

# Groovy & Grails in Depth



Simplifying Java EE with Grails

# Speaker's qualifications



- Graeme Rocher is Head of Grails Development at SpringSource
- Graeme Rocher is author of "The Definitive Guide to Grails" 1st and 2nd Edition (Apress)
- Graeme Rocher leads the development of the Grails framework and is a core committer on the Groovy language
- Graeme Rocher speaks frequently at industry conferences on the subjects of Groovy and Grails

# What is Groovy?

- Groovy is a dynamic language for the Java Virtual Machine (JVM)
- Takes inspiration from Small Talk, Python and Ruby
- Integrates with the Java language and platform at every level



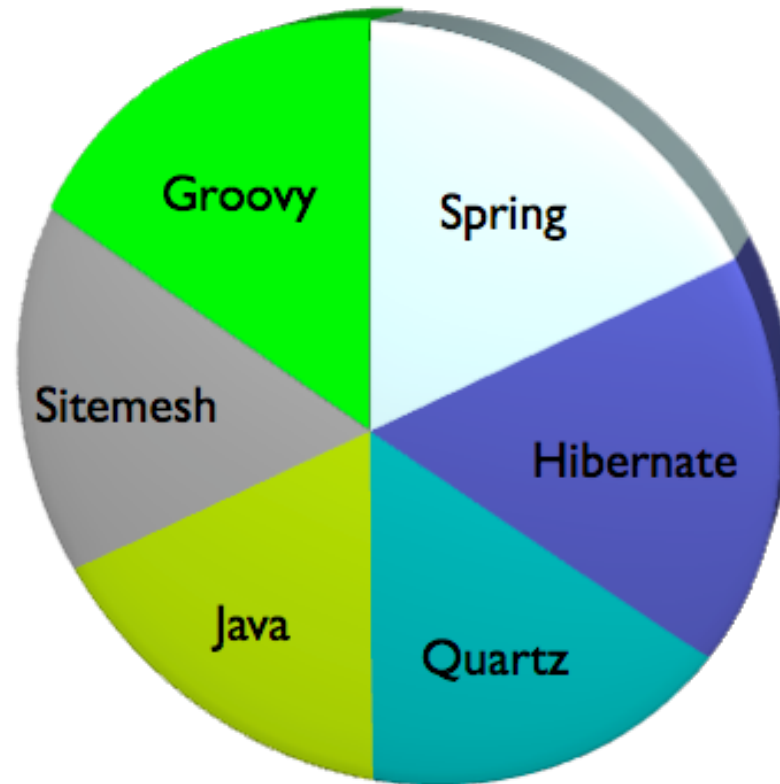
# What is Grails?

- A Web **platform** that implements the full stack from build system down to ORM layer
- Leverages existing technologies like Spring, Hibernate, Quartz etc. avoiding re-inventing the wheel
- Features and extensible plug-in system and an environment for runtime configuration built on Spring



# Solid Foundations

- Spring
- Hibernate
- Groovy
- Quartz
- SiteMesh
- Jetty
- Apache Ant
- Java







# Why Groovy & Grails?



- **All the power** of frameworks from other platforms like Ruby on Rails, Django etc.
- Built from the ground up on **Java technologies**.
  - Design for Java by Java Developers
- Leverage **existing tools and technologies**
  - Distributed caches, replicating technologies etc.
  - Profiling, monitoring and debugging tools

# Grails Headline Features



- GORM - ORM Layer built on Hibernate 
- Rich conversation support with Web Flow 
- Domain Specific Languages
  - Validation Rules
  - ORM Mapping
  - URL Mapping
- Vibrant Plug-in community
  - <http://grails.org/Plugins>
  - Over 100 user contributed plug-ins
  - Everything from web services to RIA with Flex 



- J2EE was too complex. Spring and Hibernate reduced that complexity and became very successful
- Grails represents the **next level of abstraction** for the Java EE platform.
- Built on Spring and Hibernate, but simplifying their usage with Groovy





# Getting Started



- Download from <http://grails.org/Download>
- Extract zip to disk
- Set `GRAILS_HOME` variable to location on disk
- Add `$GRAILS_HOME/bin` to your environment variables



- Download from <http://groovy.codehaus.org>
- Extract zip to disk
- Set `GROOVY_HOME` variable to location on disk
- Add `$GROOVY_HOME/bin` to your environment variables

