

Josh Long (龙之春)

@starbuxman

joshlong.com

josh.long@springsource.com

slideshare.net/joshlong

github.com/joshlong

<http://spring.io>

HAVE YOU SEEN



<http://joshlong.github.io/have-you-seen-spring-lately/>

Josh Long (龙之春)

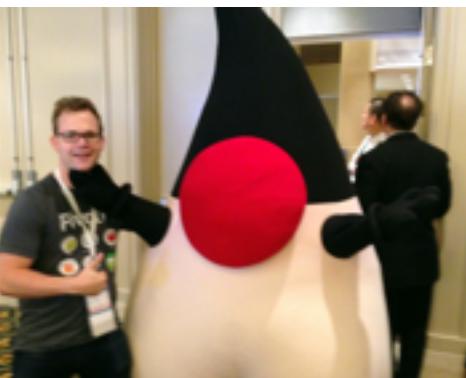
Spring Developer Advocate

@starbuxman | josh.long@springsource.com

Jean Claude
van Damme!



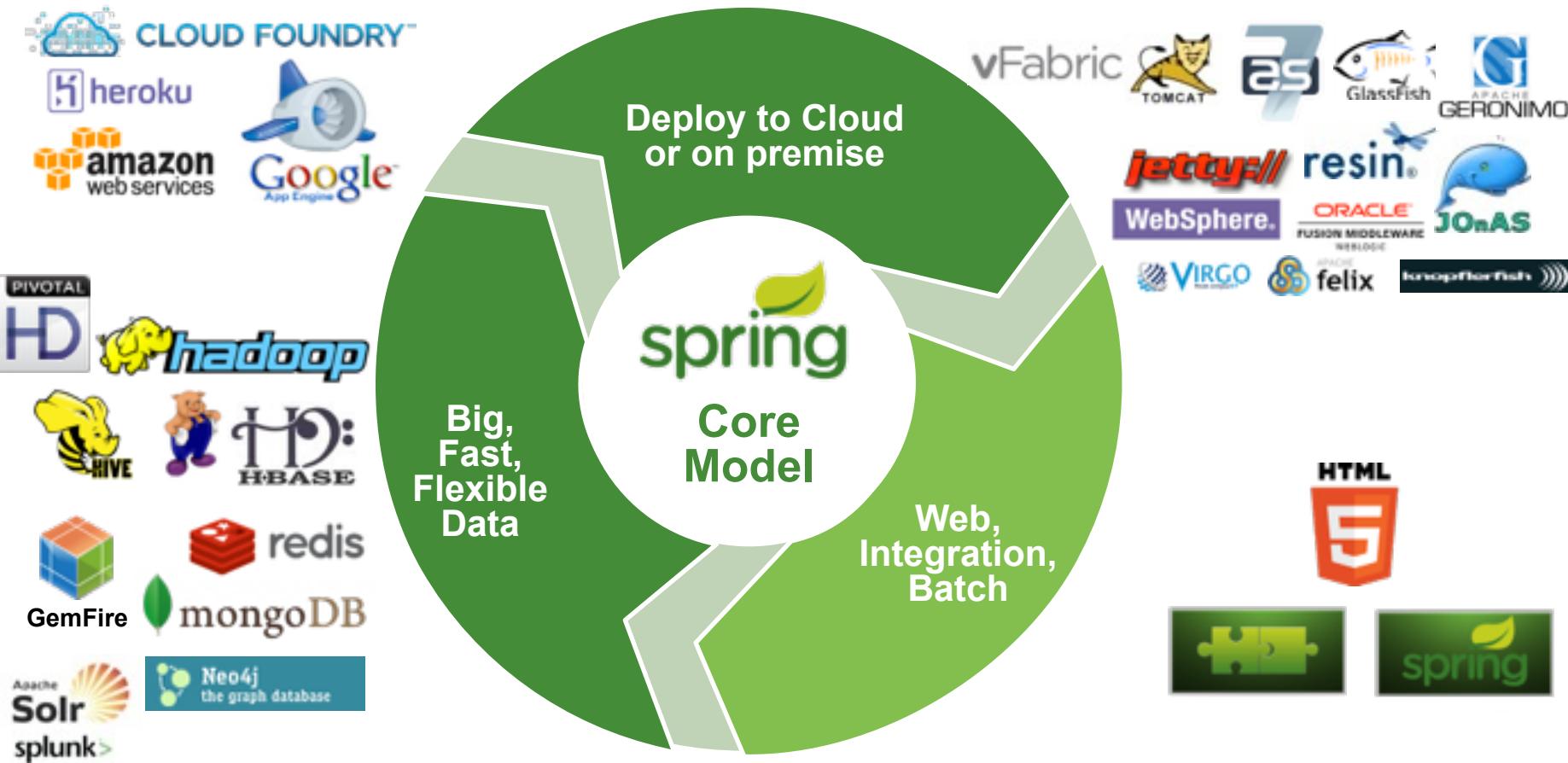
Java mascot Duke

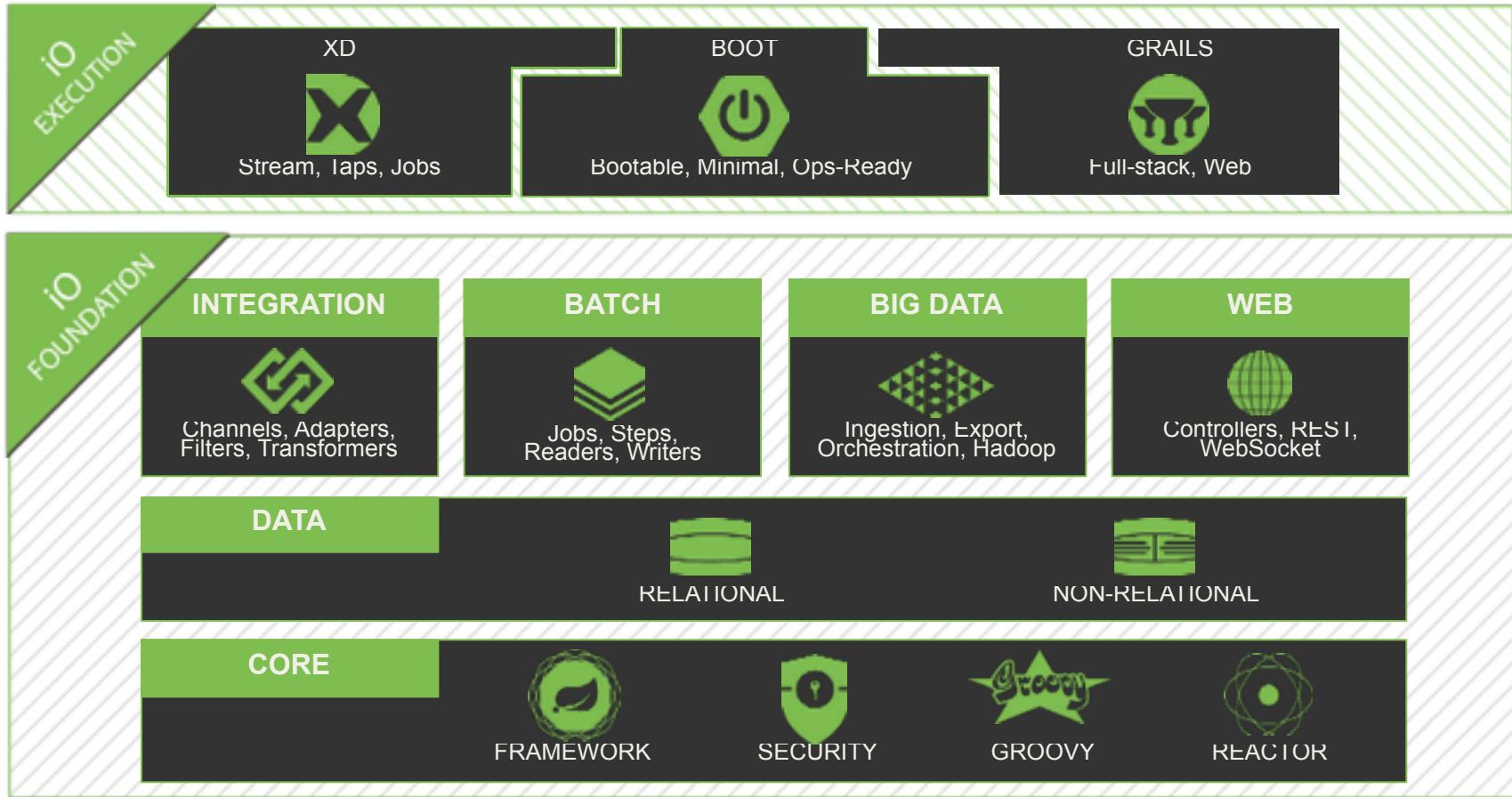


some thing's I've authored...



Pivotal







The screenshot shows the official Spring framework website at <https://spring.io>. The page features a large banner with the text "Let's build a better Enterprise." and a subtext box stating: "Spring helps development teams everywhere build simple, portable, fast and flexible JVM-based systems and applications." Below the banner, there are three main sections: "BUILD ANYTHING" (with a leaf icon), "RUN ANYWHERE" (with a cloud icon), and "REST ASSURED" (with a circular seal icon). Each section contains a brief description of the Spring framework's capabilities.

BUILD ANYTHING

Write clean, testable code against the infrastructure components of your choice and accomplish any task – without re-inventing the wheel.

RUN ANYWHERE

Keep it portable – Spring-based apps run anywhere the JVM does. Deploy standalone, in an app server, on a PaaS or all of the above.

REST ASSURED

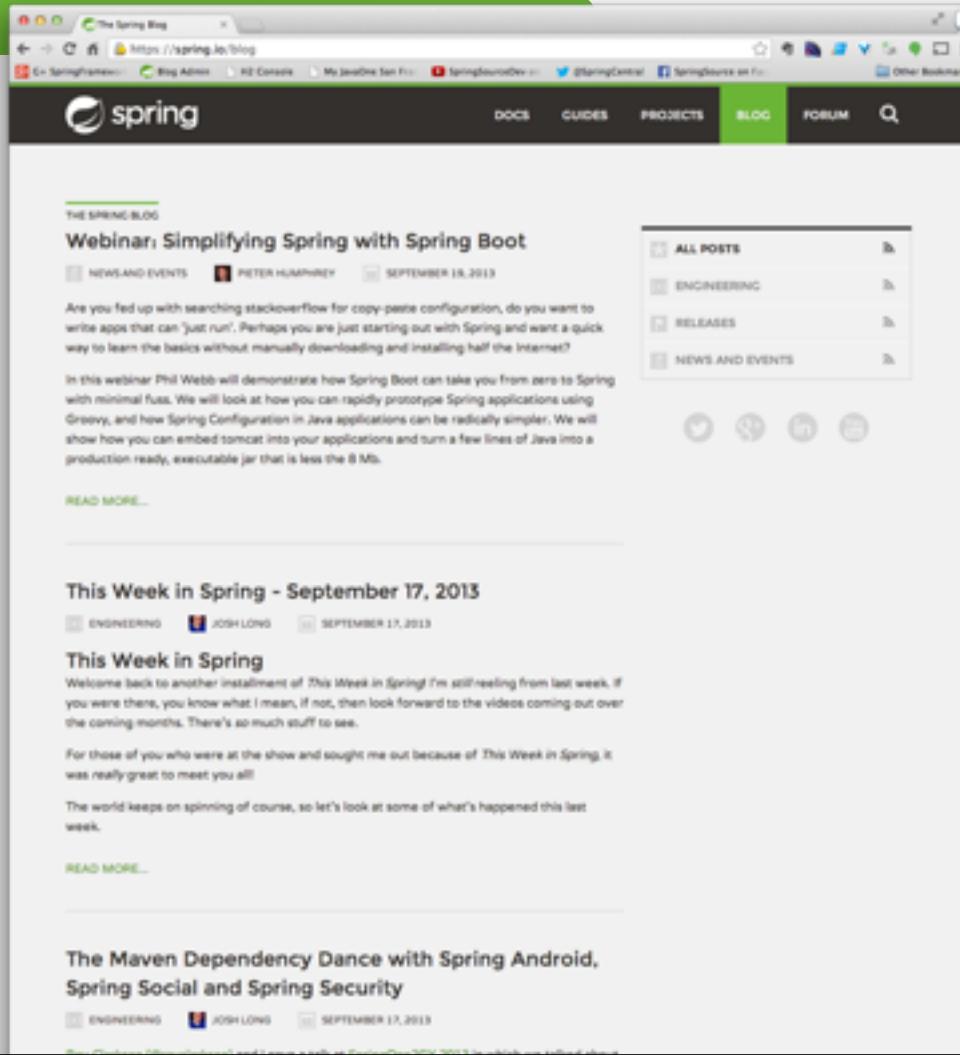
Code with confidence – Spring provides an open programming model that is comprehensive, cohesive, widely understood and well-supported.

The screenshot shows the 'Guides' section of the Spring website. At the top, there's a navigation bar with links for 'Docs', 'Guides' (which is highlighted in green), 'Projects', 'Blog', 'Forum', and a search icon. Below the navigation is a heading 'Guides' with a subtext: 'Whatever you're building, these guides are designed to get you productive as quickly as possible – using the latest Spring project releases and techniques as recommended by the Spring team.' A note below says 'Have a suggestion for a new guide? Let us know at [@springcentral](#).' There's a search bar labeled 'Find a guide...'. The main content area is titled 'Getting Started Guides' and features a grid of 12 items, each with an icon and a brief description:

Icon	Section	Description
	Building a RESTful Web Service	Learn how to create a RESTful web-service with Spring.
	Scheduling Tasks	Learn how to schedule tasks with Spring.
	Detecting a Device	Learn how to use Spring to detect the type of device that is accessing your web site.
	Consuming a RESTful Web Service	Learn how to retrieve web page data with Spring's RestTemplate.
	Building Java Projects with Gradle	Learn how to build a Java project with Gradle.
	Building Java Projects with Maven	Learn how to build a Java project with Maven.
	Uploading Files	Learn how to build a Spring application that accepts multi-part file uploads.
	Authenticating a User with LDAP	Learn how to secure an application with LDAP.
	Registering an Application with Facebook	Learn how to register an application to integrate with Facebook.
	Messaging with RabbitMQ	Learn how to create a simple publish-and-subscribe application with Spring and RabbitMQ.
	Accessing Data with Neo4j	Learn how to persist objects and relationships in Neo4j's NoSQL data store.

The screenshot shows the Spring Projects page on a web browser. The page features a header with the Spring logo and navigation links for Docs, Guides, Projects (which is the active tab), Blog, Forum, and Search. Below the header, a section titled "Projects" is displayed, followed by a brief introduction about the modular nature of Spring projects. Six project cards are arranged in a 2x3 grid:

- SPRING BOOT**: Takes an opinionated view of building Spring applications and gets you up and running as quickly as possible.
- SPRING FRAMEWORK**: Provides core support for dependency injection, transaction management, web apps, data access, messaging and more.
- SPRING XD**: Simplifies the development of big data applications by addressing ingestion, analytics, batch jobs and data export.
- SPRING DATA**: Provides a consistent approach to data access – relational, non-relational, map-reduce, and beyond.
- SPRING INTEGRATION**: Supports the well-known Enterprise Integration Patterns via lightweight messaging and declarative adapters.
- SPRING BATCH**: Simplifies and optimizes the work of processing high-volume batch operations.



