LET'S CREATE A MODERN WEB APP

SPRING

WHO AM I?

- Software Architect at Metrosoft
- piotr.laskawiec@javablog.eu
- @piotrlaskawiec
- facebook.com/javablog/
- Occasional speaker
- Trainer wannabe...

AGENDA

- Workshop details
- Spring - introduction, facts and myths
- Understand the magic - DI/IoC framework in 30 minutes
- Quick start - Spring Boot
- Managing the data - Spring Data
- Modern web app - Spring MVC
- Security above all - Spring Security
- Future - Spring 5, Kotlin

RULES

CONTEXT

MODERN WEB APP…

- Data-driven
- Self-contained
- Extensible
- Fast
- Maintainable
- Scalable
- „Reactive“

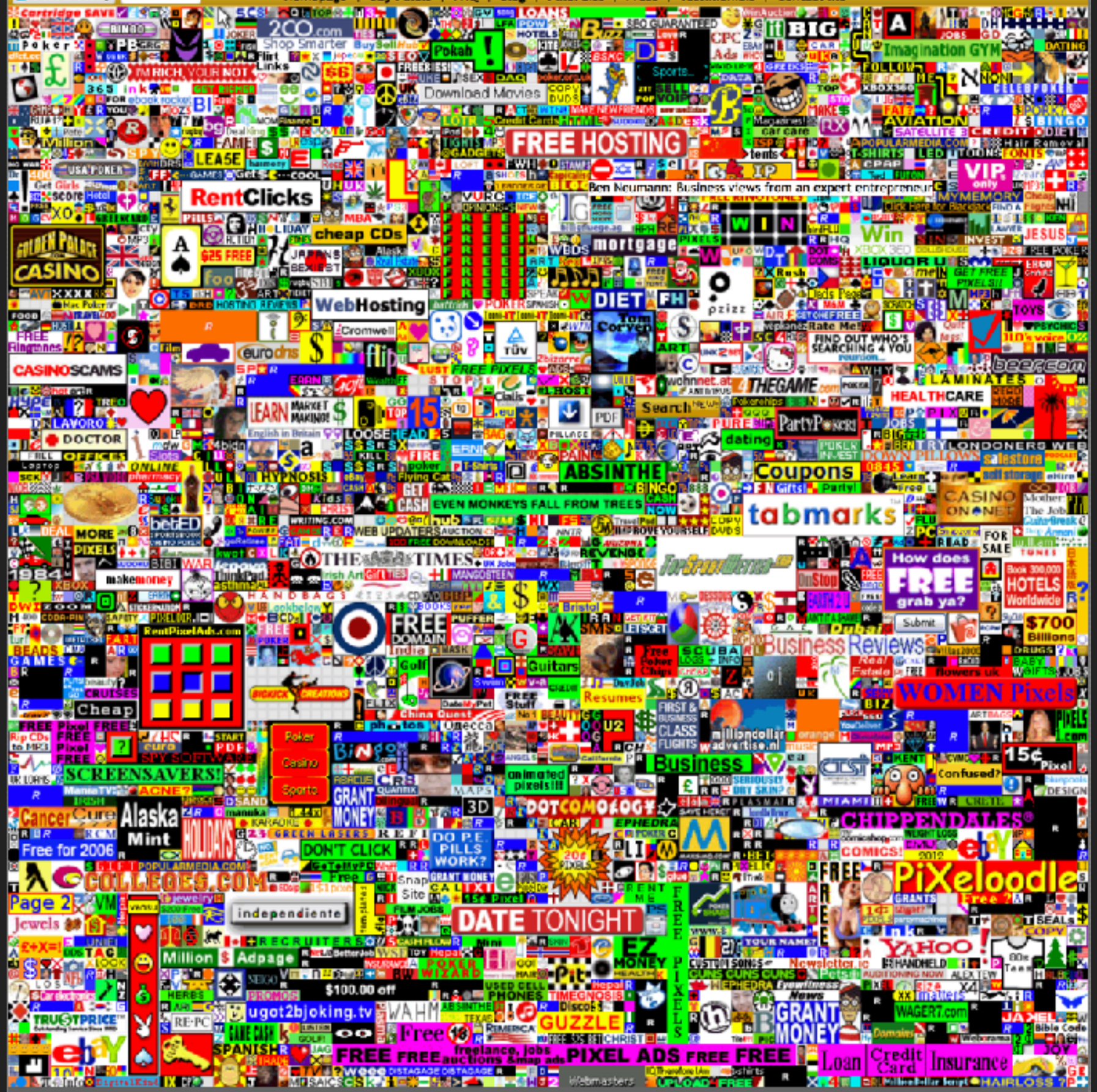
The Million Dollar Homepage™

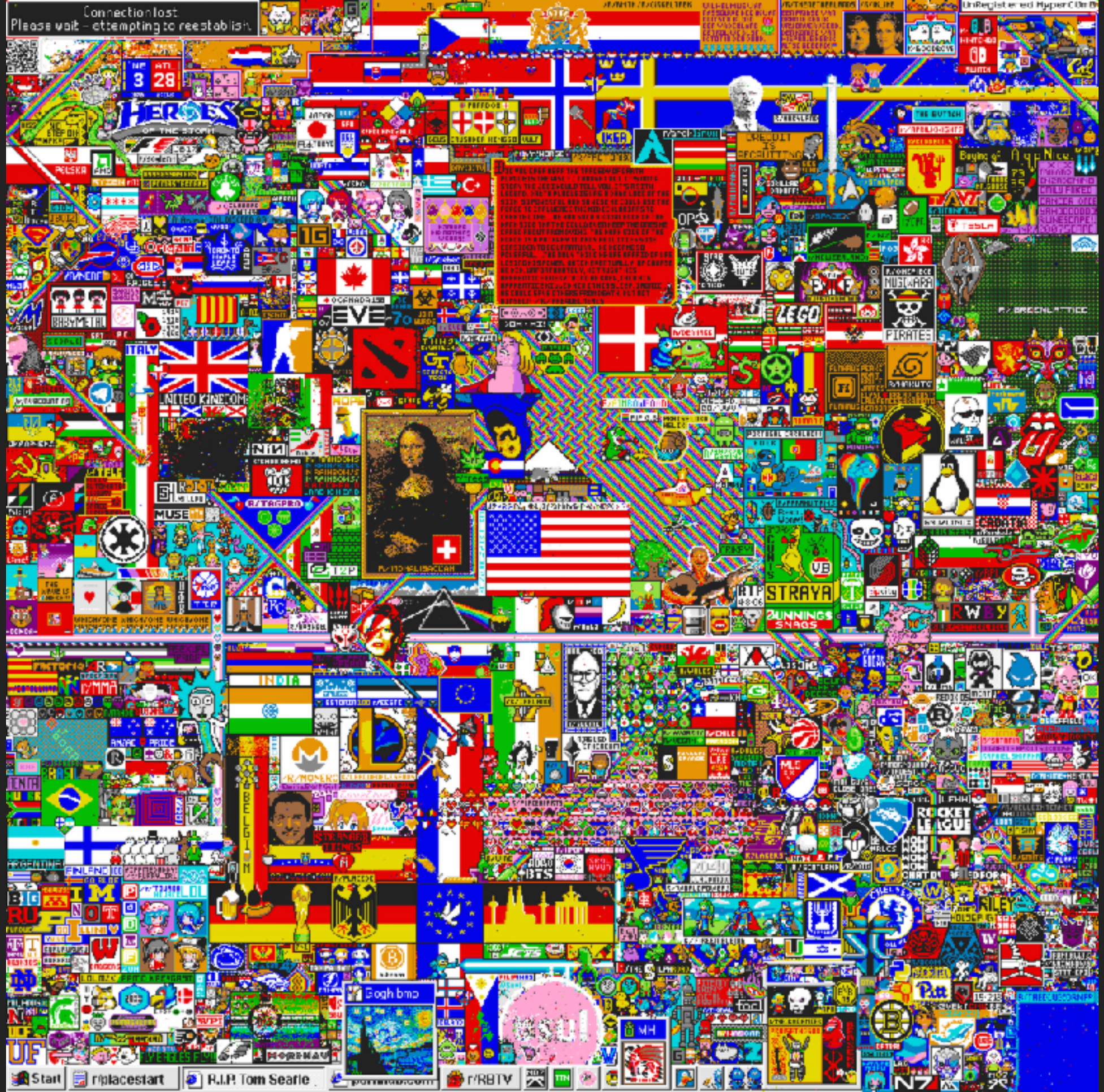
1,000,000 pixels • $1 per pixel • Own a piece of internet history!

