

The Grails Framework in 5 minutes



the search is over.



David Trattnig, 2010

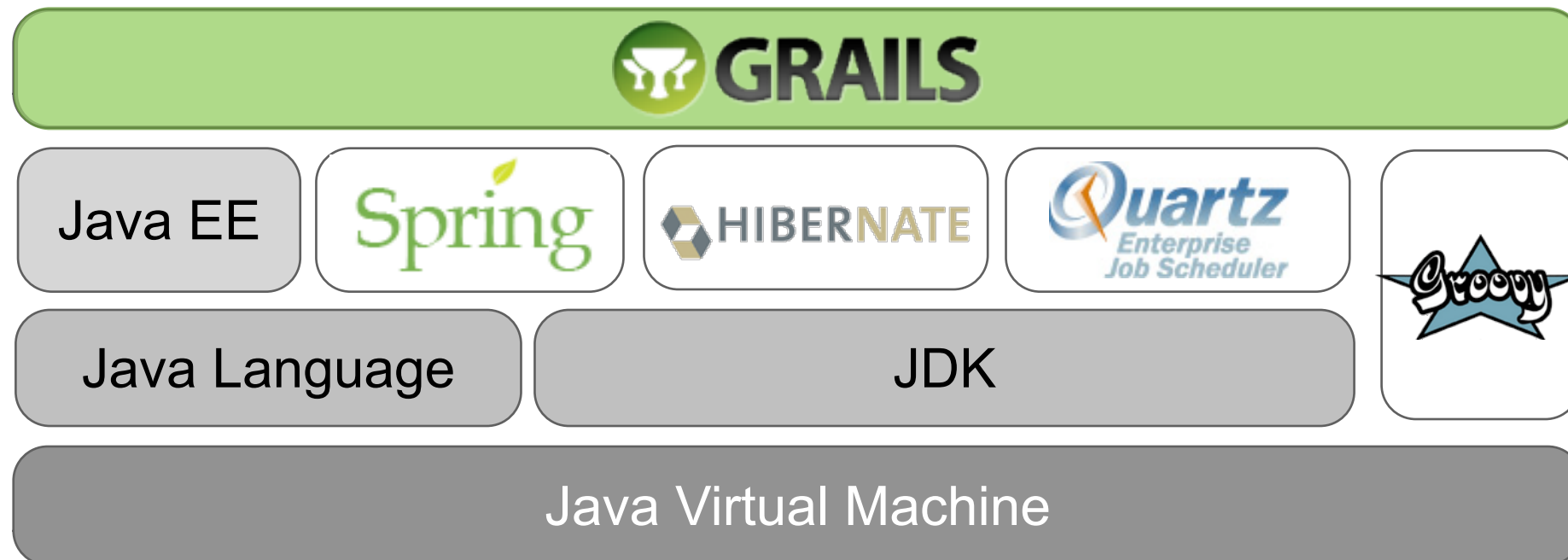
Full-Stack Framework

- Presentation
- Business Logic
- Persistence
- Build & Test
- Documentation



photo credits: <http://www.flickr.com/photos/slipstreamjc/748716731/>

An evolution.



- **Leverages existing resources**
(works with „JVM friendly“ hardware, software and humans)
- **Build upon industry-proven technologies**
(established, well tested and scalable Java frameworks & libraries)

Convention over Configuration (CoC)

- No more painful XML settings
- 80 / 20 rule
- Don't repeat yourself (DRY)