The screenshot shows the Spring.io blog homepage with a green header bar. The header includes the Spring logo, navigation links for DOCS, GUIDES, PROJECTS, BLOG (which is highlighted in green), FORUM, and a search icon. Below the header, there's a sidebar with categories: ALL POSTS, ENGINEERING, RELEASES, and NEWS AND EVENTS, each with a 'Read more' link. The main content area displays three blog posts:

- Webinar: Simplifying Spring with Spring Boot**
by Pieter Humphrey | September 18, 2013

Are you fed up with searching StackOverflow for copy-paste configuration, do you want to write apps that can 'just run'. Perhaps you are just starting out with Spring and want a quick way to learn the basics without manually downloading and installing half the Internet?

In this webinar Phil Webb will demonstrate how Spring Boot can take you from zero to Spring with minimal fuss. We will look at how you can rapidly prototype Spring applications using Groovy, and how Spring Configuration in Java applications can be radically simpler. We will show how you can embed tomcat into your applications and turn a few lines of Java into a production ready, executable jar that is less than 8 Mb.

[READ MORE...](#)
- This Week in Spring - September 17, 2013**
by Josh Long | September 17, 2013

Welcome back to another installment of This Week in Spring! I'm still reeling from last week. If you were there, you know what I mean; if not, then look forward to the videos coming out over the coming months. There's so much stuff to see.

For those of you who were at the show and sought me out because of This Week in Spring, it was really great to meet you all!

The world keeps on spinning of course, so let's look at some of what's happened this last week.

[READ MORE...](#)
- The Maven Dependency Dance with Spring Android, Spring Social and Spring Security**
by Josh Long | September 17, 2013

The Maven Dependency Dance with Spring Android, Spring Social and Spring Security

Welcome to the new Spring.io forum! If this is your first visit, be sure to check out the [FAQ](#) by clicking the link above, and for security reasons, use the [forgot password](#) link to reset your password. You may have to register before you can post; click the [register](#) link above to proceed. To start viewing messages, select the forum that you want to visit from the selection below.

FORUMS

FORUMS	LATEST ACTIVITY	MY SUBSCRIPTIONS
Directory	Topics · Posts · Last Post	
Spring Projects		
Announcements	363 · 1,014	 New "Spring 4.0" Forum Issues By davidfowl Yesterday, 06:12 PM
Container	18,400 · 81,743	 Documentation not online By marcosme Sep 13th, 2013, 01:29 PM
Data	17,299 · 60,498	 Configuring a sessionFactory in Spri... By DrOof Sep 13th, 2013, 07:05 AM
Sub-Forums:		
Data REST (66)		JDBC (60)
JPA / ORM (74)		NoSQL (1,611)
GemFire (6817)		
XD	1 · 16	View Source

CORE SPRING TRAINING
San Francisco, 29 Oct 2013

[Learn More](#)

Welcome to vBulletin!

Welcome to your vBulletin forum! You can click "Edit Site" above for site administration options.

To add or remove this module, click "Edit Site", then "Edit Page".

Latest Topics

 [Activator shutdown endpoint ...](#)
[trentirosebaum](#)
I have been able to get the applicat...
Today, 03:54 AM

 [Not able to run the Spring-Bo...](#)
[benjyha](#)
I was trying out spring-boot samp...
Today, 02:04 AM

 [Kansas City Java User Group ...](#)
[RobWinch](#)
I am going to be presenting "Sprin...
Yesterday, 05:02 PM

[View All](#)

Recent Blog Posts

There are no results that meet this criteria.



PIVOTAL TRACKER



Spring projects - GitHub

GitHub | Search or type a command

Explore Features Enterprise Blog Sign up Sign in

Repositories Members

Find a repository... Search All Branches Forks Merges

spring-boot Spring Boot Last updated 20 minutes ago Java 9k 200 37m

spring-integration Spring Integration provides an extension of the Spring programming model to support the well-known Enterprise Integration Patterns (EIP) Last updated an hour ago Java 9k 200 37m

grails-data-mapping Grails Data Mapping Project Last updated 2 hours ago Java 9k 200 37m

spring-data-redis Provides support to increase developer productivity in Java when using Redis, a key-value store. Uses familiar Spring concepts such as template classes for core API usage and lightweight repository style data access. Last updated 3 hours ago Java 9k 200 37m

spring-cloud Library to let cloud applications connect to services Last updated 11 hours ago Java 9k 200 37m

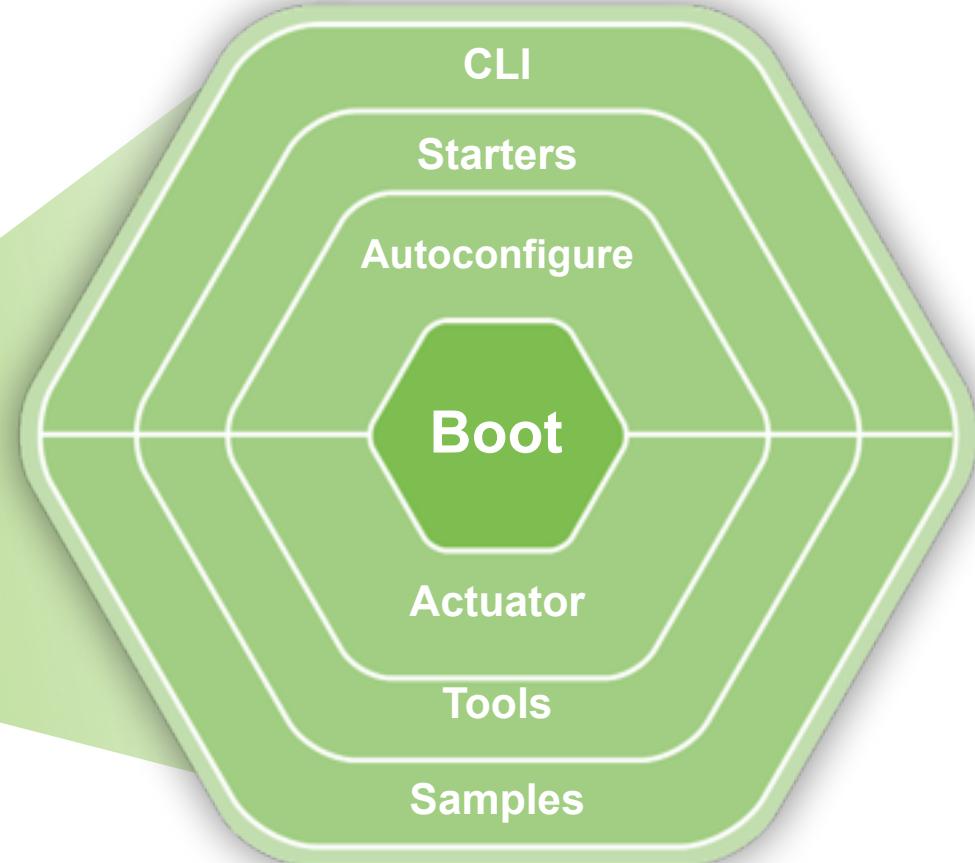
spring-ide Spring Development Environment for Eclipse Last updated 11 hours ago Java 9k 200 37m

spring-loaded Java agent that enables class reloading in a running JVM Last updated 13 hours ago Java 9k 200 37m

spring-xd Spring XD makes it easy to solve common big data problems such as data ingestion and export, real-time analytics, and batch workflow orchestration. Last updated 13 hours ago Java 9k 200 37m

spring-batch

Bootstrap your productivity



**Rob Winch** @rob_winch

@Controller

```
class ThisWillActuallyRun {  
    @RequestMapping("/")  
    @ResponseBody  
    String home() {  
        "Hello World!"  
    }  
}
```