# Grails – The Basics



## ➤ Creating and Running

- > `grails create-app gtunes`
- > `cd gtunes`
- > `grails run-app`

- Grails will load by default with an in-memory HSQLDB
- The default container is an embedded Jetty Container
- Grails is configured for automatic hot-reloading at runtime

## ➤ WAR'ing and Deploying

- > `grails run-war // or`
- > `grails war`

- Grails deploys as a standard WAR file deployable onto modern Java EE containers
- In production mode Grails is fully compiled byte code

# Grails Demo

Getting to know Grails

# Easy ORM with GORM

```
class Album {  
    String title  
    String artist  
    Date releaseDate  
    static hasMany = [songs:Song]  
}
```

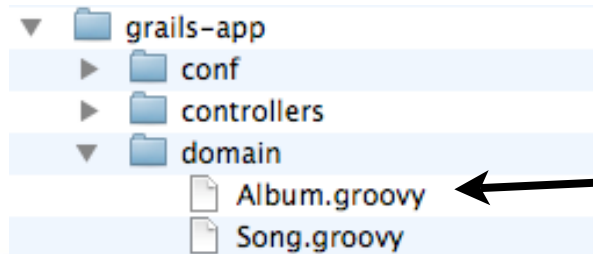
table - album

| id | title | artist | release_date |
|----|-------|--------|--------------|
|    |       |        |              |

```
class Song {  
    String title  
    Double duration  
}
```

table - song

| id | title | duration | album_id |
|----|-------|----------|----------|
|    |       |          |          |



GORM classes, also known as domain classes, go in the domain directory

# Dynamic Finders & Criteria



```
def albums = Album.list()
```

List all records

```
def recentAlbums =
```

```
    Album.findAllByReleaseDateGreaterThan(new Date() - 7)
```

```
def albumsStartingWithA =
```

```
    Album.findAllByTitleLike("A%")
```

Form method expressions

```
def albumsWithSongsAboutSummer =
```

```
    Album.withCriteria {
```

```
        songs {
```

```
            like("title", "%Summer%")
```

```
        }
```

```
    }
```

Use "like" queries

Construct criteria on the fly to query associations

- Dynamic finder and persistence methods
- Criteria with a Groovy builder
- Object-relational Mapping DSL
  - Caching
  - Legacy mapping
  - Locking strategy (optimistic/pessimistic)
- Built on Hibernate



## ➤ Comprehensive MVC framework built on Spring

- Command objects
- Data binding
- Zero configuration
- URL Mappings
- Groovy Server Pages (GSP)
- Tag libraries
- CRUD (Scaffolding)
- Internationalization (i18n)



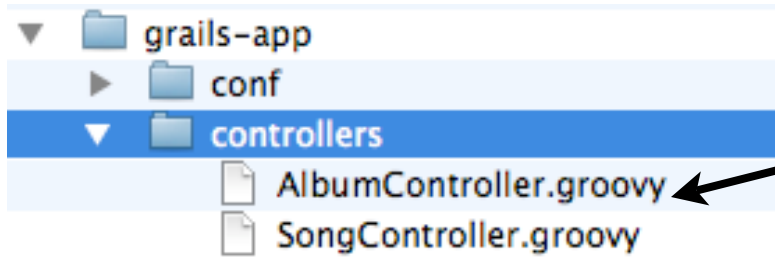


# Grails Controllers

```
class AlbumController {  
    def search = {  
        def albums= Album.findByTitleLike ("%${params.q}%")  
  
        [albumResults:albums]  
    }  
}
```