SOLD OUT!

Follow @tewy | 13K followers

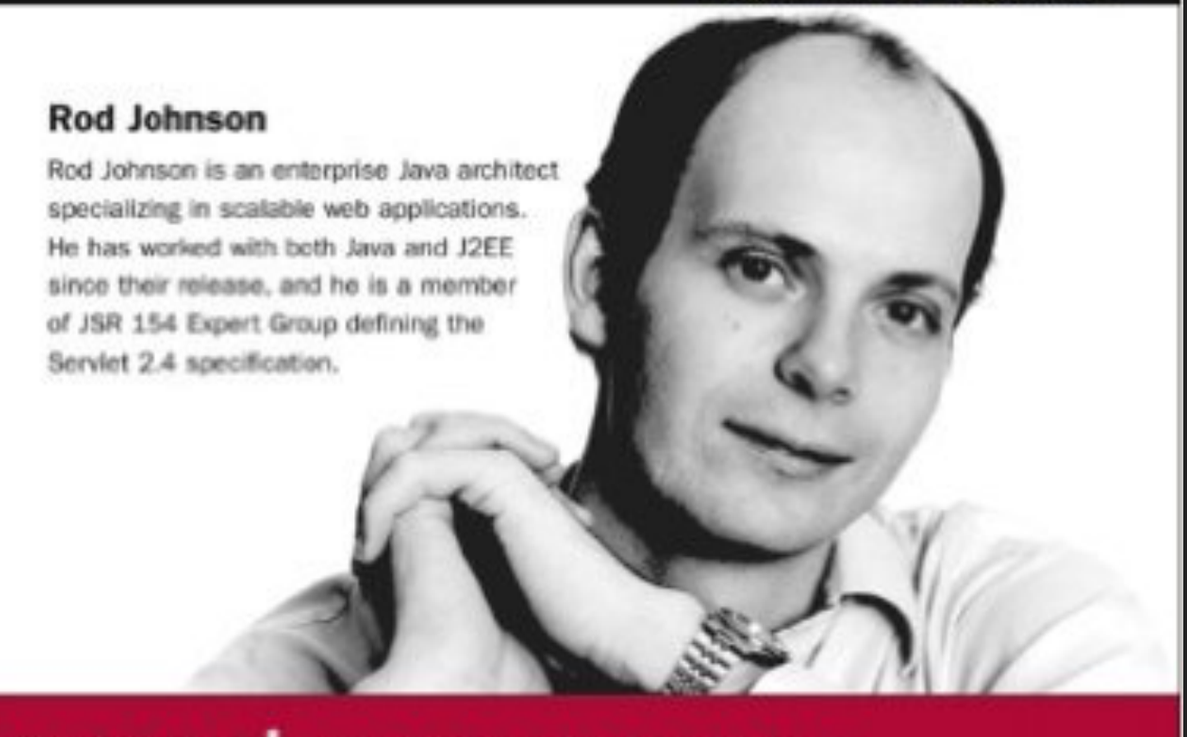
Homepage | Buy Pixels | FAQ | Blog | Pixel List | Press | Testimonials | Contact me




Connection lost
Please wait - attempting to reestablish
HEROES OF THE STORM
EVE
ITALY
UNITED KINGDOM
INDIA
BELGIUM
MUSEUM
PIRATES
LEGO
STRAYA
RWBY
Start
r/placestart
R.I.P. Tom Searle
r/RBTV