6 Aug

[Details](#)[Reply](#)[Retweeted](#)[Favorite](#)[More](#)**25**

RETWEETS

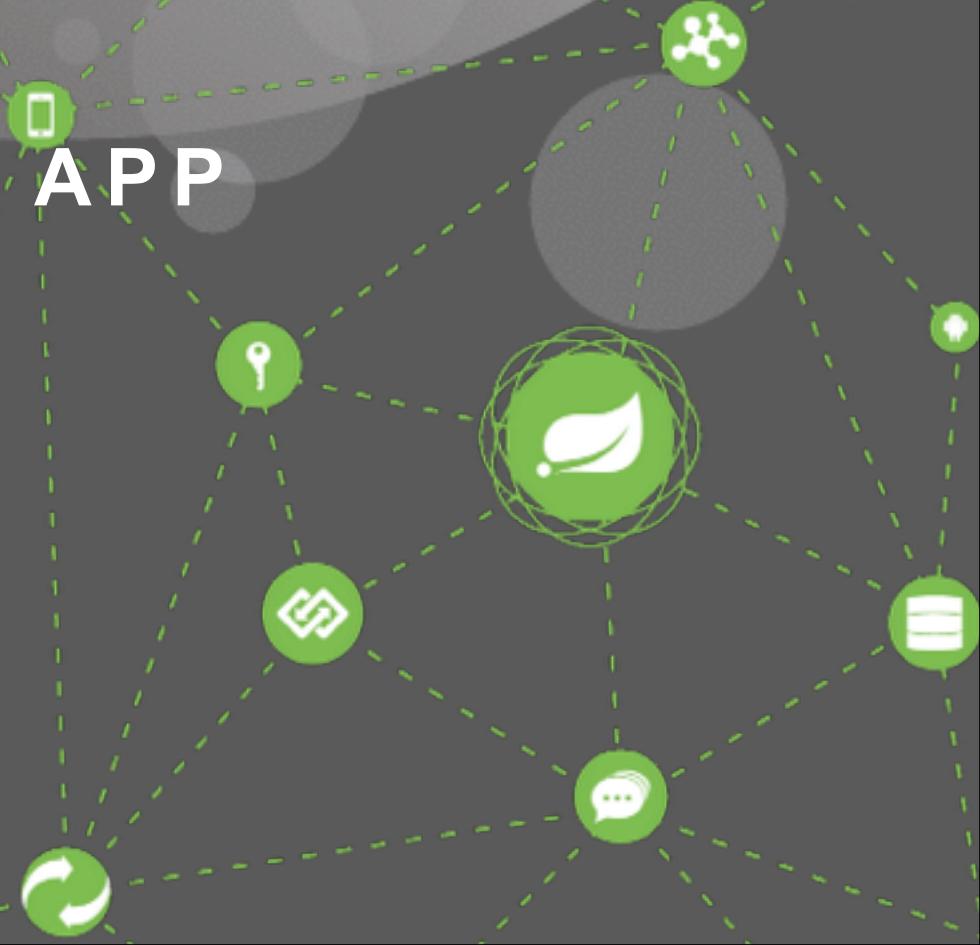
8

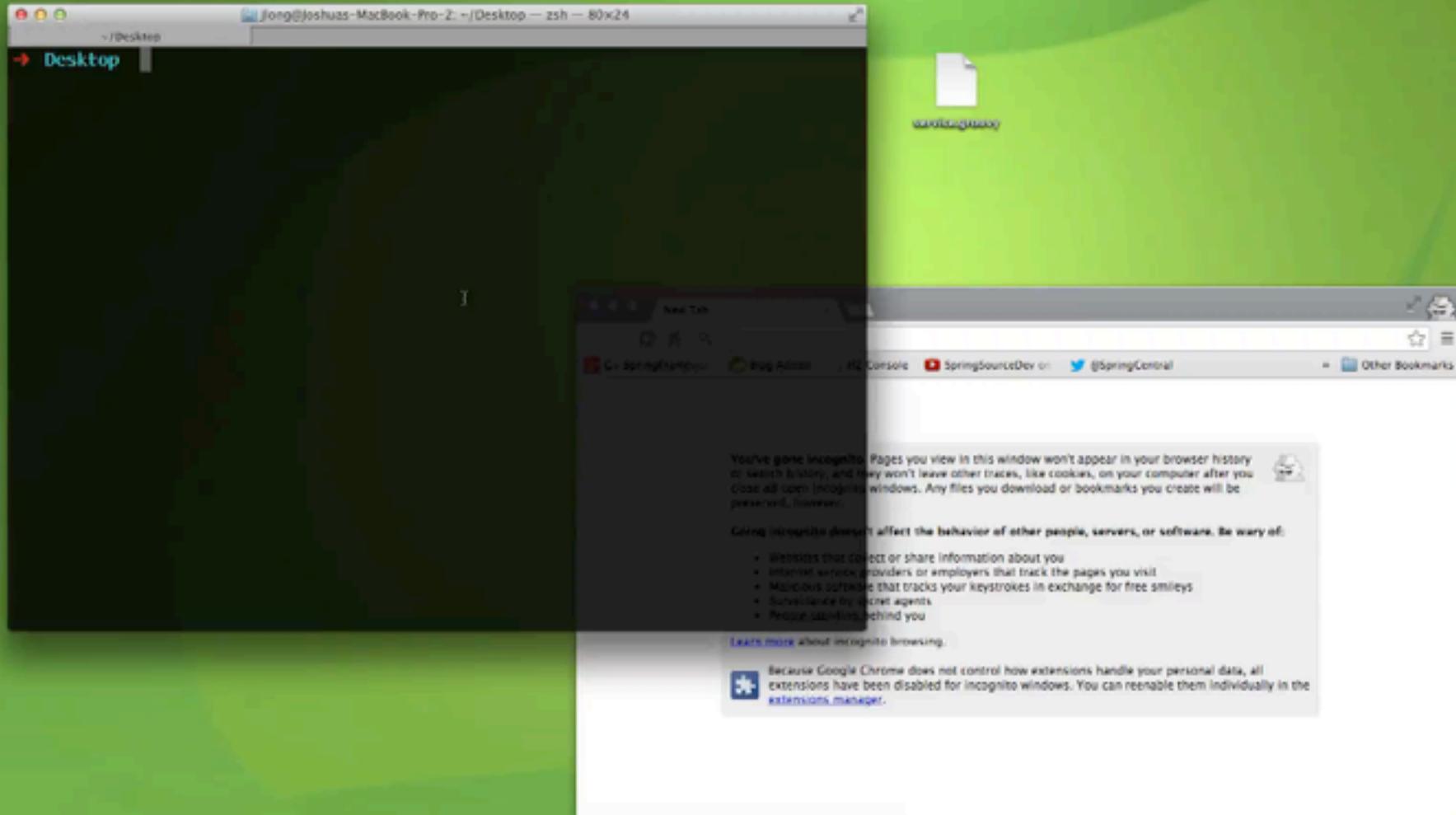
FAVORITES



3:12 PM - 6 Aug 13 · Details

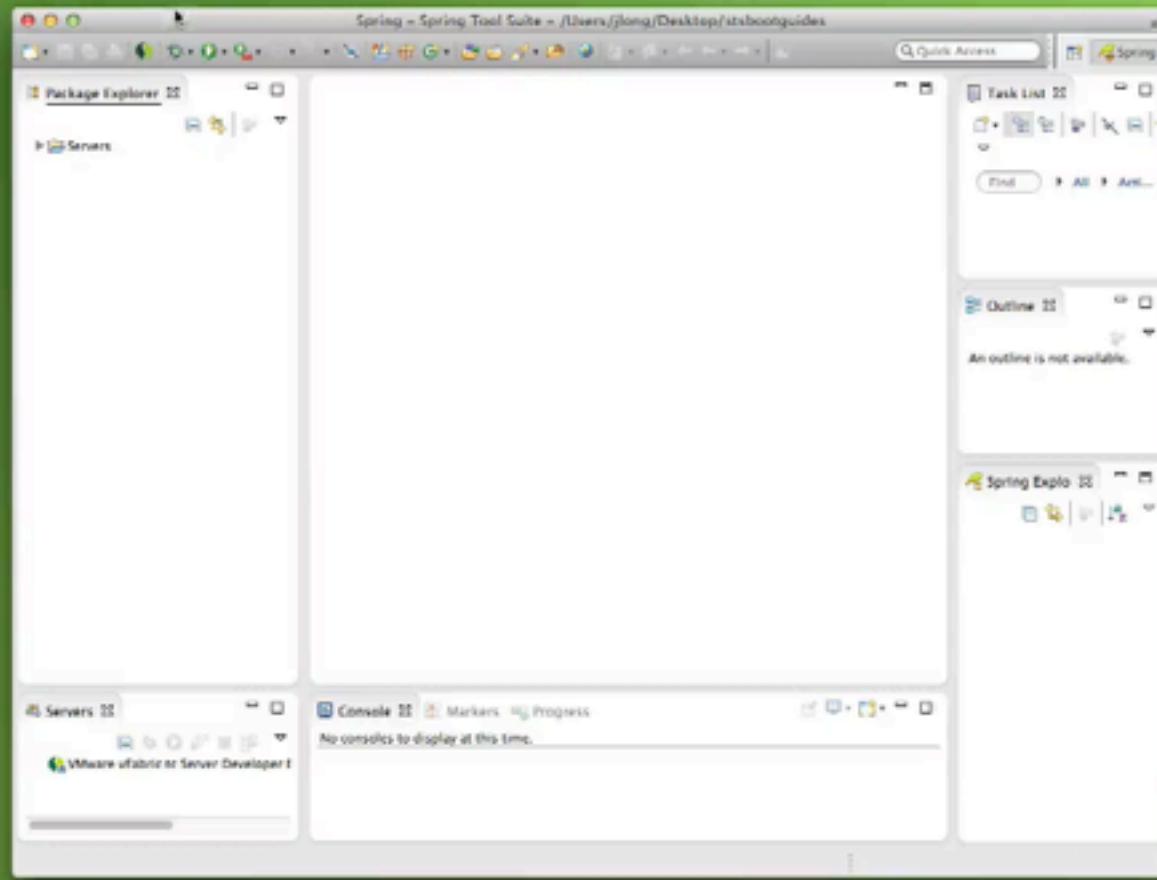
DEMO: A GROOVY BOOT APP





DEMO: BOOT, GETTING STARTED GUIDES, & STS

The background features a dark grey gradient with several semi-transparent light grey circles of varying sizes. Overlaid on this is a network diagram composed of green circular nodes connected by dashed green lines. The nodes contain icons representing different services or components: a gear, a leaf, a key, a speech bubble, a document, a diamond shape, and a circular arrow. The network is centered around a node with a circular arrow icon at the bottom center.



Spring loves the JVM: Java (5,6,7, and 8), Groovy, Scala, etc.

Scala

```
val applicationContext =  
    new FunctionalConfigApplicationContext( classOf[ServiceConfiguration])  
  
// ServiceConfiguration.scala  
class ServiceConfiguration extends FunctionalConfiguration {  
    importClass(classOf[DataSourceConfiguration])  
  
    val dataSource : DataSource = getBean("dataSource")  
  
    val jdbcTemplate = bean() {  
        new JdbcTemplate(dataSource)  
    }  
    val jdbcTransactionManager = bean("txManager") {  
        new JdbcTransactionManager(dataSource)  
    }  
}
```

Groovy

```
def bb = new BeanBuilder()
bb.loadBeans("classpath: * SpringBeans.groovy")
def applicationContext = bb.createApplicationContext()

// MySpringBeans.groovy
import o.s.f.jdbc.core.JdbcTemplate
import o.s.f.jdbc.datasource.DataSourceTransactionManager

beans {

    jdbcTemplate(JdbcTemplate) {
        dataSource = dataSource
    }

    transactionManager(DataSourceTransactionManager) {
        dataSource = dataSource
    }
}
```

Java

```
ApplicationContext applicationContext =  
    new AnnotationConfigApplicationContext( ServiceConfiguration.class );
```

```
// ServiceConfiguration.java  
@Configuration  
@ComponentScan  
@Import(DataSourceConfiguration.class)  
@EnableTransactionManagement  
class ServiceConfiguration {  
  
    @Bean  
    public JdbcTemplate jdbcTemplate(DataSource dataSource) throws Exception {  
        return new JdbcTemplate( dataSource );  
    }  
  
    @Bean  
    public PlatformTransactionManager jdbcTransactionManager(DataSource dataSource){  
        return new JdbcTransactionManager (dataSource);  
    }  
}
```

a quick note on



Delayed again! from 09/2013 to as late as 03/2014!

....

Meanwhile, even **C++** has lambdas...

```
int main()
{
    // Create a vector object that contains 10 elements.
    vector<int> v;
    for (int i = 0; i < 10; ++i) {
        v.push_back(i);
    }

    // Count the number of even numbers in the vector by
    // using the for_each function and a function object.
    int evenCount = 0;
    for_each(v.begin(), v.end(), FunctorClass(evenCount));

    // Print the count of even numbers to the console.
    cout << "There are " << evenCount
        << " even numbers in the vector." << endl;
}
```

a quick note on



IDE Support:

IntelliJ IDEA has had Java 8 support **for a year**

Eclipse won't have any until June 2014 (...!!)

Eclipse-based Spring Tool Suite (STS) has beta Java 8 support.

a quick note on



Spring 4.0 to be GA against Developer Preview by end of 2013.

Method references are a great fit for Spring!

```
JdbcTemplate jdbcTemplate;  
  
// method references  
private Customer map(ResultSet rs, int rowNum)  
    throws SQLException {  
    return new Customer( rs.getString("name"), rs.getInt("age") );  
}  
  
// let it satisfy the `RowMapper` functional interface  
Customer customer = jdbcTemplate.queryForObject(  
    "select name, age from customers ",  
    this::map);
```

a quick note on



Spring 4.0 to be GA against Developer Preview by end of 2014.

lambas are a great fit for Spring *and* Twitter!

A screenshot of a Twitter post from user @starbuxman. The post includes a profile picture of a man, the user's name and handle, the date (21 Sep), the tweet text, code snippets, interaction buttons, and a retweet/favorite count.

Josh Long @starbuxman
Java 8/Spring 4.0 already work well together: #lambdas #ftw
21 Sep

jdbc.queryForObject(sql,
 (rs, rowNum) -> new Customer(rs.getLong("id")));

[Collapse](#) [Reply](#) [Delete](#) [Favorite](#) [More](#)

28	6
RETWEETS	FAVORITES

Profile icons for the user and six others are shown below the interaction buttons.

a quick note on



JSR-310 - Date and Time

```
// declarative date format
import java.time.*;
import org.springframework.format.annotation.*;

public class Customer {

    @DateTimeFormat(iso=ISO.DATE)
    private LocalDate birthDate;

    @DateTimeFormat(pattern="M/d/yy h:mm")
    private LocalDateTime lastContact;

}
```

a quick note on
Java 8

Repeatable Annotations

```
@Scheduled(cron = "0 0 12 * * ?")
@Scheduled(cron = "0 0 18 * * ?")
public void performTempFileCleanup() {
    ...
}
```

LET'S TURN OUR GAZE TO A TIME, LONG AGO, WHEN...

spring 



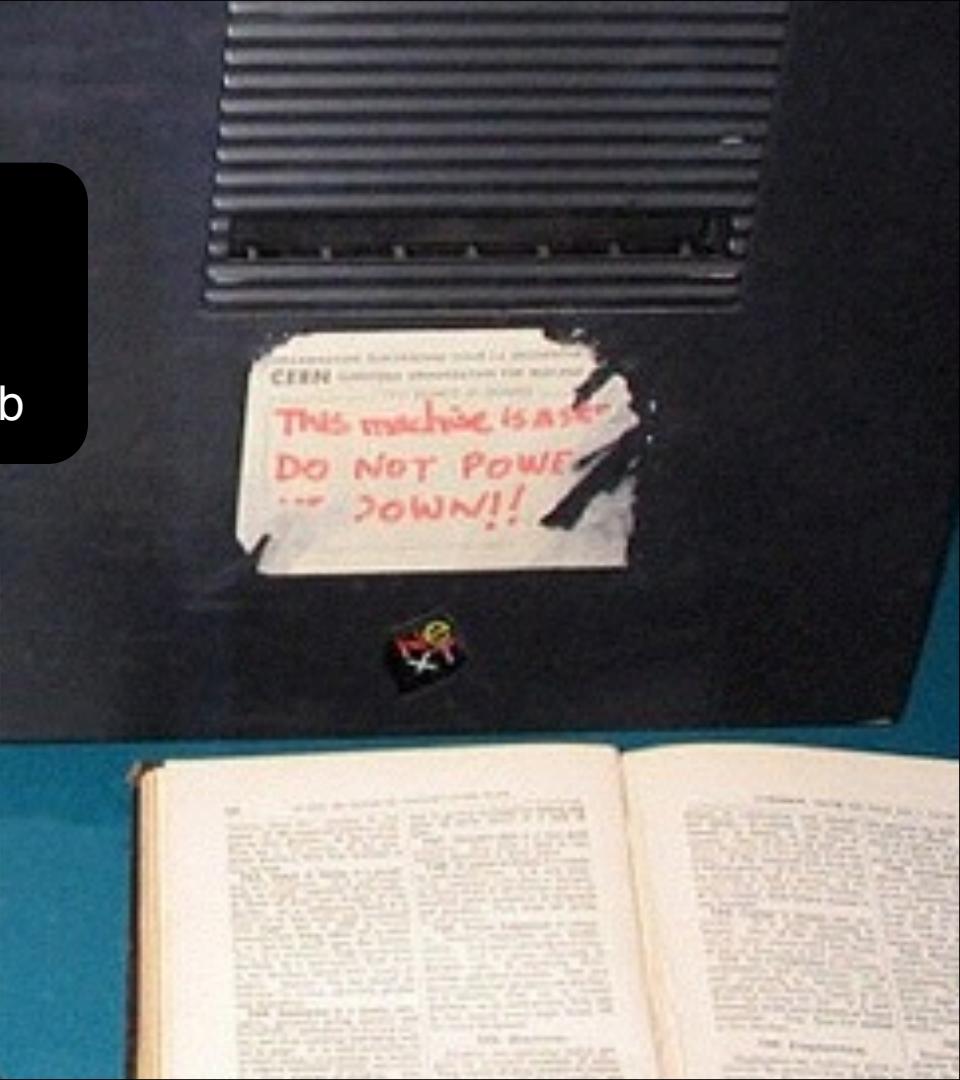
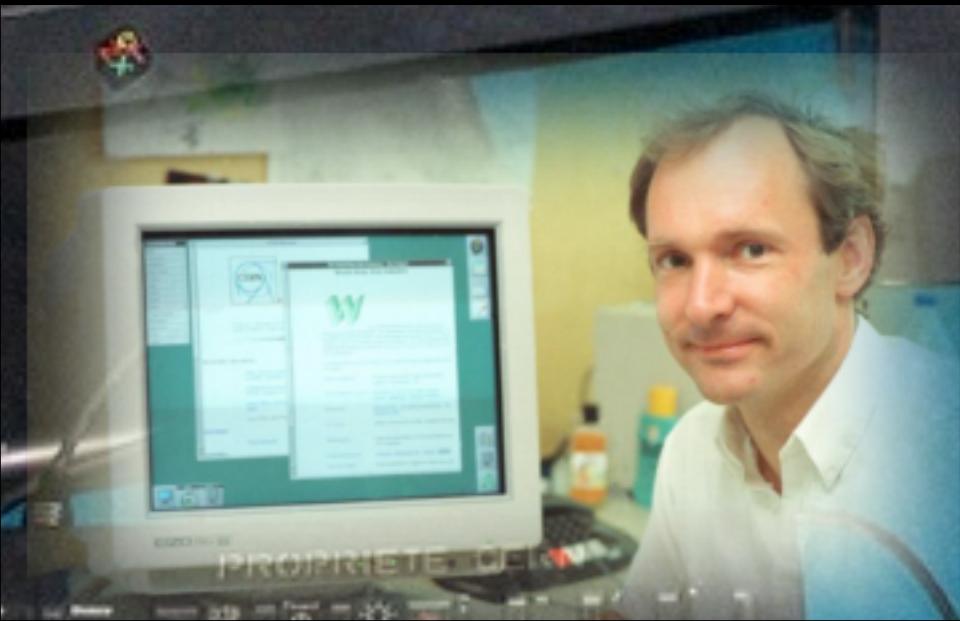
Microsoft Internet Explorer
Version 1.0
Windows 3.1
Windows 95
Windows 98
Windows ME
Windows XP
Windows Vista
Windows 7
Windows 8
Windows 10

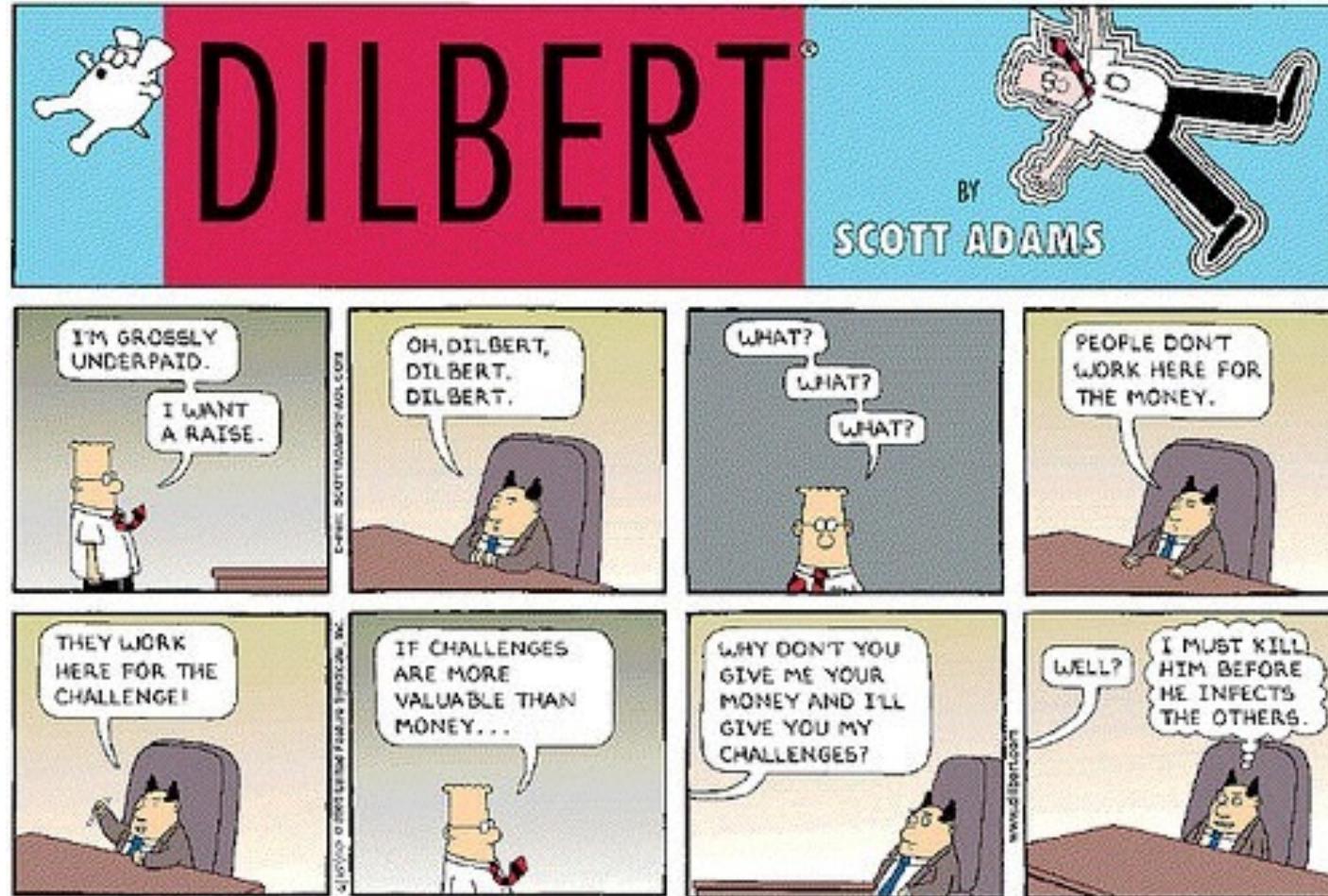
dumb clients roamed the earth... like dinosaurs

THEN MAGIC HAPPENED...