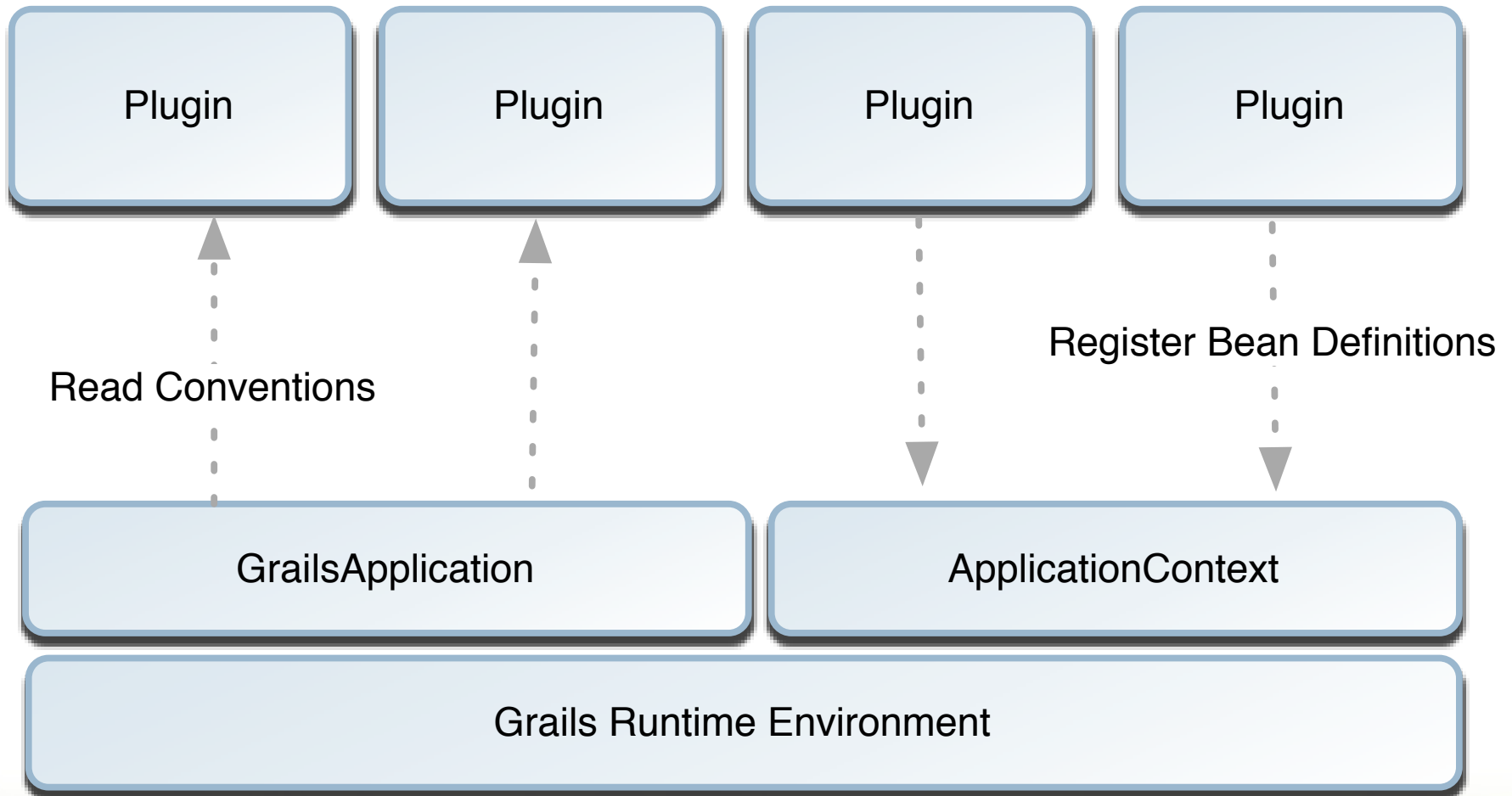
Actions are properties assigned a block or closure

Return a model as a map



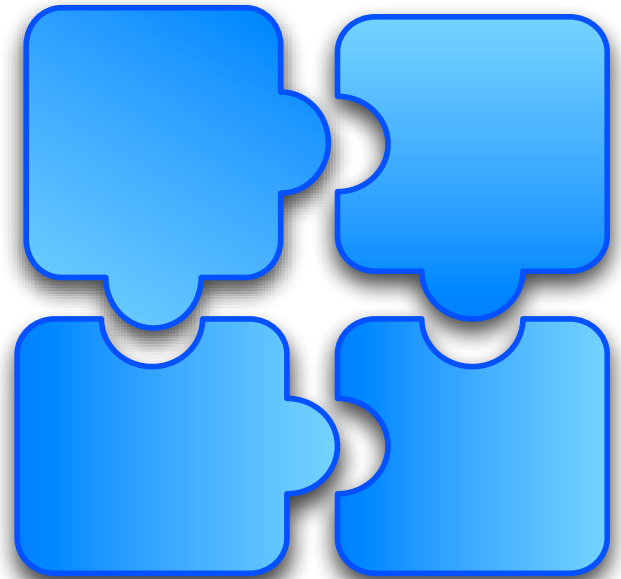
Controllers go in the "controllers" directory