SPRING

Programmer to Programmer™



Rod Johnson

Rod Johnson is an enterprise Java architect specializing in scalable web applications. He has worked with both Java and J2EE since their release, and he is a member of JSR 154 Expert Group defining the Servlet 2.4 specification.

expert one-on-one

J2EE™ Design and Development

wrox

Updates, source code, and Wrox technical support at **www.wrox.com**

SPRING

- IoC (Inversion of Control)/DI (Dependency Injection) container
- lightweight
- modular
  - AOP (Aspect-Oriented Programming)
  - JDBC abstraction layer
  - MVC framework
  - Hibernate integration code
  - Jakarta Commons Logging API is the only mandatory dependency
- non-intrusive

Spring Framework Runtime

Data Access/Integration

- JDBC
- ORM
- OXM
- JMS
- Transactions

Web

- WebSocket
- Servlet
- Web
- Portlet

- AOP
- Aspects
- Instrumentation
- Messaging

Core Container

- Beans
- Core
- Context
- SpEL

Test

IOC/DI?

IOC/DI – WHY DO WE EVEN BOTHER?

- Loosely coupled components
- Easier testing
- External configuration
- Plug-in configuration

IOC/DI IN SPRING

- Container
- Beans
  - BeanDefinition class
- BeanFactory and ApplicationContext classes

BEAN SCOPES

- singleton
- prototype
- request
- session
- globalSession
- application
- websocket
- thread*
- custom

CONFIGURATION
XML VS ANNOTATIONS

CLASSPATH SCANNING

- @Configuration and @Bean
- Stereotype annotations:
  - @Component
  - @Repository
  - @Service
  - @Controller

DETAILS

- proxing
- lifecycle callbacks
  - @PostConstruct
  - @PreDestroy
- AOP

CODING TO INTERFACES?

SPRING
HANDS-ON

TINY SPRING IN 30 MINUTES

TINY SPRING – ASSUMPTIONS

- Public constructor injection
- Singletons only
- @Element marker annotation
- Strict type matching
- Interfaces are not supported

TINY SPRING – NEW FEATURES

- @WhenCreated
- @Proto
- @Lazy

SPRING BOOT

SPRING FRAMEWORK

SPRING BOOT

SPRING BOOT

- „Zero Effort Spring"
- Powered by Spring Framework
- Starter dependencies
- Auto-configuration
- Actuator

SPRING INITIALIZR bootstrap your application now

Generate a Gradle Project with Java and Spring Boot 1.5.6

Project Metadata

Artifact coordinates

Group

it.stacja

Artifact

springfunapp

Name

springfunapp

Description

Stacja IT - Modern Spring-based Web App

Package Name

it.stacja.springfunapp

Packaging

Jar

Java Version

1.8

Too many options? Switch back to the simple version.

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Dependencies

Web × Security × Lombok × Thymeleaf × H2 × JPA ×

Generate Project ⌘ + ⏎

AUTO-CONFIGURATION

- @ConditionOnBean / @ConditionOnMissingBean
- @ConditionOnClass / @ConditionOnMissingClass
- @ConditionOnExpression
- @ConditionOnJava
- @ConditionOnJndi
- @ConditionOnProperty
- @ConditionOnResource

MAKE JAR NOT WAR

Josh „starbuxman" Long

SPRING ACTUATOR ENDPOINTS 1/2

- /actuator
- /auditevents
- /autoconfig
- /beans
- /dumps
- /health

SPRING ACTUATOR ENDPOINTS 2/2

- /info
- /metrics
- /mappings
- /shutdown (!!!)
- /trace

SPRING DATA


CouchDB
relax


Redis


Neo4j
the graph database


Cassandra


mongoDB


membase


riak

SQL/NOSQL DATABASES

- Relational
- Graph
- Document
- Key-Value
- Column

SPRING DATA – SUPPORTED TECHNOLOGIES

- JPA
- LDAP
- MongoDB
- Redis
- Cassandra
- Solr
- Couchbase
- DynamoDB
- Elasticsearch
- Hazelcast
- Neo4j
- Gemfire