1989

magic happens: Tim Berners Lee &
Robert Cailliau create the world wide web







BACKBONE.JS



ANGULARJS
by Google



The screenshot shows a Java-based PC Emulator window with a terminal session. The terminal displays the following text:

```
Real Time Clock Driver v1.12ac
JS clipboard: I/O at 0x03c0
Serial: 8250/16550 driver $Revision: 1.90 $ 4 ports, IRQ sharing disabled
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a XScale
RAMDISK driver initialized: 16 RAM disks of 4096K size 1024 blocksize
loop: loaded (max 8 devices)
Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx
hda: JSLinux HARDDISK, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
hda: max request size: 128KiB
hda: 116736 sectors (59 MB) w/256KiB Cache, CHS=115/16/63
    hda: unknown partition table
TCP cubic registered
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI Shortcut mode
Time: pit clocksource has been installed.
VFS: Mounted root (ext2 filesystem) readonly.
Freeing unused kernel memory: 128k freed
Booted in 5.236 s
Welcome to JS/Linux
/var/root # ls -la
total 8
drwxr-xr-x    2 root      root          100 May  6 22:29 .
drwxrwxrwt    4 root      root          80 May  6 22:29 ..
-rw-----    1 root      root           7 May  6 22:29 .ash_history
lrwxrwxrwx    1 root      root          9 May  6 22:29 dos -> /root/dos
-rw-r--r--    1 root      root         242 May  6 22:29 hello.c
/var/root #
```

The terminal window has a "Clear clipboard" button in the top right corner. The URL bar shows "tellard.org/jslinux/".

SO HOW DO OUR SMART CLIENTS CONNECT TO OUR SERVICES?

spring 



when someone uses SOAP because it's "simple"

REST Is..

- URI's identify resources
- HTTP verbs describe a limited set of operations that can be used to manipulate a resource
 - GET
 - DELETE
 - PUT
 - POST
 - less used other verbs
- Headers help describe the messages

The Richardson Maturity Model

Level 0: swamp of POX

Uses HTTP mainly as a tunnel through one URI
e.g., **SOAP**, **XML-RPC**

Usually features on HTTP verb (**POST**)

<http://martinfowler.com/articles/richardsonMaturityModel.html>

The Richardson Maturity Model

Level 1: resources

Multiple URIs to distinguish related nouns

e.g., **/articles/1**, **/articles/2**, vs. just **/articles**

<http://martinfowler.com/articles/richardsonMaturityModel.html>

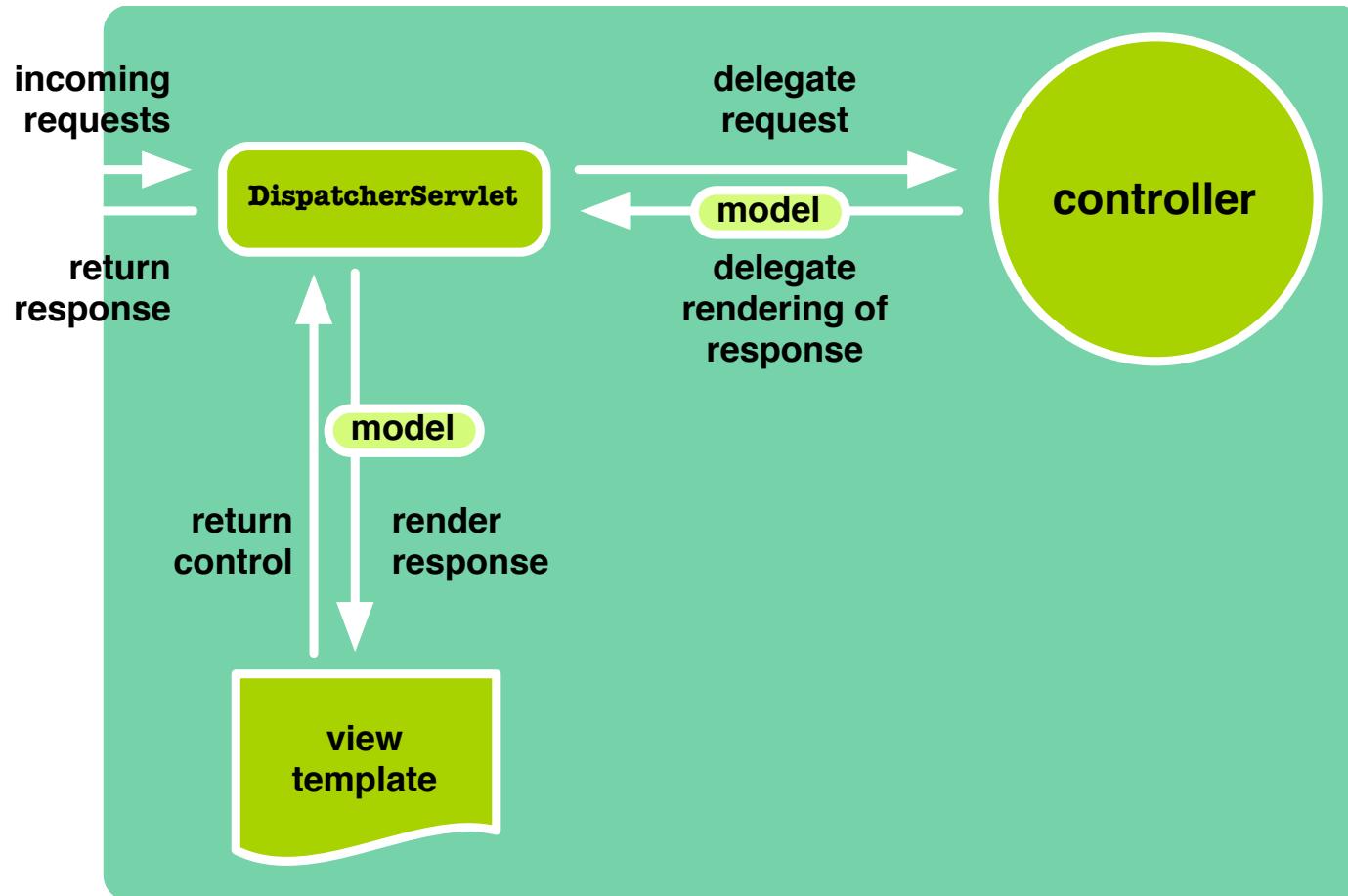
The Richardson Maturity Model

Level 2: HTTP verbs

leverage transport-native properties to enhance service
e.g., **HTTP GET** and **PUT** and **DELETE** and **POST**

Uses idiomatic HTTP controls like status codes, headers

<http://martinfowler.com/articles/richardsonMaturityModel.html>



```
import static ...RequestMethod.* ;
import static ...MediaType.* ;

@RestController
@RequestMapping (value = “/users/”)
class UserController {

    @RequestMapping (method = DELETE, value = “/{user}”)
    User delete(@PathVariable Long user) {
        return crmService.removeUser(user);
    }

    @RequestMapping (method = GET, value = “/{user}”)
    User read(@PathVariable Long user) {
        return crmService.findById(user);
    }

    ..
}
```

DEMO: BUILDING REST WEB SERVICES

The background features a dark grey gradient with several semi-transparent light grey circles of varying sizes. Overlaid on this are several green circular nodes connected by dashed green lines, forming a network. The nodes contain icons representing different web services: a smartphone, a key, a leaf inside a circular border, a document with horizontal lines, a speech bubble, and a circular arrow.

H2 Console Apache Tomcat/7.0.32 - Advanced Rest Client App

localhost:8001/h2/login.jsp?sessionId=4f6dd42e132e38693b5e0e3a21f7de...

Apps G+ SpringFramework Blog Admin H2 Console SpringSourceDev on Twitter @SpringCentral SpringSource on Facebook Other Bookmarks

English Preferences Tools Help

Login

Saved Settings:

Setting Name:

Driver Class:

JDBC URL:

User Name:

Password:

The Richardson Maturity Model

Level 3: Hypermedia Controls (aka, HATEOAS)

No *a priori* knowledge of service required

Navigation options are provided by service and *hypermedia* controls

Promotes longevity through a *uniform interface*

<http://martinfowler.com/articles/richardsonMaturityModel.html>

<http://roy.gbiv.com/untangled/2008/rest-apis-must-be-hypertext-driven>

The Richardson Maturity Model

Level 3: Hypermedia Controls (aka, HATEOAS)

Links provide possible navigations from a given resource

Links are dynamic, based on resource state.

```
<link href="http://...:8080/users/232/customers"  
      rel="customers"/>
```

<http://martinfowler.com/articles/richardsonMaturityModel.html>

<http://roy.gbiv.com/untangled/2008/rest-apis-must-be-hypertext-driven>

```
import static ...RequestMethod.* ;
import static ...MediaType.* ;

@RestController
@RequestMapping(value = "/users/" )
class UserController {

    @Inject ResourceAssembler<User, Resource<User>> userResourceAssembler;

    @RequestMapping(method = DELETE, value = "/{user}")
    Resource<User> delete(@PathVariable Long user) {
        return userResourceAssembler.toResource(crmService.removeUser(user));
    }

    @RequestMapping(method = DELETE, value = "/{user}")
    Resource<User> read(@PathVariable Long user) {
        return userResourceAssembler.toResource(crmService.findById(user));
    }

    ..
}
```

```
@Component
class UserResourceAssembler implements ResourceAssembler<User, Resource<User>> {

    public Resource<User> toResource(User user) {
        Resource<User> userResource = new Resource<User>(user);

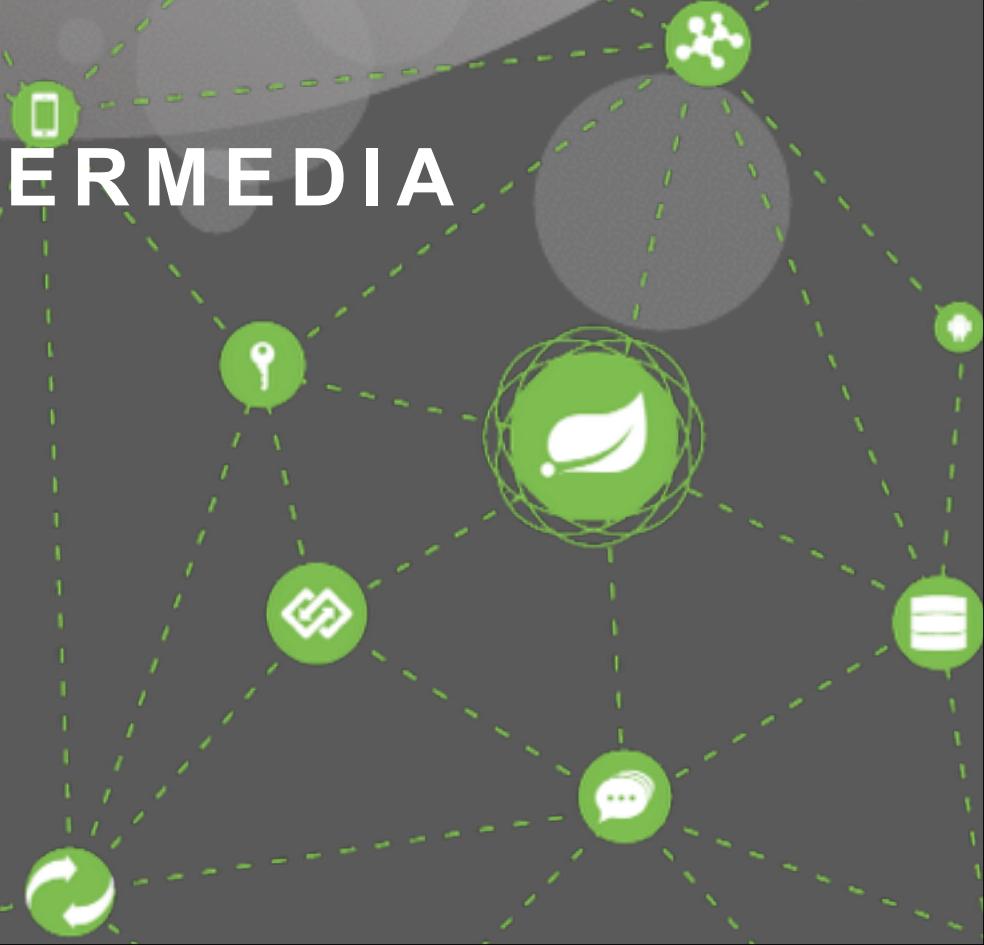
        Collection<Link> links = new ArrayList<Link>();

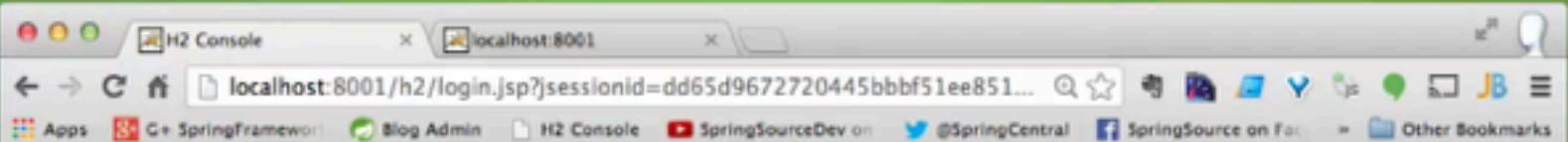
        Link toPhoto = linkTo( methodOn(UserProfilePhotoController.class)
            .loadUserProfilePhoto( user.getId() ))
            .withRel("photos");

        Link toSelf = linkTo( methodOn( UserController.class))
            .loadUser( user.getId() )
            .withSelfRel();

        userResource = new Resource<User>(user, Arrays.asList(toSelf, toPhoto));
    }
}
```

DEMO: EMBRACING HYPERMEDIA





English

Preferences Tools Help

Login

Saved Settings:

Generic H2 (Embedded)

Setting Name:

Generic H2 (Embedded)

Save

Remove

Driver Class:

org.h2.Driver

JDBC URL:

jdbc:h2:mem:crm

User Name:

sa

Password:

(empty field)

Connect

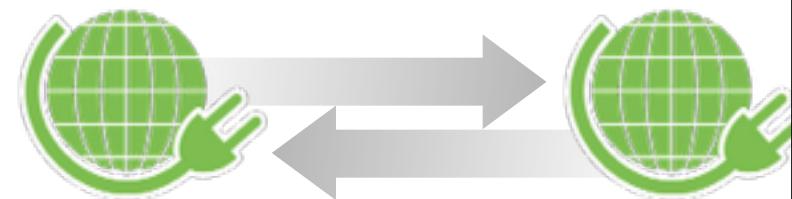
Test Connection

WebSockets delegating to implementations on app servers like GlassFish, Tomcat, and Jetty

Supports **Sock.js** server, superset of WS.

higher level STOMP supported on WS

supports JSR 356 (**javax.websocket.***) support.



