The Grails® framework is the high-productivity toolkit enabling innovators worldwide to build robust, scalable enterprise web applications.

What is Grails?

The Grails® framework is an Open Source, high-productivity toolkit for creating large-scale web applications, including e-commerce websites, content management systems (CMS), RESTful web services, and more.

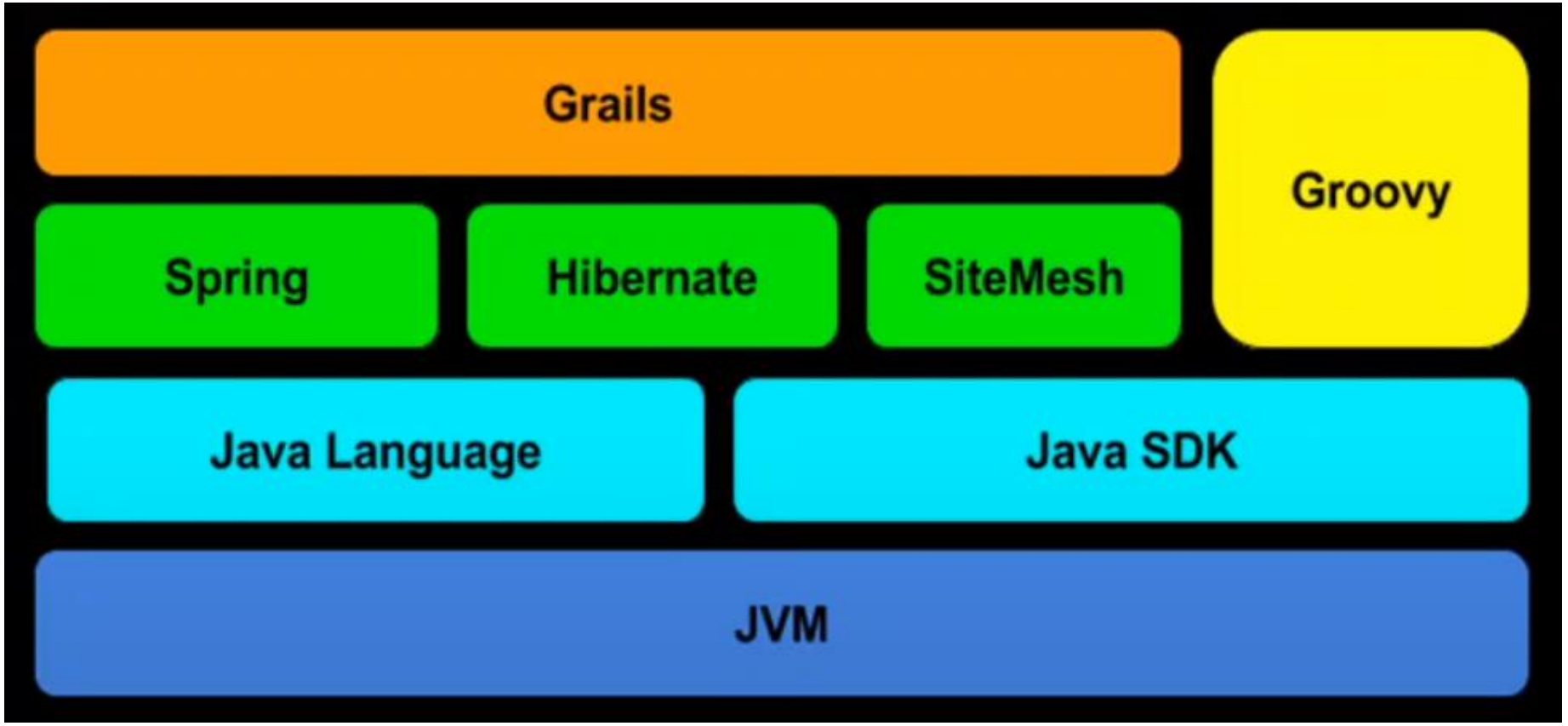
Grails allows to quickly create web applications; its scaffolding capabilities let you create a new project within few minutes. Grails is based on the convention over configuration idea which allows the application to auto-wire itself based on naming schemes

Grails uses JavaEE as the architectural basis and Spring for structuring the application via dependency injection.

Object relationship mapping (ORM) with GORM

Grails uses GORM (Grails Object Relational Mapping) for the persistence of the domain model. GORM is based on Hibernate. You can test with the HSQLDB and run in production against another database simply by changing the configuration file (DataSource.groovy).

Grails Eco System



Who’s using Grails Framework?

Benefits of Using Grails

Increased Developer Productivity

- Incorporating convention over configuration, the Grails framework eliminates much of the configuration effort typically required in building robust web applications.

Faster Time to Market

- The efficiency and precision of Grails code means faster development with fewer bugs and less code.

Agility

- Iterative development is a hallmark of the Grails framework because teams can carve out functionality in short sprints, with checkpoints and testing along the way.

Reusability

- The Grails plugin system enables code reuse across projects, allowing development teams to spend less time solving common web problems and more time implementing real business requirements.

Low Maintenance

- Foundational design assumptions eliminate the need for boilerplate code and account for an easier-to-navigate framework, thereby reducing maintenance complexity and project costs.

Cost Savings

- As an Open Source framework, Grails can be broadly adopted and used with no vendor lock-in, contracts, license management, or run-time licensing fees.

