



GRAILS

Dynamic Tags

Mark-up Re-invented

Skills Matter

- The UK Open Source Training Company
 - Training on: Agile Development, Test Driven Development, Java, Eclipse, Spring, Hibernate, Ajax, Ruby on Rails, Struts Ti, Wicket, JavaServer Faces, Tapestry, Beehive, AOP, RIFE and more!
- Training straight from the Source
 - Experts write and teach our courses: Adrian Colyer, Chad Fowler, Howard M. Lewis Ship Craig Larman, Dave Crane, Kevlin Henney, Rob Harrop, Kito Mann, Rod Johnson and many more
- Partners with leading edge companies
 - BEA, IBM, Interface21, Sun, Tangosol, wso2

Your Speaker

- Graeme Rocher
 - Chief Technology Officer at Skills Matter
 - Project Lead of Grails
 - Member of the Groovy JSR-241 Expert Group
 - Author of “The Definitive Guide to Grails” (Apress)

Agenda

- What is Grails?
- The Problem with JSP tags
- Introduction to GSP
- Grails Dynamic Tag Libraries
- Summary

What is Grails?

- Grails: MVC web framework inspired by:
 - Convention over Configuration
 - Don't Repeat Yourself (DRY)
 - Ruby on Rails
- Built on solid foundations:
 - Spring + Spring MVC
 - Hibernate
 - SiteMesh
 - Quartz
- Why Groovy?
 - Expressive (like Ruby, Python etc.)
 - Meta-Programming
 - Closure Support
 - Java Integration



Why Grails?

- The Java platform is incredibly rich (more open source libraries than any other)
- Java servers and environments are **mature** and robust
- There are millions of Java developers world wide
- Static typing and IDEs **are** important
- But, there are fundamental problems with current Java web frameworks...

What Grails Does

- A full stack framework based on tried and **trusted Java** technologies. No re-inventing the wheel!
- Allowing a **blended approach** of static and dynamic typing. The best of both worlds.
- Giving Java developers the choice when it comes to the **right tool** for the job
- By embracing JEE standards with Groovy it allows you to maintain your existing investment in Java and get all the productivity benefits of a dynamic platform

Problem with JSP Tags

- **Too flexible**, the opposite of an anaemic API
- Vast range of possibilities
 - Nested, inter-related, flow control etc.
- **Over-complicated** & inaccessible
- JSP tags have an XML descriptor that is loaded by the container. Change it and you have to restart!
- How many have written a tag library for a project?

The Problem Remains

- How do you cleanly separate **logic** and **presentation**?
- Without tags or helpers this borders on impossible
- But, tags are complicated, so developers don't bother.
- Result == Mix of logic and presentation



GSP to the Rescue!



- Grails has its own view technology called Groovy Server Pages (GSP)
- Roughly similar to JSP except in Groovy!
- Behind GSP Grails supports **Dynamic Tag Libraries**
- Covers all the common cases: Simple, Logical & Iterative tags

```
<%-- GSP Examples --%>
<g:def var="world" value="\`world\' " />
Hello ${world}!
Hello <%=world%>
Hello <% out << world %>
<%-- Iteration --%>
<g:def var="items" value="\`[1,2,3]\` " />
<g:each in="\`${items}\`">
  <p>${it}</p>
</g:each>
```

A Few GSP Snippets

A Dynamic Tag Library

- **Class names** end with the “TagLib” suffix by convention
- **Property** names translate to tag names
- Each property is a closure that accepts the tag’s **attributes** and **body**

```
class DateTagLib {  
    Tag Name  
    ↙  
    def dateFormat =  
        { attrs, body ->  
        }  
}
```

Tag Attributes Tag Body

The diagram illustrates the structure of a TagLib class. The class name 'DateTagLib' is shown with an arrow pointing to the property name 'dateFormat'. The property 'dateFormat' is defined as a closure that takes two arguments: 'attrs' and 'body'. Arrows point from the labels 'Tag Attributes' and 'Tag Body' to these arguments in the closure definition.

```
// In the tag library
def dateFormat = { attrs, body ->
    out <<new SimpleDateFormat(attrs.format)
        .format(attrs.date)
    }

<!-- Then later in GSP -->
<g:dateFormat format="dd-MM-yyyy"
    date="{new Date()}" />
```

Dynamic Tags in Action



Demo

More on Dynamic Tags

- Many new features planned for dynamic tags
 - JSP tag support
 - Namespace support
- Community “Contribute a Tag” programme
 - <http://grails.org/Contribute+a+Tag>
 - Many user submissions already
 - Acegi Library, JavaBean Library etc.

Grails & The Future

- Grails 0.4 out shortly (Early January). Plug-in architecture available in 0.4 snapshots now!
- “The Definitive Guide to Grails” out here: <http://apress.com>
- New site <http://aboutgroovy.com> opening tomorrow!
- Conference in May 2007 - The Grails eXchange 2007 - <http://www.skillsmatter.com/grailsexchange>

Summary + QA

Thanks for coming!