Handling WebSocket Subscription

@Controller

```
class PortfolioController {
```

```
....
```

@MessageMapping("/app/trade")

```
void handleTradeOrder(Trade trade, Principal principal) {  
    trade.setUsername(principal.getName());  
    tradeService.executeTrade(trade);  
}
```

@SubscribeEvent("/app/positions")

```
List<PortfolioPosition> subscribeToPortfolios(Principal principal) {  
    String user = principal.getName();  
    Portfolio portfolio = portfolioService.findPortfolio(user);  
    return portfolio.getPositions();  
}
```

Handling Messages Over STOMP

@Controller

```
public class GreetingController {
```

```
// Mapping based on "destination" header
```

```
@MessageMapping("/greeting")
```

```
@SendTo("/topic/greetings")
```

```
public String greet(@MessageBody String greeting) {
```

```
    return "[" + getTimestamp() + "]: " + greeting;
```

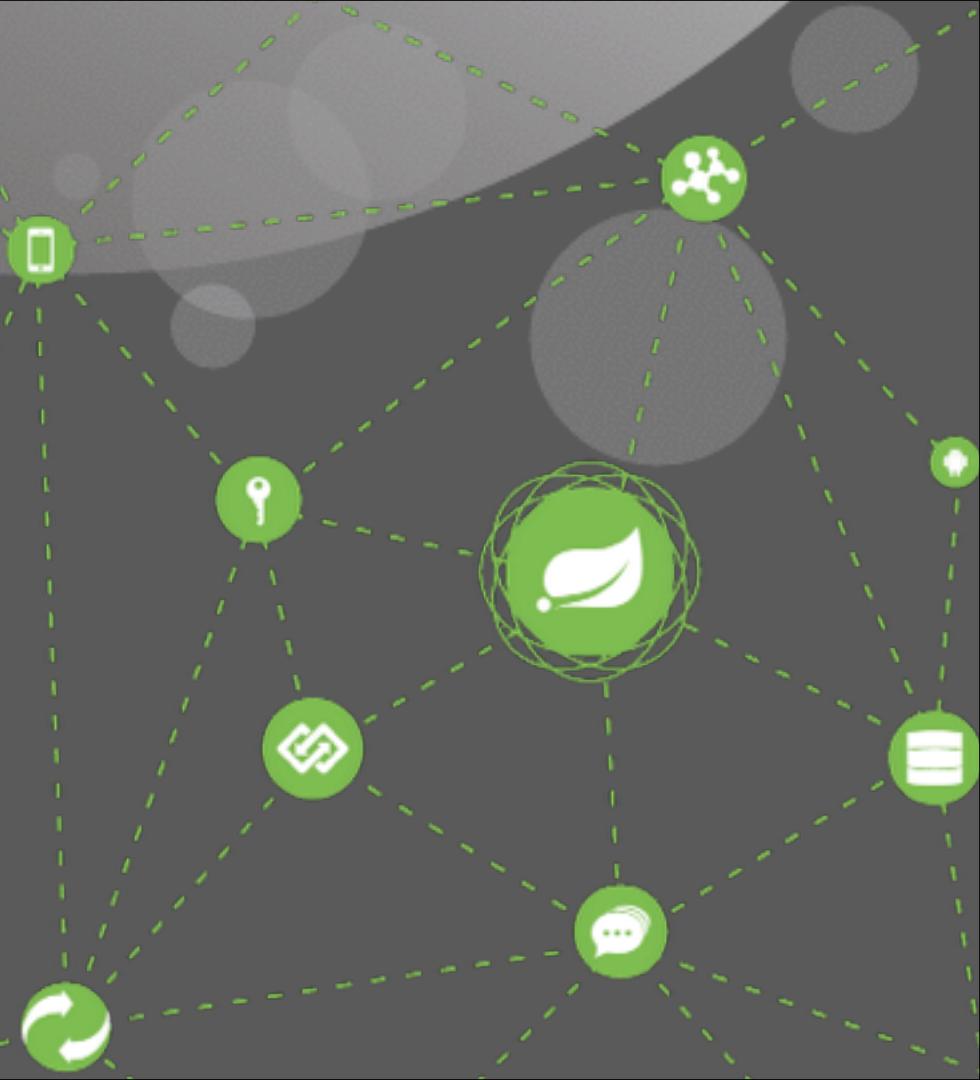
```
}
```

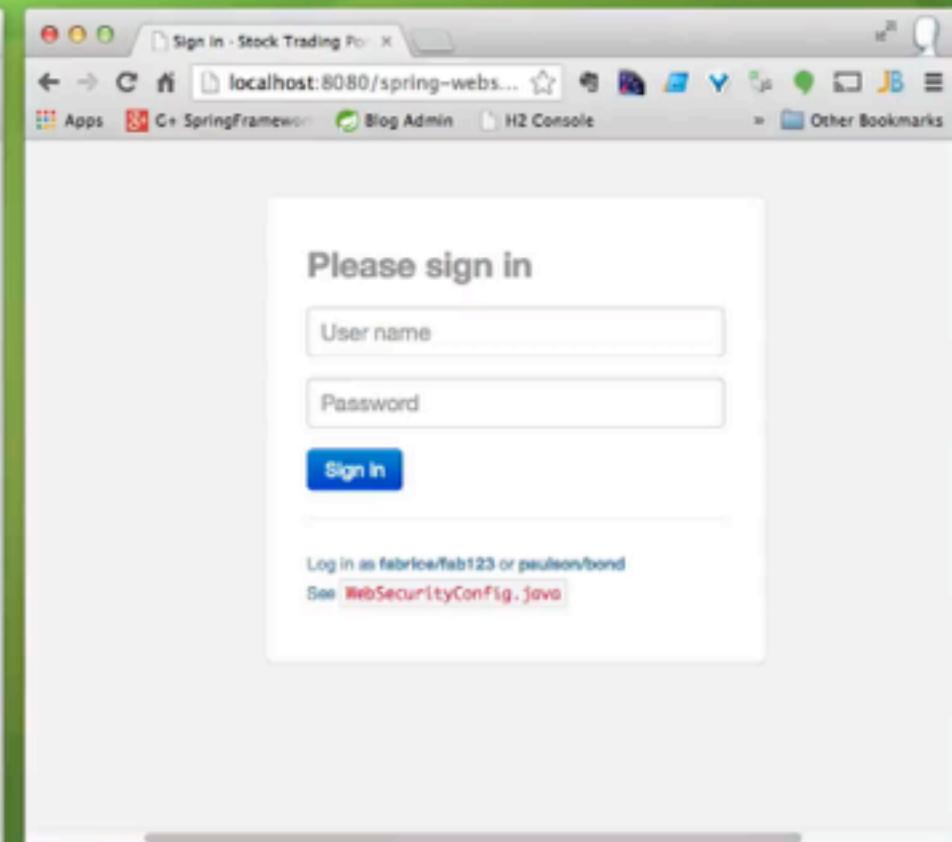
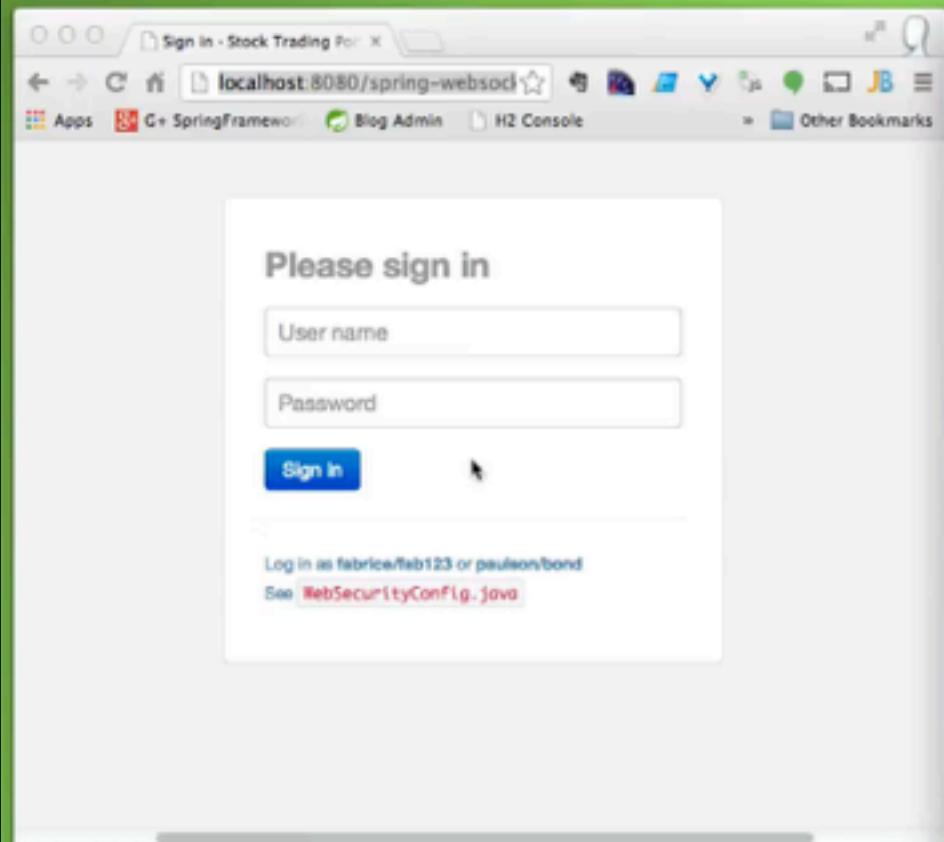
```
}
```

Configuring a STOMP WebSocket broker

```
@Configuration  
@EnableWebSocketMessageBroker  
public class Config implements WebSocketMessageBrokerConfigurer {  
  
    @Override  
    public void registerStompEndpoints(StompEndpointRegistry r) {  
        r.addEndpoint("/portfolio"); // WebSocket URL prefix  
    }  
  
    @Override  
    public void configureMessageBroker(MessageBrokerConfigurer c) {  
        c.enableSimpleBroker("/topic/"); // destination prefix  
    }  
}
```

DEMO: WEBSOCKETS





Spring Security is a modern security framework for a modern age

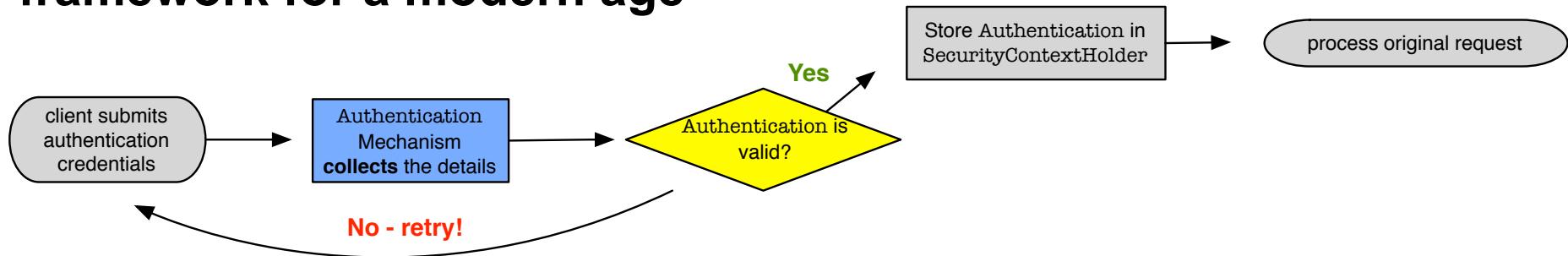
- OAuth
- sign in / out & “Remember Me” support
- Kerberos
- CAS
- pam
- ActiveDirectory / LDAP
- Encryption (TLS/SSL integration)
- x509 certificates
- OpenID



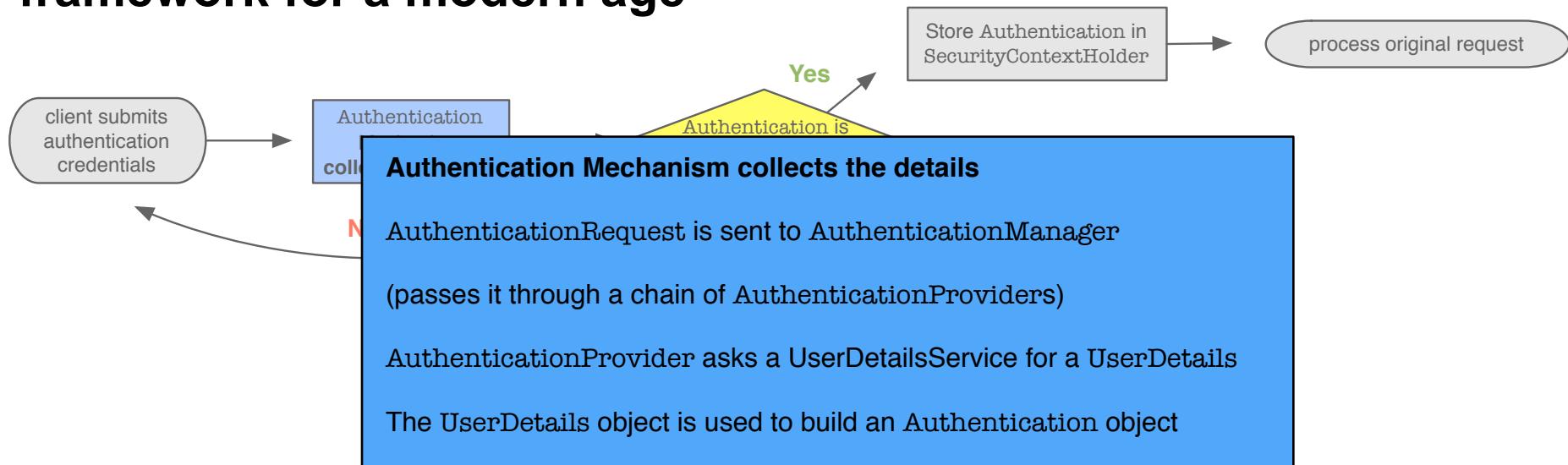
PARANOIA

Sometimes paranoia's just having all the facts. William S. Burroughs

Spring Security is a modern security framework for a modern age



Spring Security is a modern security framework for a modern age



Spring Security features a clean Java configuration API

```
@Configuration  
@EnableWebSecurity  
class SecurityConfig extends WebSecurityConfigurerAdapter {  
  
    @Autowired  
    public void configureGlobal(AuthenticationManagerBuilder auth)  
        throws Exception {  
        auth  
            .inMemoryAuthentication()  
            .withUser("user").password("password").roles("USER");  
    }  
  
    @Override  
    protected void configure(HttpSecurity http) throws Exception {  
        http.authorizeRequests()  
            .antMatchers("/").permitAll()  
            .antMatchers("/api/*").authenticated();  
        http.formLogin()  
            .loginPage("/login")  
            .passwordParameter("pw")  
            .usernameParameter("user");  
        http.x509();  
    }  
}
```