Flat Learning Curve

- Convention over configuration, sensible defaults, opinionated APIs, and the Groovy language combine to make the Grails framework easy to learn for Java developers.

Features of Grails

Groovy Lineage

- The Grails framework is based on Apache Groovy, a language for the Java platform designed to enhance developer productivity.

Spring Boot Foundation

- The Grails framework is built on top of Spring Boot and leverages Spring Boot's time-saving features, such as Spring-powered dependency injection.

Seamless Java Integration

- The Grails framework seamlessly and transparently integrates and interoperates with Java, the JVM, and existing Java EE containers.

Optimized Reloading Agent

- The Grails framework includes a development-time reloading agent that supports the dynamic reloading of code changes, thereby reducing the number of container restarts in the development environment.

Built-In Testing Framework

- The Grails framework's built-in testing framework helps maintain code quality throughout the development process and reduces defects in the final product.

Plugin Library

- Developers can build plugins that extend and enhance the Grails framework, or they can access existing plugins published by a vibrant plugin community.

Pragmatic Strategy

- The Grails framework applies the “Don't Repeat Yourself” (DRY) principle, thereby eliminating repetition and hidden bugs, and enabling faster and easier enhancements.

COMPARISON OF THE GRAILS FRAMEWORK TO OTHER TECHNOLOGIES

	Grails	Play	Struts/JSF	Rails	Node.JS	Spring MVC	Spring Boot
Targets JVM	X	X	X			X	X
Designed for Groovy	X						
Extensible via Plugins	X	X		X			
Built On Spring	X					X	X
Corporate Sponsored	X	X		X		X	X
Integrated ORM	X	X		X			
Full Stack	X			X			
Convention over Configuration	X	X		X			

Environment setup(4.0.6)

https://grails.org/download.html

Create Project from Web

http://start.grails.org/

Create Project using CLI

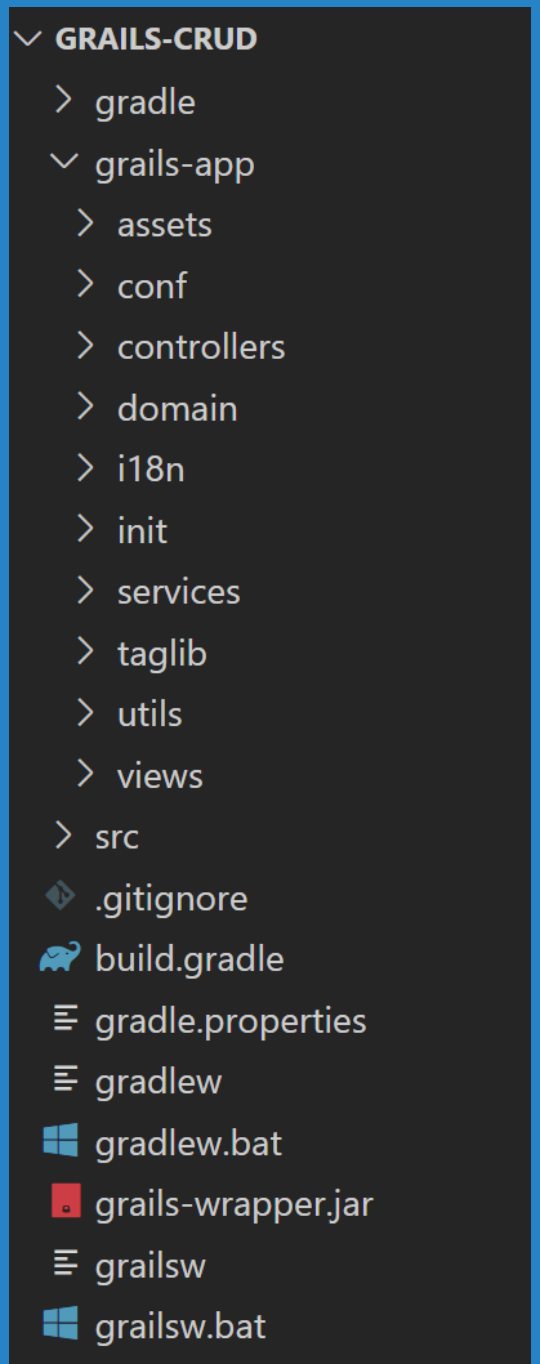
Create App:

- grails create-app com.kht.grails-crud
- cd grails-crud

open grails Interactive terminal

- type grails in cmd
- help for help commands

Groovy Directory Structure



Grails prefers convention over configuration, therefore the location of files defines their purpose. Let's see what we have in the grails-app directory:-

assets	– a place where we store static assets files like styles, JavaScript files or images
conf	– contains project configuration files:
application.yml	contains standard web app settings like data source, mime types, and other Grails or Spring related settings
resources.groovy	contains spring bean definitions
logback.groovy	contains logging configuration
controllers	– responsible for handling requests and generating responses or delegating them to the views. By convention, when a file name ends with *Controller, the framework creates a default URL mapping for each action defined in the controller class
domain	– contains the business model of the Grails application. Each class living here will be mapped to database tables by GORM
i18n	– used for internationalization support
init	– an entry point of the application
services	– the business logic of the application will live here. By convention, Grails will create a Spring singleton bean for each service
taglib	– the place for custom tag libraries
views	– contains views and templates

Resources

Resources:

https://objectcomputing.com/products/grails/resources

https://slack.grails.org/

https://grails.org/ecosystem.html

https://gsp.grails.org/latest/guide/tags.html