# Plugins - Extend & Conquer



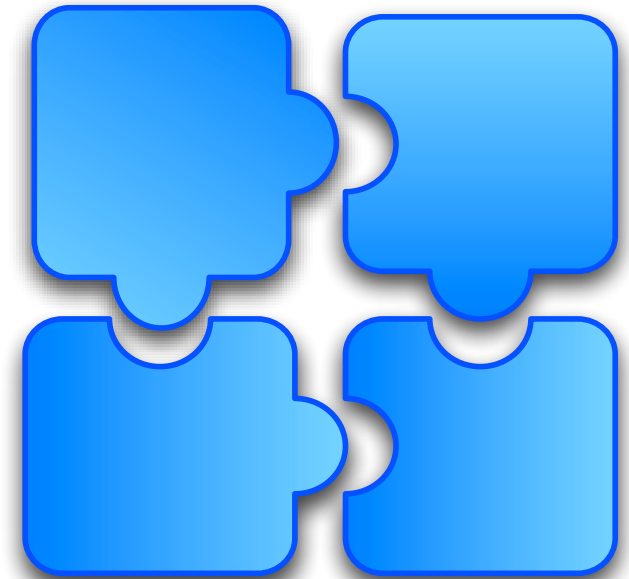
# What plugins enable...

---



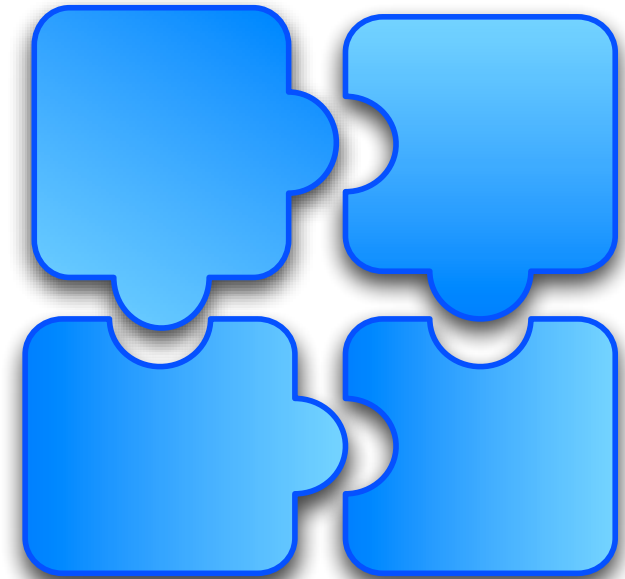
# What plugins enable...

- **Test Grails:** selenium, fitnessse, code coverage etc.



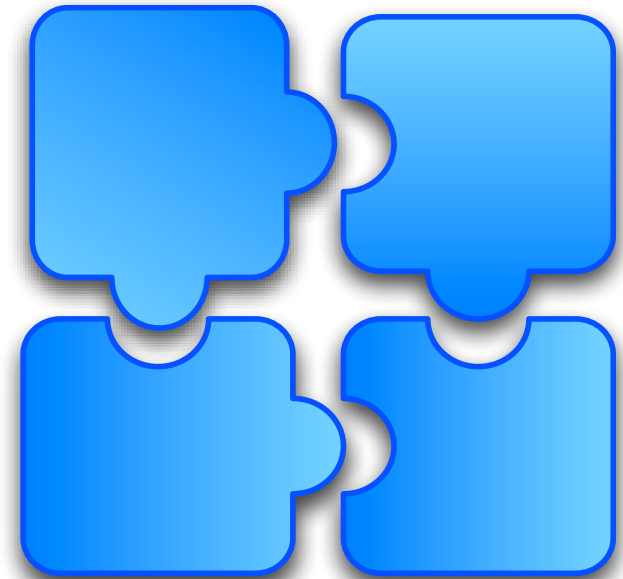
# What plugins enable...

- **Test Grails:** selenium, fitnessse, code coverage etc.
- **Rich Grails:** Flex, GWT, GrailsUI (YahooUI) etc.