* source: visual.ly/60-seconds-social-media

A Connected World in 00:60 seconds

 **700k**
messages sent

 **1090**
visitors

 **2000**
checkins

 **175k**
tweets

 **7610**
searches

 **2MM**
videos viewed

 **3125**
photos uploaded

 StumbleUpon
7630
messages sent

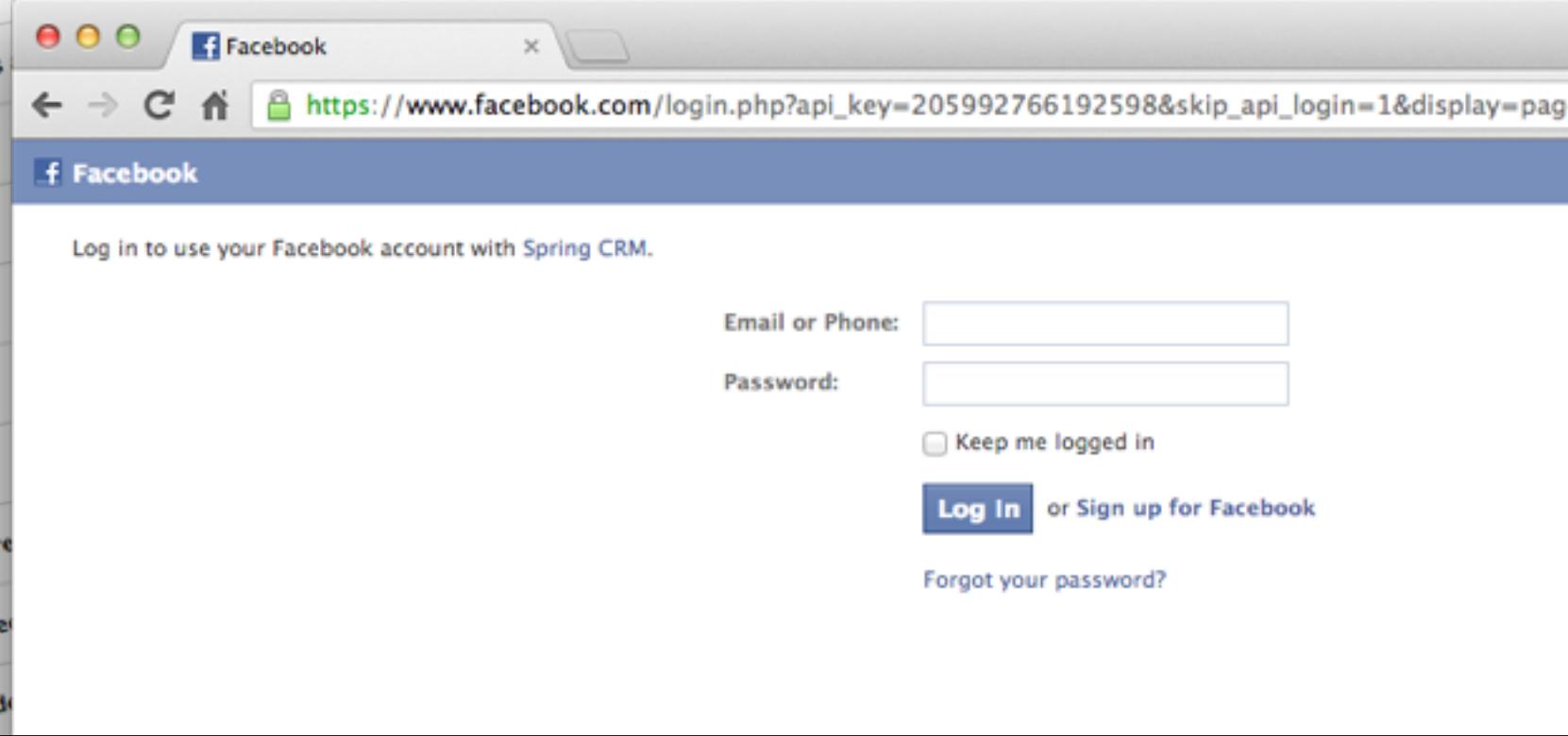
 www.raileurope.com	Friends	Edit 
 Stampit	Friends	Edit 
 Zillow Homes and Digs	Friends	Edit 
 Sidewalk	Friends	Edit 
 Lyft	Friends	Edit 
 Sincerely	Friends	Edit 
 Udacity	Only Me	Edit 
 Wrapp	Only Me	Edit 
 Picture Perfect Contest	Custom	Edit 
 Vimeo	Friends and Networks	Edit 
 SlideShare	Friends and Networks	Edit 
	Friends and Networks	Edit 

 www.raileurope.com Stampit Zillow Homes Sidewalk Lyft Sincerely Udacity Wrapp Picture Vimeo Slide

Friends

Friends

Friends

Edit Edit Edit Edit 

Facebook

https://www.facebook.com/login.php?api_key=205992766192598&skip_api_login=1&display=page

f Facebook

Log in to use your Facebook account with Spring CRM.

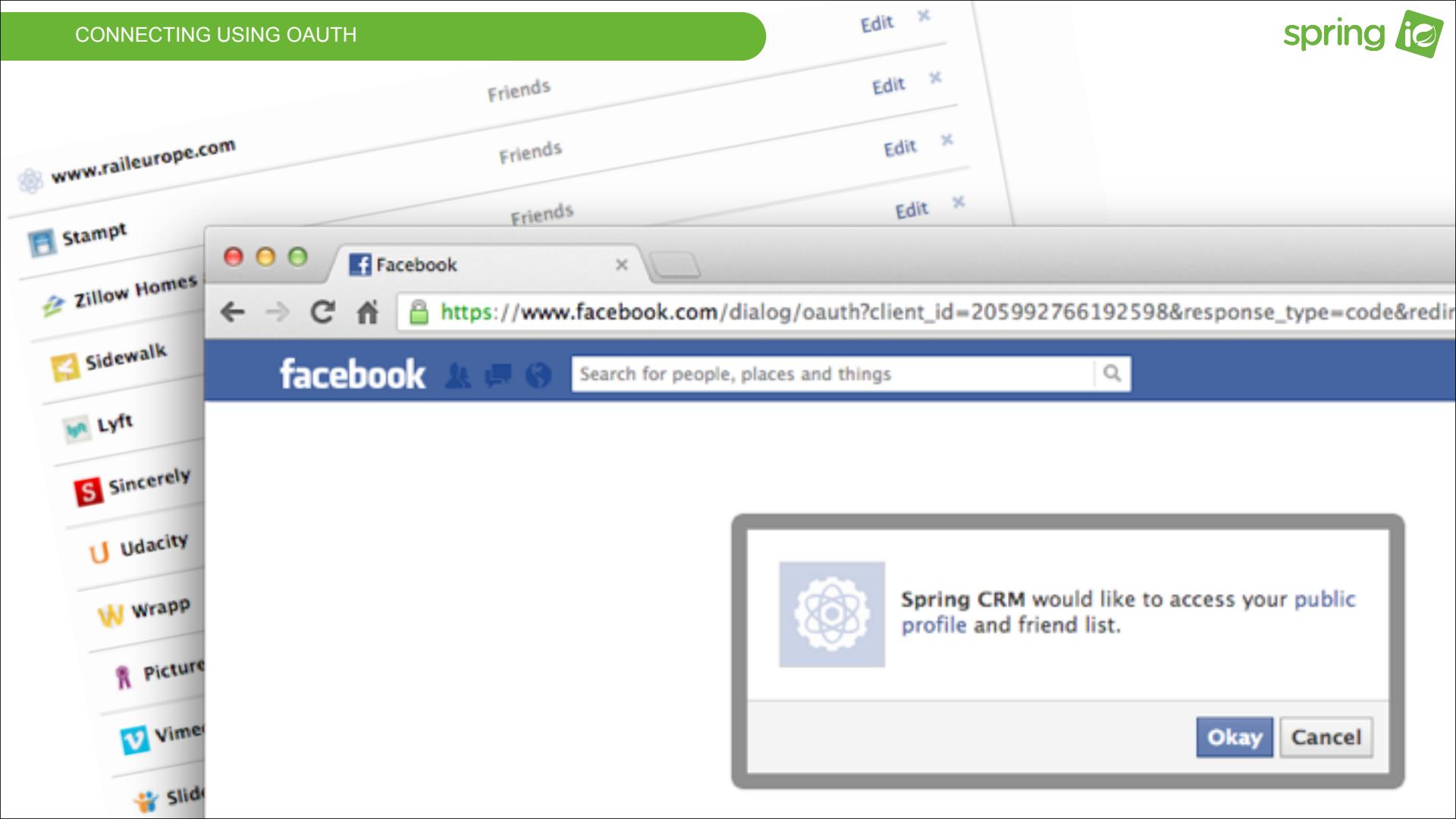
Email or Phone:

Password:

Keep me logged in

Log In or Sign up for Facebook

Forgot your password?



A screenshot of a web browser window titled "Facebook". The URL in the address bar is https://www.facebook.com/dialog/oauth?client_id=205992766192598&response_type=code&redirect_uri=https://www.raileurope.com/auth/facebook. The browser's title bar also shows "Facebook". The main content area displays a Facebook-style dialog box with a blue header containing the Spring logo (an atom icon) and the text "Spring CRM would like to access your public profile and friend list." At the bottom right of the dialog are "Okay" and "Cancel" buttons.

The browser window is overlaid on a background of a whiteboard with various application logos and names listed vertically on the left side:

- www.raileurope.com
- Stampit
- Zillow Homes
- Sidewalk
- Lyft
- Sincerely
- Udacity
- Wrapp
- Picture
- Vimeo
- Slide

At the top of the whiteboard, there are four "Friends" status indicators, each with an "Edit" link next to it.

The screenshot shows a web browser window with the title "Api Providers - SpringSource". The address bar indicates the page is "GitHub, Inc. [US] https://github.com/SpringSource/spring-social/wiki/Api-Providers". The main content is a table listing various third-party API providers, their authentication methods, and GitHub repositories.

Salesforce Force.com	OAuth 2	https://github.com/umut/spring-social-salesforce
SlideShare	API Key	
SmugMug	OAuth 1.0	
SoundCloud	OAuth 2	https://github.com/michaelfeuerle/spring-social-soundcloud
Springpad	OAuth 1.0	
StatusNet	OAuth 1.0a	
Stripe.com	OAuth 2	
StumbleUpon Badge API	No Auth/Unsecured API	
TripIt	OAuth 1.0	https://github.com/SpringSource/spring-social-tripit
Tumblr	OAuth 1.0a	https://github.com/sdouglass/spring-social-tumblr
Twitter	OAuth 1.0a	https://github.com/SpringSource/spring-social-twitter
Urban Airship	HTTP Basic	https://github.com/SpringSource/spring-mobile-urbanairship
Veevop	OAuth 2	
Viadeo	OAuth 2	https://github.com/Treydone/spring-social-viadeo
Vimeo	OAuth 1.0a	https://github.com/soldierkam/spring-social-vimeo
Vkontakte	OAuth 2	https://github.com/vkolodrevskiy/spring-social-vkontakte
Weibo	OAuth 1.0a	https://github.com/vergnes/spring-social-weibo
Windows Live/Microsoft (Hotmail, Messenger, Xbox)	OAuth 2	https://github.com/sachin-handlekar/spring-social-live
Xero	OAuth 1.0a	

Setting up Spring Social

```
@Configuration  
@EnableJdbcConnectionRepository  
@EnableFacebook(appId="${facebook.clientId}",  
                 appSecret="${facebook.clientSecret}")  
@EnableTwitter(appId="${twitter.consumerKey}",  
               appSecret="${twitter.consumerSecret}")  
@EnableLinkedIn(appId="${linkedin.consumerKey}",  
                 appSecret="${linkedin.consumerSecret}")  
public class SocialConfig {  
  
    // ...
```

Setting up Spring Social

@Controller

```
class FacebookFriendsController {  
  
    @Inject Facebook facebook;  
  
    @RequestMapping(value="/facebook/friends", method=GET)  
    public String showFeed(Model model) {  
        model.addAttribute("friends", facebook.friendOperations().getFriendProfiles());  
        return "facebook/friends";  
    }  
}
```

DEMO: CONNECTING APPLICATIONS TO USERS WITH SPRING SOCIAL

The background features a dark grey gradient with several semi-transparent light grey circles of varying sizes. Overlaid on this is a network graph composed of green circular nodes and dashed green lines. The nodes represent different social media or application entities. One node contains a white key icon, another a leaf, one has a speech bubble, and others have abstract symbols. Some nodes are interconnected by multiple lines, while others are isolated.

Spring Social Showcase

[Connections](#)

[Twitter](#)

[Facebook](#)

[LinkedIn](#)

Username

Password

[Sign In](#)

Some test user/password pairs you may use are:

- habuma/freebirds
- kdonald/melbourne
- rclarkson-atlanta

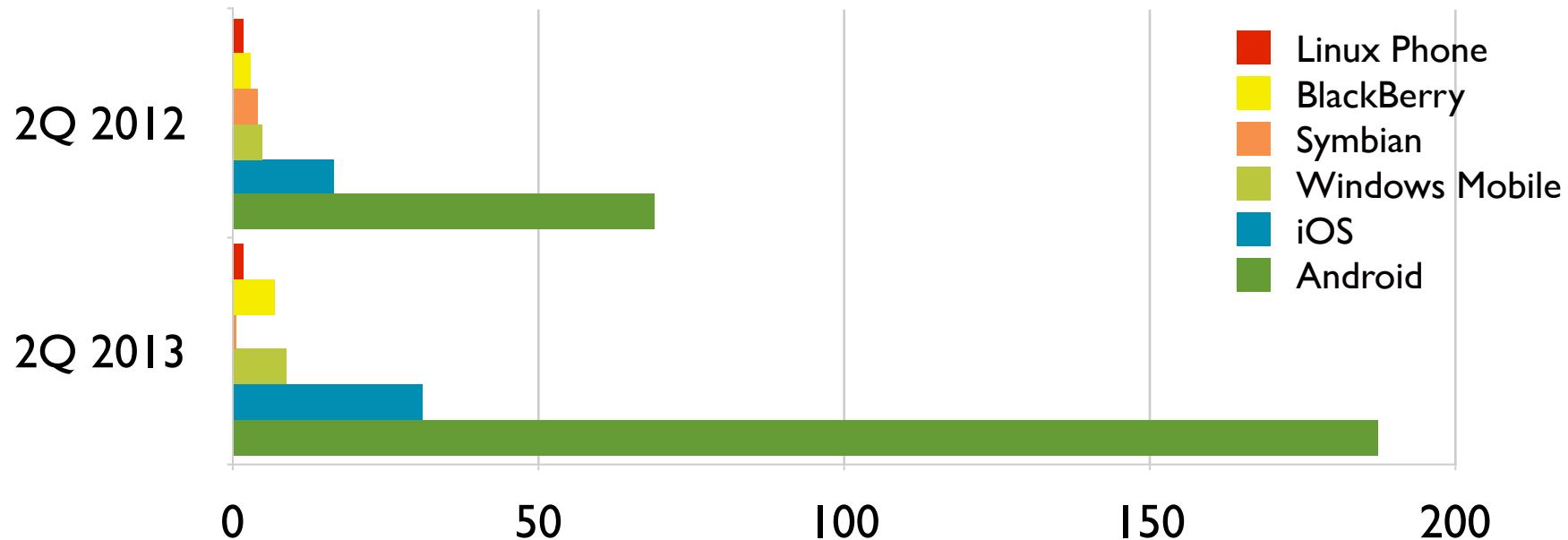
Or you can [signup](#) with a new account.

[Sign in with Twitter](#)

[Sign in with Facebook](#)

[Sign in with LinkedIn](#)

units (in millions) of units of each type of platform shipped



<http://www.idc.com/getdoc.jsp?containerId=prUS24257413>

Spring Android brings Spring core's **RestTemplate**.

Spring Social and **Spring HATEOAS** work as well.



DEMO: SMARTER MOBILE CLIENTS



Sign In

Please enter your username and password to log into the application.

User Name:

Password:

Submit

The preloaded (demonstration!) usernames and passwords are in the file /services/src/main/resources/crm-schema-*.sql, where * is either postgresql or h2. You might consult those if this is your first time signing in.