CONCEPT

- Mapping
  - JPA
- Repositories
- CrudRepository
  - JpaRepository
  - MongoRepository

MAPPING

```
@Entity
public class Data {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String content;

    // getters, setters
}
```

REPOSITORY

```
public interface DataRepository extends JpaRepository<Data, Long> {
  ...
}
```

EXAMPLES 1/2

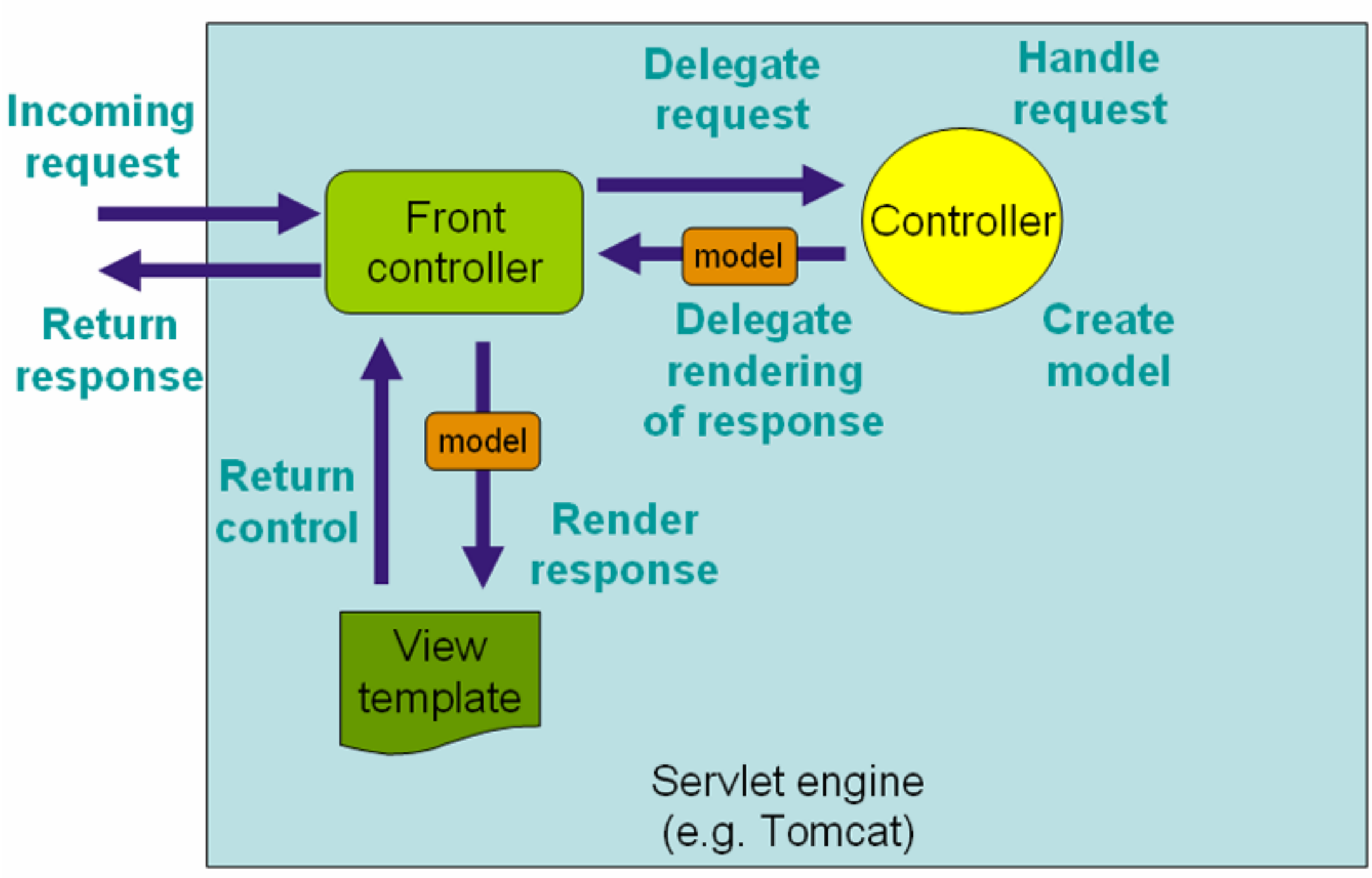
- List<Data> findBy...(String prop)
- List<Data> findBy...And...(String prop1, String prop2)
- List<Data> findTop3By...(String prop)
- List<Data> findDistinctBy...(String prop)
- List<Data> findBy...IgnoreCase(String prop)
- List<Data> findBy...IsNull(String prop)