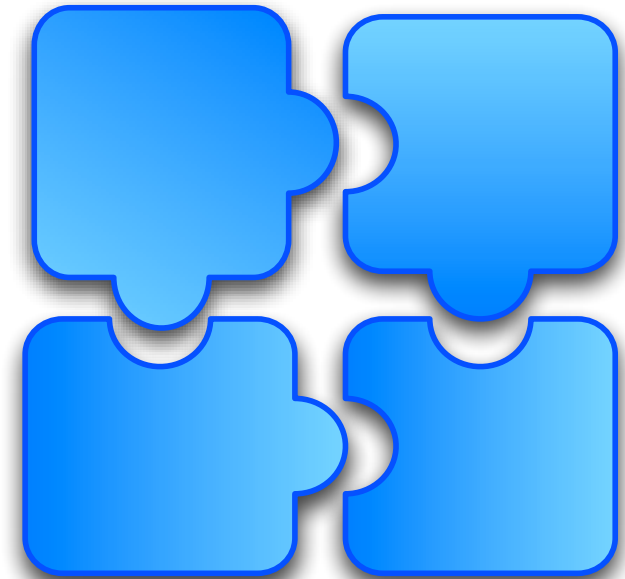
# What plugins enable...

- **Test Grails:** selenium, fitnessse, code coverage etc.
- **Rich Grails:** Flex, GWT, GrailsUI (YahooUI) etc.
- **Secure Grails:** Spring Security, JSecurity, OpenID etc.



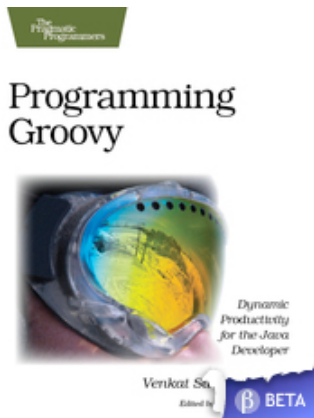
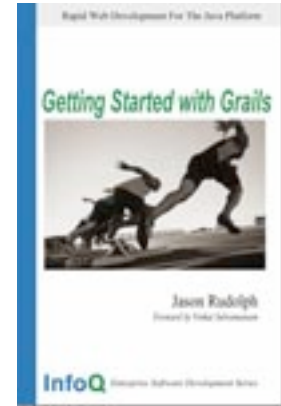
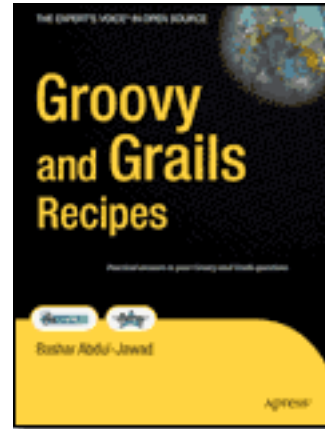
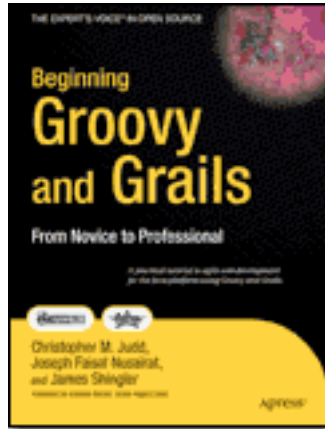
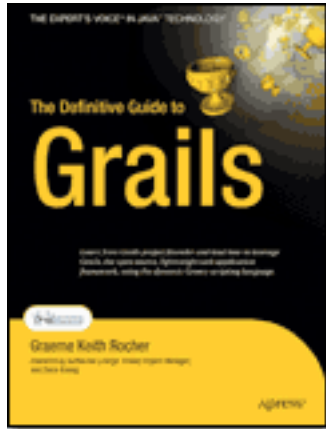
# What plugins enable...

- **Test Grails:** selenium, fitnessse, code coverage etc.
- **Rich Grails:** Flex, GWT, GrailsUI (YahooUI) etc.
- **Secure Grails:** Spring Security, JSecurity, OpenID etc.
- **Integrate Grails:** Search, Jasper Reports, JMS etc,





# Books



# For More Information

---



## ➤ Grails

- Web Site: <http://grails.org>
- User Guide: <http://grails.org/doc/1.1.x/>

## ➤ Groovy

- <http://groovy.codehaus.org>

## ➤ SpringSource

- <http://www.springsource.com>

- Grails is not just a web framework, but a web platform
- Grails is the next generation abstraction of the Java EE platform built on Spring and Hibernate
- Groovy and Grails integrate tightly with Java, the Java Virtual Machine at every level providing an easy migration path
- Grails == Simplifying Java EE on the web

# Q & A

Thank You!