Spring works well in Java EE environments

Supports Java EE 7: Date/Time API, JTA 1.2, JMS 2.0, JPA 2.1, Bean Validation 1.1, the new concurrency API in JSR-236, Servlet 3.1, and WebSockets (JSR 356)

Even participated in a few JSRs: the websocket JSR (**356**) and the Batch JSR (**352**)

BUILD SUCCESSFUL

```
Total time: 9.366 secs
→ spring-music git:(master) ✘ cf push
Using manifest file manifest.yml

Creating spring-music... OK

Creating route j1-spring-music-466b.cfapps.io... OK
Binding j1-spring-music-466b.cfapps.io to spring-music... OK
Uploading spring-music... OK
Preparing to start spring-music... OK
----> Downloaded app package (19M)
----> Downloading OpenJDK 1.7.0_21 JRE from http://download.pivotal.io.s3.amazonaws.com/openjdk/lucid/x86_64/openjdk-1.7.0_21.tar.gz (10.8s)
      Expanding JRE to .java (1.0s)
----> Downloading Auto Reconfiguration 0.7.1 from http://download.pivotal.io.s3.amazonaws.com/auto-reconfiguration/auto-reconfiguration-0.7.1.jar (1.2s)
      Modifying /WEB-INF/web.xml for Auto Reconfiguration
----> Downloading Tomcat 7.0.42 from http://download.pivotal.io.s3.amazonaws.com/tomcat/tomcat-7.0.42.tar.gz (3.1s)
      Expanding Tomcat to .tomcat (0.1s)
      Downloading Buildpack Tomcat Support 1.1.1 from http://download.pivotal.io.s3.amazonaws.com/tomcat-buildpack-support/tomcat-buildpack-support-1.1.1.jar (0.0s)
----> Uploading droplet (55M)
Checking status of app 'spring-music'.....
  0 of 1 instances running (1 starting)
  0 of 1 instances running (1 starting)
  1 of 1 instances running (1 running)
Push successful! App 'spring-music' available at http://j1-spring-music-466b.cfapps.io
```





```
@Configuration  
@Profile("cloud")  
public class CloudConfig {  
  
    @Bean public Cloud cloud() throws CloudException {  
        CloudFactory cloudFactory = new CloudFactory();  
        return cloudFactory.getCloud();  
    }  
  
    @Bean public DataSource dataSource() {  
        return cloud().getSingletonServiceConnector(DataSource.class, null);  
    }  
}
```

YOU MIGHT CHECK OUT, FOR EXAMPLE, THE SPRING MUSIC APP



Spring Music

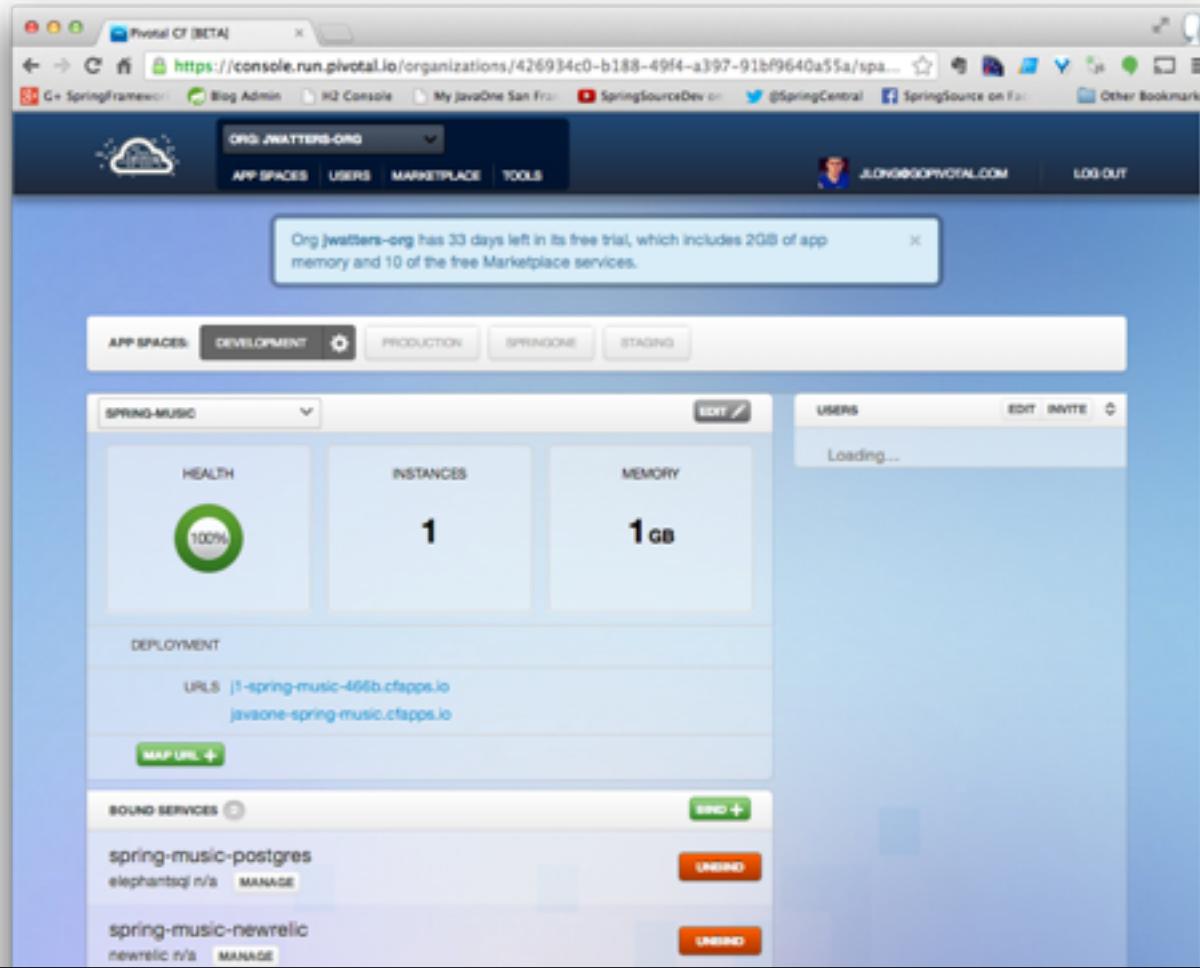
Profiles: cloud,postgres
Services: spring-music-postgres,spring-music-newrelic

Albums

[view as: list | sort by: title artist year genre | click on any field to edit]

IV Led Zeppelin 1971 Rock 0	Nevermind Nirvana 1991 Rock 0	What's Going On Marvin Gaye 1971 Rock 0	Are You Experienced? Jimi Hendrix Experience 1967 Rock 0
The Joshua Tree U2 1987 Rock 0	Abbey Road The Beatles 1969 Rock 0	Rumours Fleetwood Mac 1977 Rock 0	Sun Sessions Elvis Presley 1978 Rock 0
Exile on Main Street The Rolling Stones 1972 Rock 0	Born to Run Bruce Springsteen 1975 Rock 0	London Calling The Clash 1980 Rock 0	Hotel California The Eagles 1976 Rock 0
Led Zeppelin Led Zeppelin 1969 Rock 0	Pet Sounds The Beach Boys 1966 Rock 0	Synchronicity Police 1983 Rock 0	Achtung Baby U2 1991 Rock 0





The screenshot shows the Pivotal CF (Beta) web interface. At the top, it displays the URL <https://console.run.pivotal.io/organizations/426934c0-b188-49f4-a397-91bf9640a55a/spaces>. The dashboard for the organization "jwatters-org" is shown. A message box indicates that the organization has 33 days left in its free trial, which includes 20GB of app memory and 10 of the free Marketplace services. Below this, the application "SPRING-MUSIC" is listed. The application summary shows:

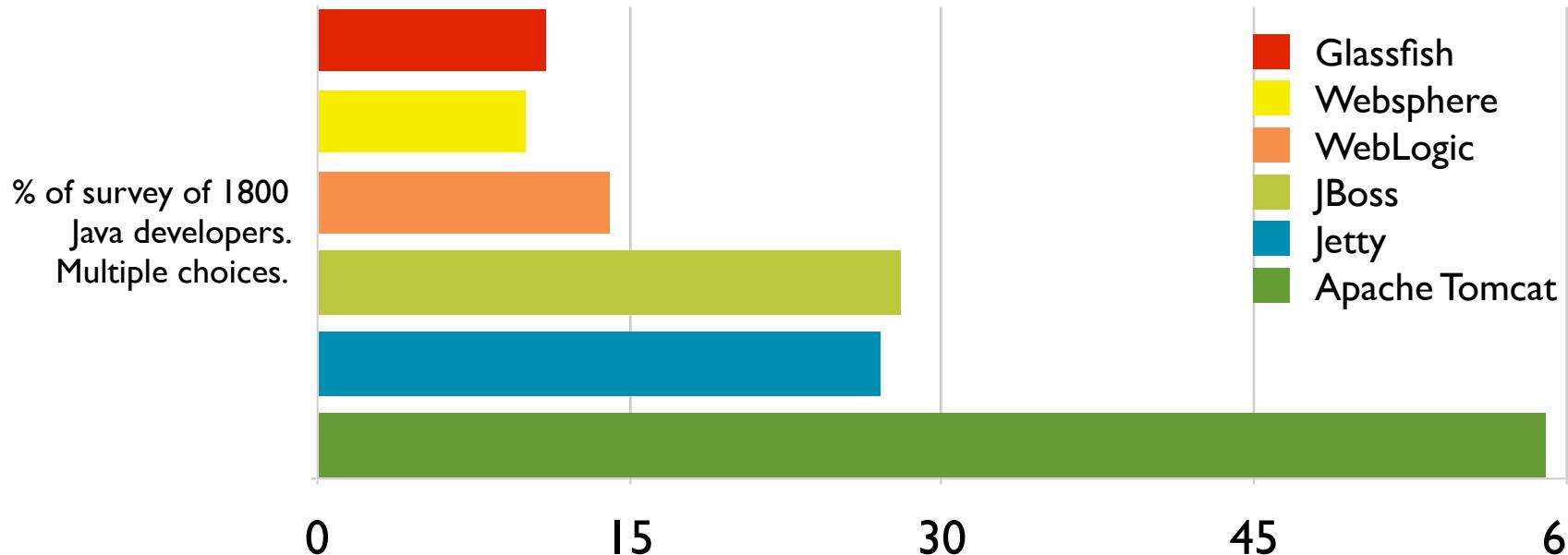
- HEALTH: 100% (green circle)
- INSTANCES: 1
- MEMORY: 1 GB

The deployment section shows URLs: <https://spring-music-466b.cfapps.io> and <https://javaone-spring-music.cfapps.io>. There is a "MAP URL" button. The "BOUNDED SERVICES" section lists "spring-music-postgres" and "elephantsql n/a" with a "MANAGE" button, and "spring-music-newrelic" and "newrelic n/a" with a "MANAGE" button. A "BIND +" button is also present.



The screenshot shows the New Relic Web transactions dashboard for the application "spring-music". The left sidebar includes links for Applications, Transactions, Mobile, Servers, Dashboards, Plugins (NEW), and Tools. The main area displays the "Web transactions" section with tabs for Overview, Map, and Web transactions. Under "Web transactions", the "App server" tab is selected, showing a breakdown of time consumption by endpoint: /appServlet (51.1%), /AlbumController/albums (35.9%), and /InfoController/info (13%). The "App performance" tab is active, showing an "App server breakdown" chart from 18:00 to 20:30. The chart highlights a spike for the "/AlbumController/albums" endpoint at approximately 19:47, with a duration of 67.3 ms. Below the chart is a legend for transaction types: Other (light blue), other (orange), RESULTSET (dark blue), album - SELECT (green), PoolingDataSource.getConnection() (brown), and AlbumController.albums (purple). The "Throughput" section shows an average of 0.0278 rpm.

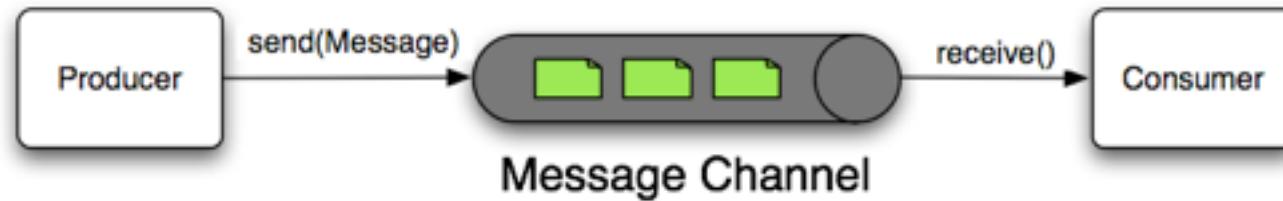


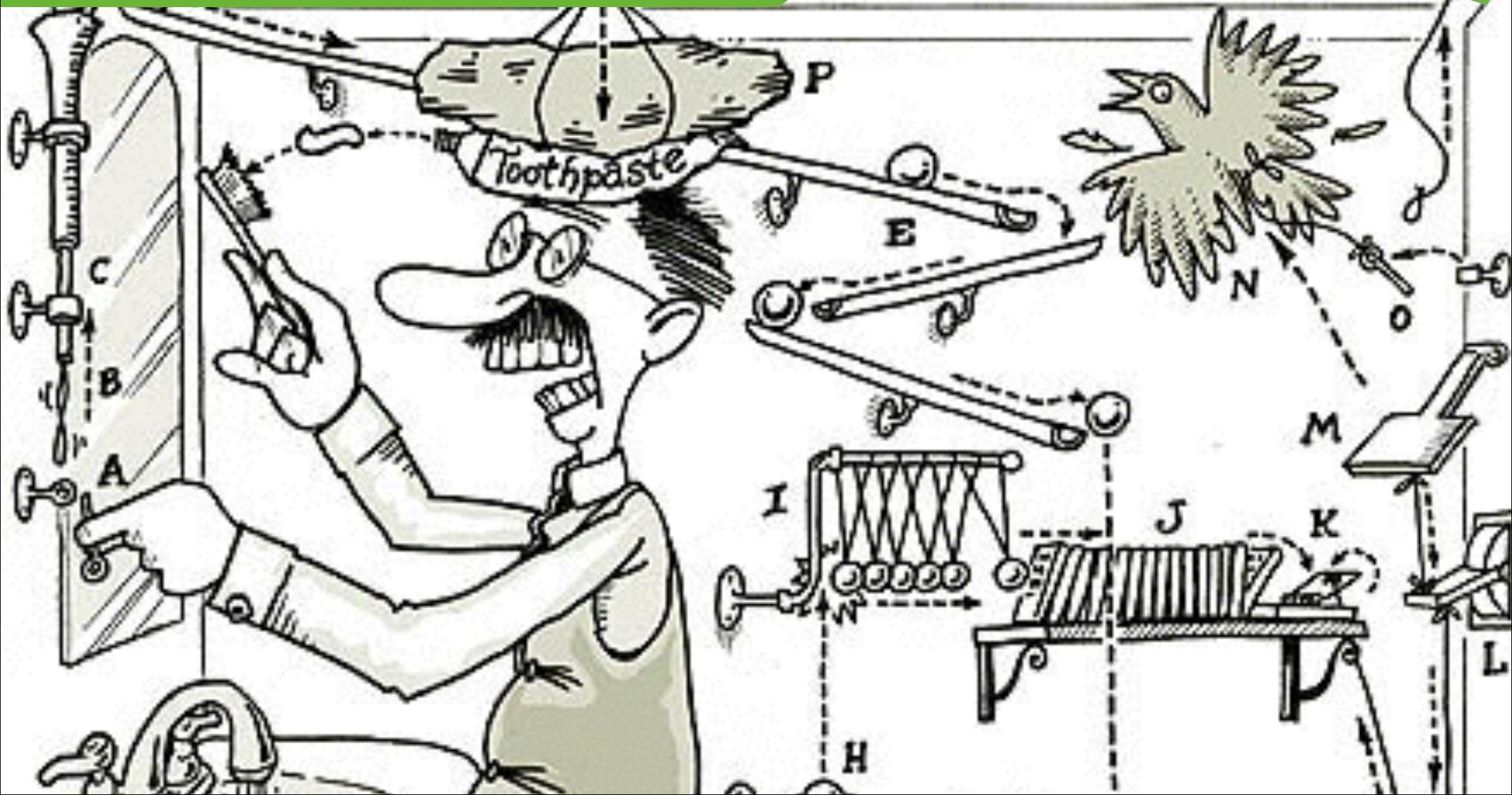


Source: ZeroTurnaround Developer Productivity Report 2012

Integration

- hard to get right
- patterned after “EIP book” patterns
- supports numerous adapters and gateways for external systems





```
<?xml version="1.0" encoding="UTF-8"?>
<beans ...>

    <int:channel id="jobsToLaunch" />
    <int:channel id="filesReceived" />
    <int:channel id="jobExecutions" />

    <file:inbound-channel-adapter auto-create-directory="true" channel="filesReceived"
        directory="#{systemProperties['user.home']} /Desktop/in">
        <int:poller fixed-rate="1000"/>
    </file:inbound-channel-adapter>

    <int:transformer ref="fileBatchJobRequestTransformer"
        input-channel="filesReceived" output-channel="jobsToLaunch" />

    <int:service-activator input-channel="jobsToLaunch"
        ref="jobMessageHandler" output-channel="jobExecutions" />

    <stream:stdout-channel-adapter id="jobsPrinted" append-newline="true" channel="jobExecutions" />

</beans>
```

```
@Configuration
@Import(BatchConfiguration.class)
@ComponentScan
@ImportResource("/com/joshlong/spring/walkingtour/services/integration/context.xml")
class IntegrationConfiguration {

    @Bean
    JobLaunchingMessageHandler jobMessageHandler(
        JobLauncher jobLauncher) throws Exception {
        return new JobLaunchingMessageHandler(jobLauncher);
    }

}
```

Batch Processing...