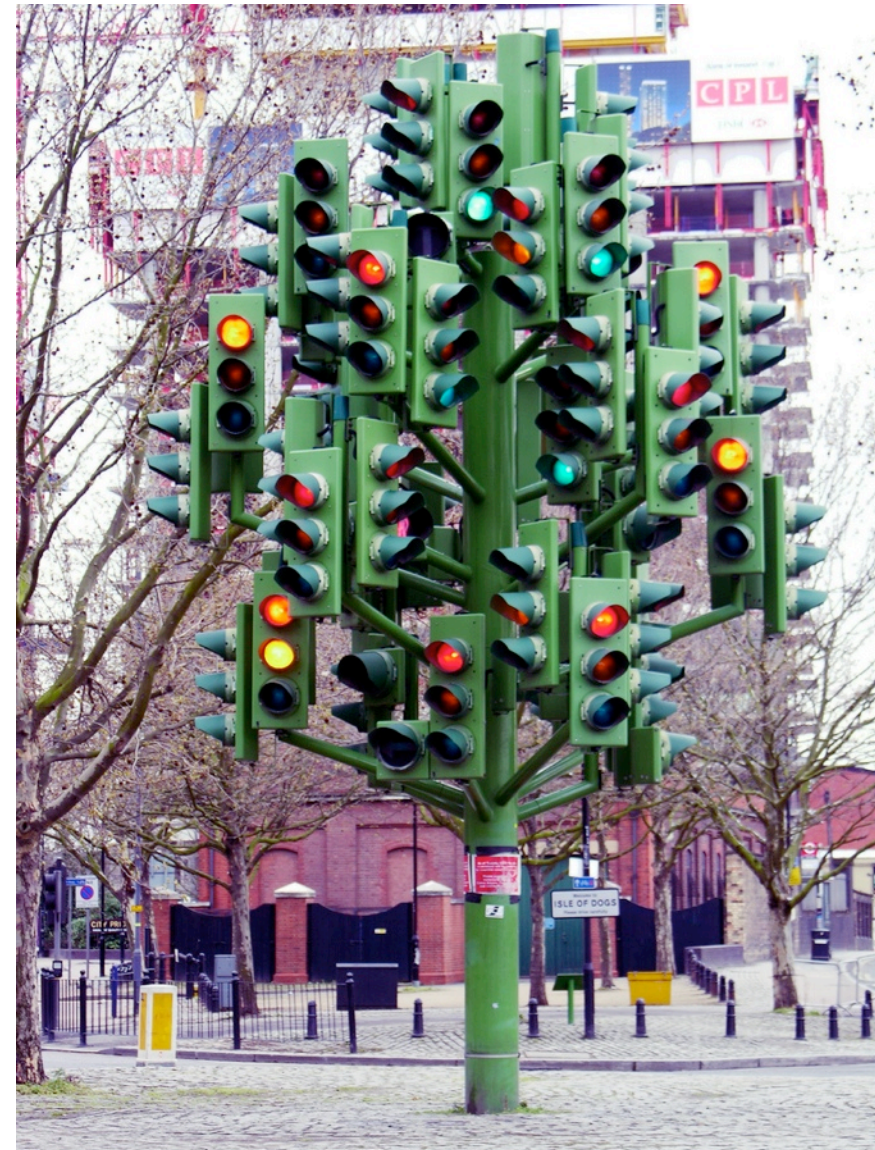


photo credits: <http://www.flickr.com/photos/squirmelia/2451382877>

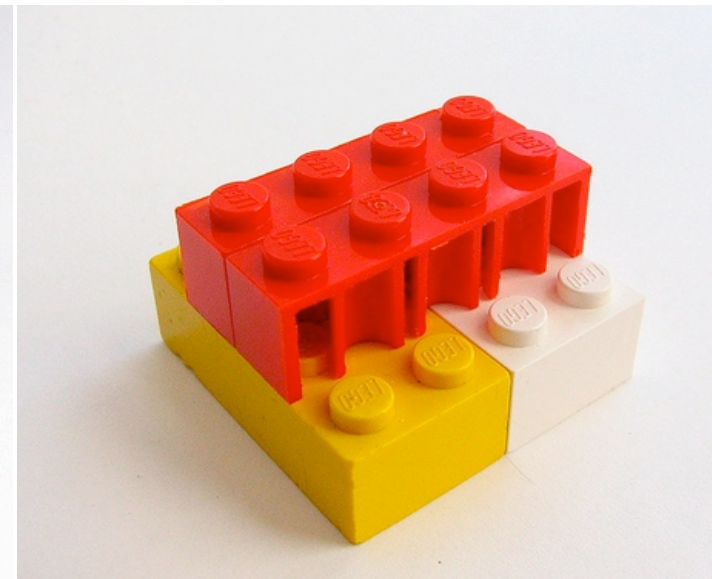
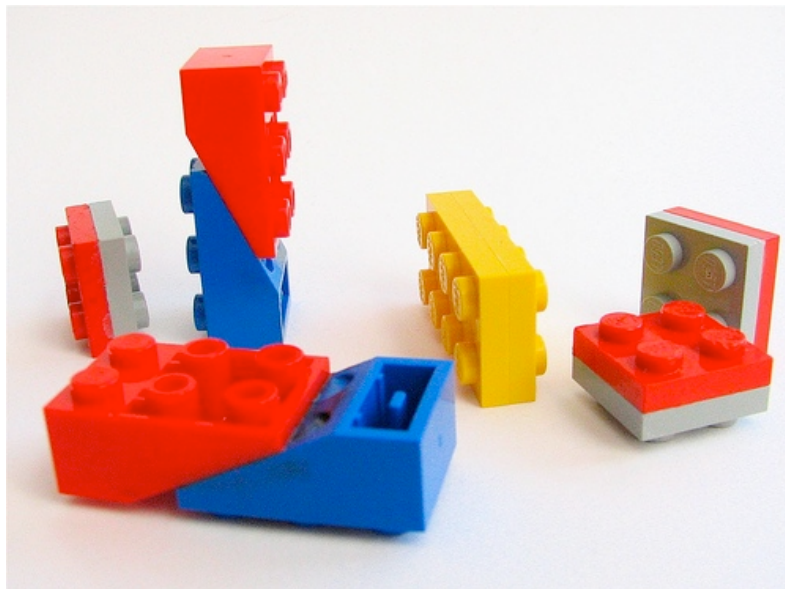
Domain Driven Design (DDD)

- Focus on the domain, rather than technology
- Rapid prototyping
- Adaptable architecture



Grails Plugin System

- 350+ Plugins available
- Modularized app architecture
- Reuse established Java libraries



Groovy, the language

- Less and more expressive code
- Mix Java & Groovy
- Dynamic language for changing requirements
- Compiles to Java bytecode
- „Relaxed“ learning-curve for Java programmers

Possible Pitfalls

- difficult debugging - nature of dynamic languages
(but enforces proper test-cases)
- multiple sources for possible defects
(stack of many but well-tested Java frameworks)
- „Fear of learning a *new* language“
(Split the project in a Grails-Part and pure Java-Part for „traditional“ developers)

Success Stories...



... are happy with  **GRAILS**