EXAMPLES 2/2

- Stream<Data> findBy...(String prop)
- Future<Data> findBy...(String prop)
- Page<Data> findAll(Pageable p)
- Page<Data> findBy...(String prop, Pageable p)
- List<Data> findAll(Sort s)
- Page<Data> findBy...(String prop, Pageable p, Sort s)

H2 CONSOLE

SPRING MVC


Incoming request
Return response
Front controller
Delegate request
Handle request
Controller
model
Delegate rendering of response
Create model
model
Return control
Render response
View template
Servlet engine (e.g. Tomcat)

TEMPLATES

VS

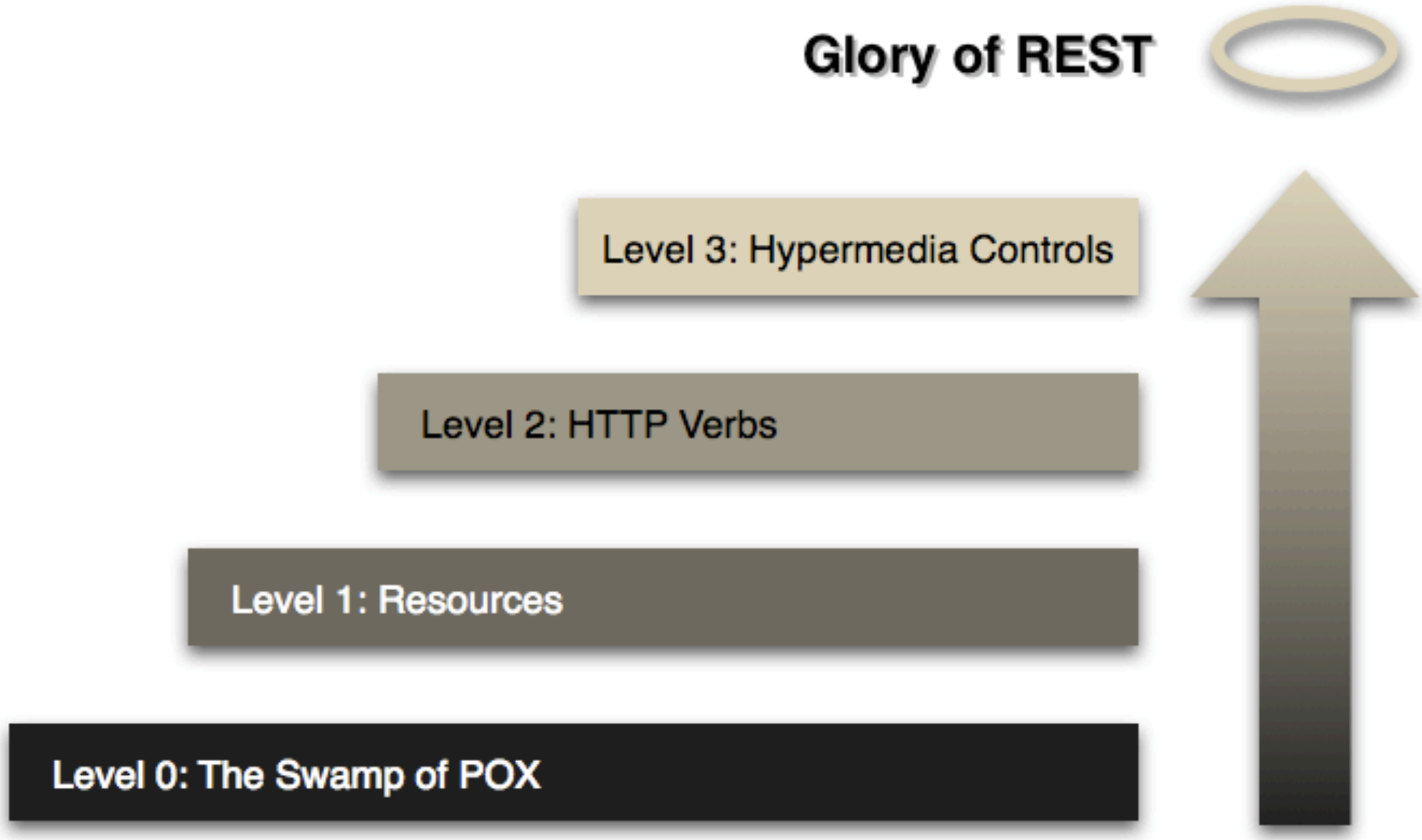
REST API

VIEW TECHNOLOGIES

- Velocity
- FreeMarker
- Groovy Markup Templates
- Thymeleaf
- JSP

FAST START

- @Controller
- @RequestMapping
  - @GetMapping
  - @PostMapping
  - ...
- @ResponseBody
- @RestController
- ResponseEntity


Glory of REST
Level 3: Hypermedia Controls
Level 2: HTTP Verbs
Level 1: Resources
Level 0: The Swamp of POX

HATEOAS

Hypermedia as the Engine of Application State

SSE

WEBSOCKETS

ASYNC

SPRING SECURITY

SPRING SECURITY

- Support for Authentication and Authorization
- Protection aginst common attacks
- Servlet API integration
- Spring MVC and Spring Data integration
- Support for WebSockets

DEFAULTS

- HTTP headers:
  - X-Frame-Options
  - X-XSS-Protection
- CSRF protection
- HTTP Strict Transport Security

REST SECURITY

