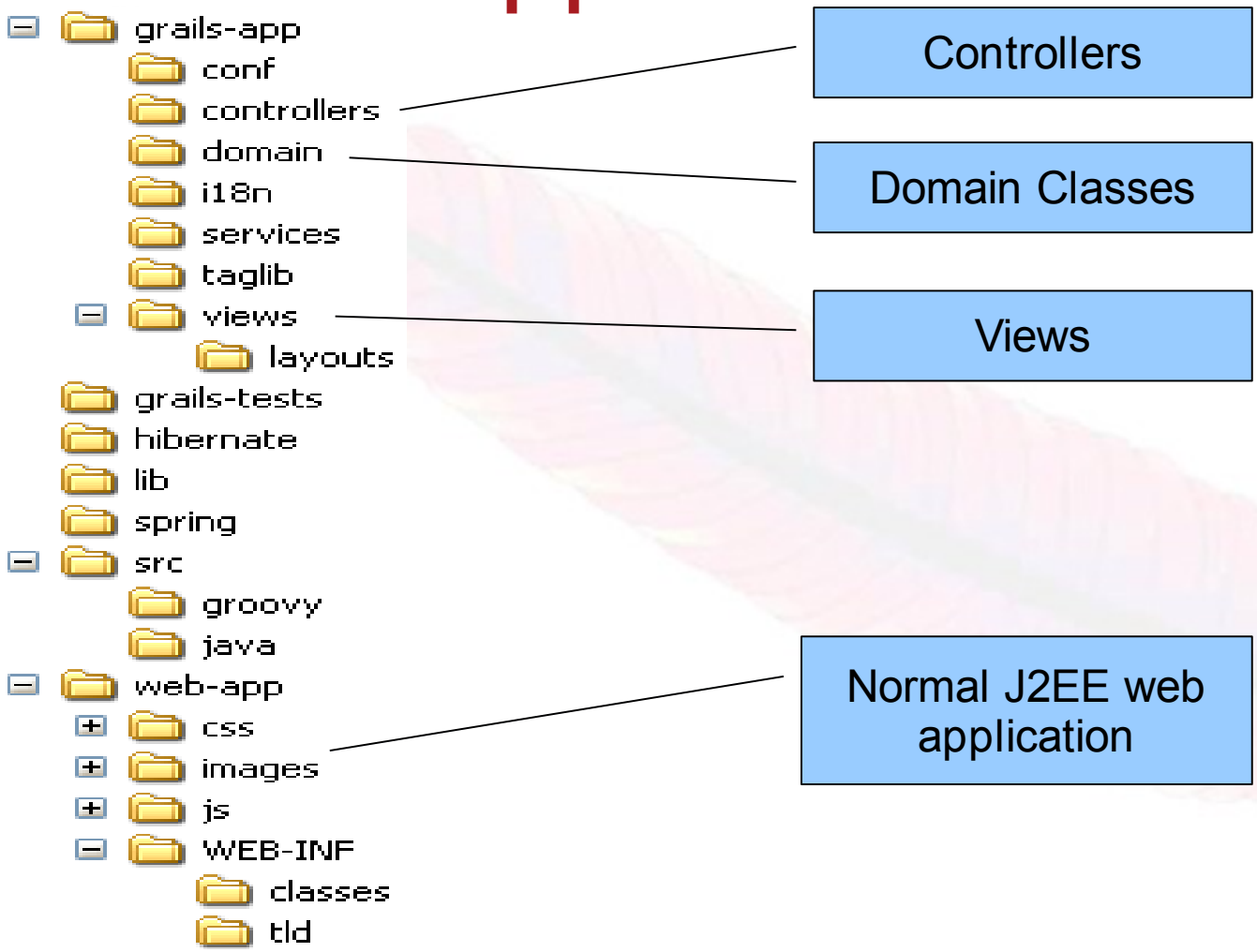
- Provides several commands to auto generate common requirements in a web app.
- Based on open source technologies like Spring, Hibernate and SiteMesh
- Grails hides underlying complexity
- Focus is on rapid development and simplicity



Grails Installation

- Extract contents
- Set GRAILS_HOME environment variable
- Add GRAILS_HOME\bin to the PATH variable
- Run the *grails* command from the command prompt to check usage

Grails Application Structure



GORM

- Domain Classes are objects that are mapped to the database
- Uses Hibernate 3 underneath
- Link domain classes using relationships
- Provide powerful dynamic methods for CRUD (Create/Read/Update/Delete) operations
- An HSQL database comes built-in with Grails

Grails Development

- Example -Develop a Clothes Manager web application
  - Use MySQL database
    - Setup DB
    - Setup ApplicationDataSource
  - Create Domain Classes

Grails Development - Domain Classes

- Create Domain Classes
  - Cabinet.groovy
  - Shirt.groovy
  - Trouser.groovy
- Define relationships

Grails Development -Controllers

- Controllers are responsible for handling the requests in the web application
- Use grails -generate-all to generate controllers for our 3 domain classes
- Once controller is done processing a request, it has to delegate to an appropriate view

Grails Development -Views

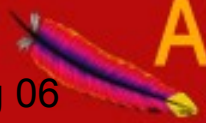
- Controllers delegate to view based on a convention mechanism
- So list closure in ShirtController will delegate to the view /grails-app/views/shirt/list.gsp or /grails-app/views/shirt/list.jsp.
- Note that you can use JSPs

Grails Development -Deploy

- Built in Resin server
- Directly run your application by using the grails run-app command
- Lets use Tomcat instead
  - WAR file creation using grails war
  - Deploy the WAR file just like any other J2EE WAR file

Grails Custom View and Controller

- Grails currently only auto generates views based on a domain class
- Manually create a new view
- Edit web.xml for mapping the new view
- Deploy and try out on Tomcat

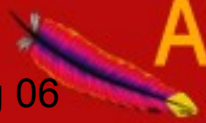


Grails Services

- Services are named in the format XXService.groovy
- Directory /grails-app/services
- Can have Business Logic
- You can call these services from within controllers

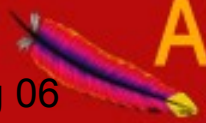
Grails Tags

- Range of tags for both GSPs and JSPs
- No configuration
- No updating of TLD descriptors
- Edit grails-app\taglib\ApplicationTagLib.groovy or create a new file XXTagLib.groovy



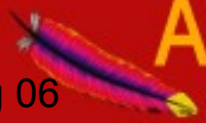
Grails - Ajax

- Can work with Prototype, Yahoo or Dojo
- Configure the library and then just use the grails tags
- Use render method to send text, markup, JSON or OpenRico responses



Grails

- Testing
- IDE Integration
- Validation



Thank You

- Further Reading
  - groovy.codehaus.org
  - grails.org
- Several books in the works
  - Groovy in Action (Manning)
  - Groovy Programming: An Introduction for Java Developers (Morgan Kaufmann)

Harshad Oak - harshad@rightrix.com