- Session-based auth is not so bad…
- HTTP Authentication
- OAuth 1/OAuth 2
- JWT
- API tokens

FUTURE

SPRING 5

THE WORLD IS CHANGING

- More and more connected users
- Users' expectations are still rising
- Cloud everywhere…
- Say goodbye to Moor's law

REACTIVE MANIFESTO

www.reactivemanifesto.org

RESPONSIVE

RESILIENT

SCALABLE

MESSAGE-DRIVEN

REACTIVE SYSTEMS

- Responsive
  - System responds in a timely manner
- Resilent
  - System stays responsive in the face of failure (replication, isolation, delegation)
- Scalable
  - System stays responsive under varying workload (automatic resource allocation, sharding)
- Message Driven
  - Asynchronous message-passing, location transparency, back-pressure, non-blocking

WHAT ABOUT REACTIVE PROGRAMMING?

„SUBSET OF ASYNCHRONOUS PROGRAMMING AND A PARADIGM WHERE THE AVAILABILITY OF NEW INFORMATION DRIVES THE LOGIC FORWARD RATHER THAN HAVING CONTROL FLOW DRIVEN BY A THREAD-OF-EXECUTION"

O_o

Internet

„GENERAL PROGRAMMING TERM THAT IS FOCUSED ON REACTING TO CHANGES, SUCH AS DATA VALUES OR EVENTS."

O_o

Internet

ASYNCHRONOUS EVENT-DRIVEN PROGRAMMING

PROJECT REACTOR

@Controller, @RequestMapping

Spring MVC

Spring Web Reactive

Servlet API

Reactive HTTP

Servlet Container

Servlet 3.1, Netty, Undertow

PROJECT REACTOR

- Mono
- Flux
- Support for RxJava

KOTLIN