- avoids idle computing resources
- allows the system to use different priorities for batch and interactive work
- supports numerous sinks and sources
- supports **JSR 352 - Batch Applications**

ComputerWeekly.com

Article from ComputerWeekly.com 4 May 2000

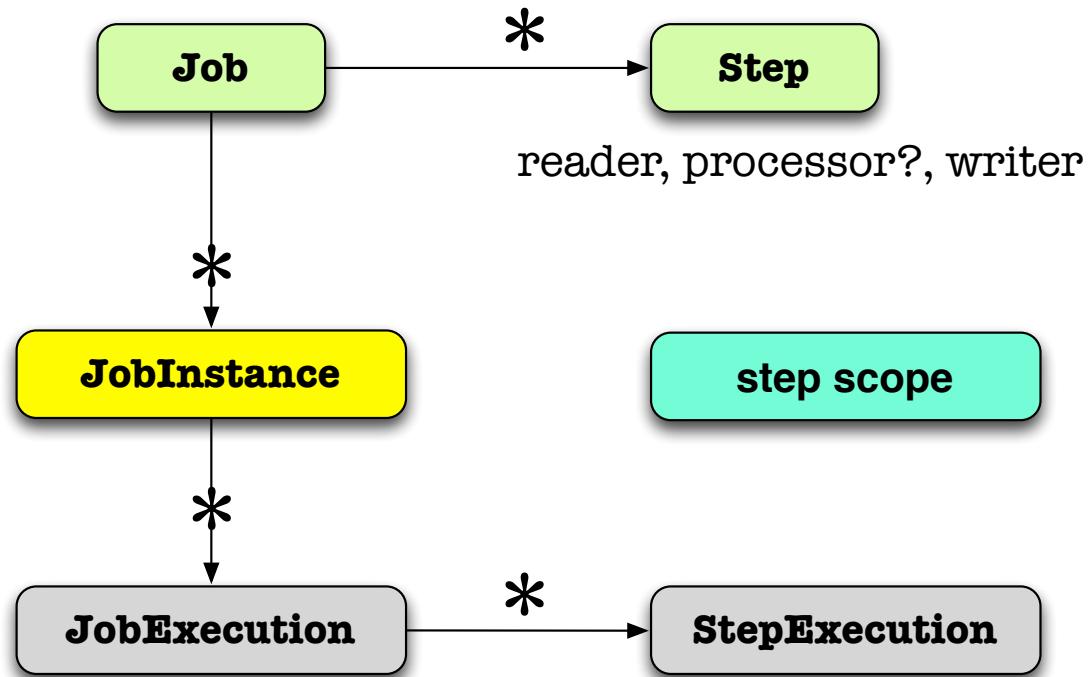
The Department of Social Security has suffered from two major IT failures over the past year. What were the failures, and what action is being taken? *Tony Collins* reports.

Failure 2: Overpayment by BACS

In an unrelated incident, about 112,000 claimants of income support received double their expected payments by automated credit transfer, directly to their bank accounts, when an EDS "Autobacs" batch file was accidentally processed twice.

An irony of the problem was that, having scrapped the Debt Accounting and Management System which was designed to collect overpayment debt, the Department, a few months later, went on to accidentally overpay 112,000 Income Support claimants.

The Department has repeatedly rejected Computer Weekly's requests for a detailed explanation of what went wrong. However staff say the problem began with a bug in a DSS batch programme, which was not intercepted by an EDS "Autobacs" system that collects and reconciles the batch programmes.



```
@Bean Job customerLoaderJob( JobBuilderFactory jobs, Step csvToRDBMS ) {  
    return jobs.get("customerLoaderJob")  
        .flow(csvToRDBMS).end().build();  
}
```

```
@Bean Step csvToRDBMS(  
    StepBuilderFactory stepBuilderFactory,  
    PlatformTransactionManager platformTransactionManager,  
    ItemReader<Customer> itemReader,  
    ItemProcessor<Customer, Customer> itemProcessor,  
    ItemWriter<Customer> itemWriter) {
```

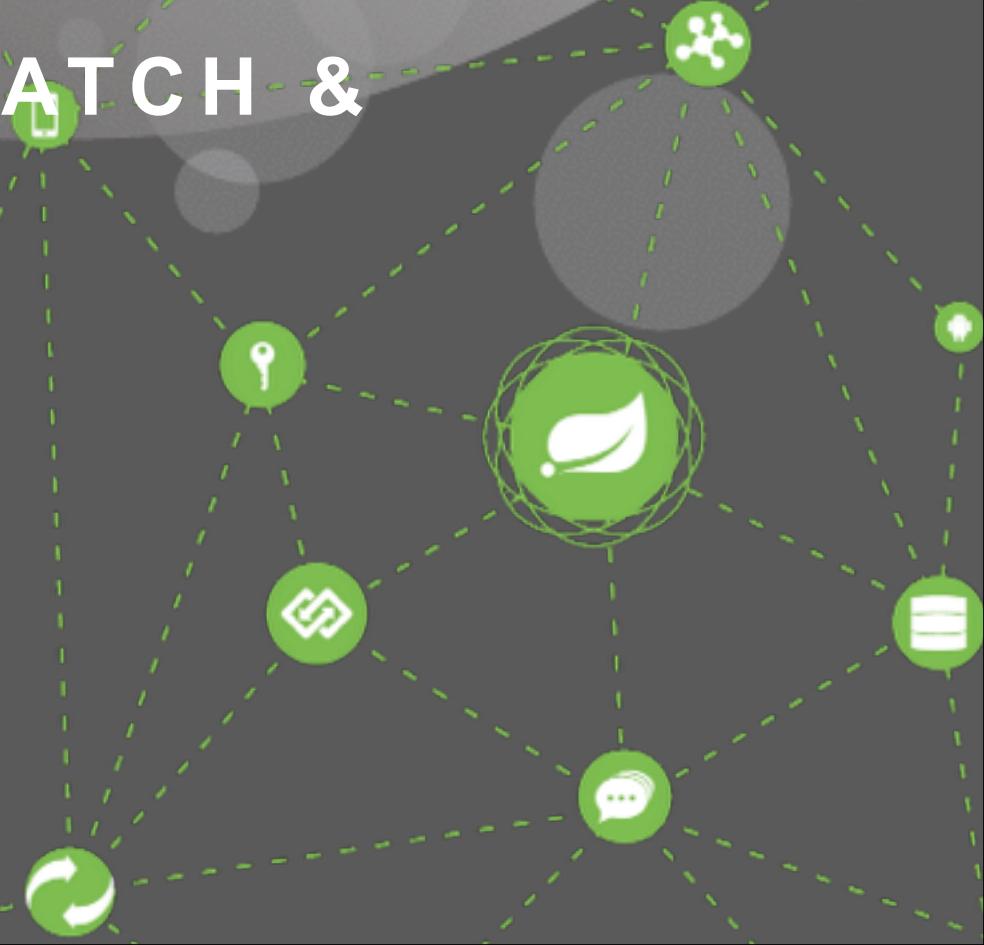
```
    StepBuilder builder = stepBuilderFactory.get("csvToRDBMS");
```

```
    return builder.<Customer, Customer>  
        .chunk(3)  
        .reader(itemReader)  
        .processor(itemProcessor)  
        .writer(itemWriter)  
        .transactionManager(platformTransactionManager)  
        .build();  
}
```

FlatFileItemReader

JdbcBatchItemWriter

DEMO: USING SPRING BATCH & INTEGRATION



Spring - services/src/main/java/com/joshlong/spring/walkingtour/se... w

H2 Console

192.168.2.14:8082/login.js...

Apps G+ SpringFramework Blog Admin H2 Console Other Bookmarks

English Preferences Tools Help

Login

Saved Settings: Generic H2 (Embedded)

Setting Name: Generic H2 (Embedded) Save Remove

Driver Class: org.h2.Driver

JDBC URL: jdbc:h2:tcp://localhost/-ibatch_orm

User Name: sa

Password:

Connect Test Connection

ESS

File View Action Dropbox Arrange Share

Back

FAVORITES

- AirDrop
- jlong
- Dropbox
- Macintosh HD
- Desktop

Applications

Documents

Downloads

Movies

Music

Pictures

Google Drive

Photos

Camera Uploads

Macintosh HD > Users > jlong > Desktop > in

0 items, 67.95 GB available

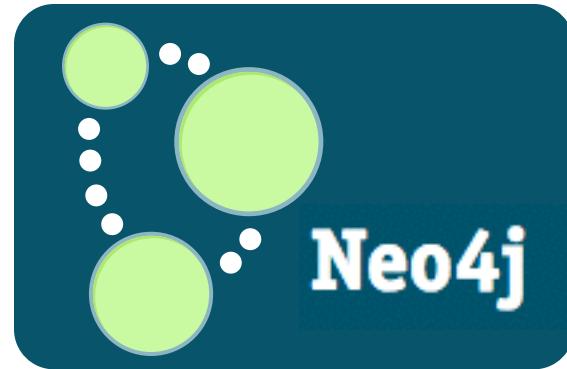
Line	Name
1	Hoellter,Juergen
2	Fisher,Mark
3	Gierke,Oliver
4	Winch,Rob
5	Pollack,Mark
6	Colyer,Adrian
7	Syer,Dave
8	Laddad,Ramnivas
9	Long,Josh
10	Beams,Chris
11	

Line: 1 | CSV

Tab Size: 2

Writable Smart Insert 11 : 6

Spring supports *all* the SQLs:
NoSQL, **NOSQL**, and **SQL**



```
import org.springframework.data.repository.PagingAndSortingRepository;
import org.springframework.data.repository.query.Param;
import org.springframework.data.rest.repository.annotation.RestResource;

import java.util.List;

@RestResource (path = "users", rel = "users")
public interface UserRepository extends PagingAndSortingRepository<User, Long> {

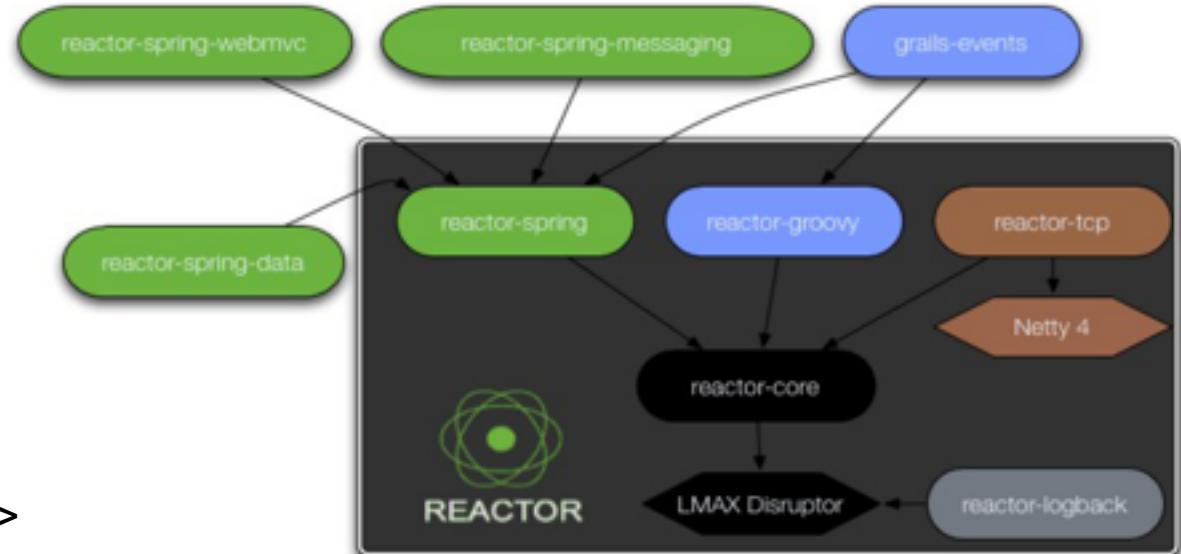
    // select * from users where username = ?
    User findByUsername(@Param("username") String username);

    // select * from users where firstName =? or lastName = ? or username = ?
    List<User> findUsersByFirstNameOrLastNameOrUsername(
        @Param("firstName") String firstName,
        @Param("lastName") String lastName,
        @Param("username") String username);

}
```

The future is Reactive

Reactor: foundational library. Drives drivers, servers, data integration libraries, domain integration libraries, and evented architectures



Used in: Spring Integration, websocket => STOMP support, Grails, etc.

without Spring

```
Environment env = new Environment();
```

```
Reactor reactor = Reactors.reactor()  
    .env(env)  
    .dispatcher(RING_BUFFER)  
    .get();
```

```
reactor.on$("topic"), (Event<String> ev) -> {  
    System.out.println("Hello " + ev.getData());  
});
```

```
reactor.notify("topic", Event.wrap("John Doe"));
```

ThreadPoolExecutorDispatcher
BlockingQueueDispatcher
...
RingBufferDispatcher

with Spring

```
@EnableReactor  
 @Configuration  
 @ComponentScan  
 class ReactorConfiguration {  
  
     @Bean  
     Reactor reactor(Environment env) {  
         return Reactors.reactor().env(env)  
             .dispatcher(  
                 Environment.RING_BUFFER)  
             .get();  
     }  
 }
```

```
@Component  
 class EvenConsumer {  
  
     @Selector("test.topic")  
     public void onTestTopic(String s) {  
         log.info("onTestTopic: {}", s);  
     }  
 }  
  
 @Service  
 class EventPublisher {  
  
     @Autowired  
     Reactor reactor;  
  
     public void fire(String msg) {  
         reactor.notify(  
             "test.topic", Event.wrap( msg ));  
     }  
 }
```

Streams like Netflix RxJava Observable, JDK 8 Stream

```
Stream<String> str = ...;  
  
str.map(String::toUpperCase)  
    .filter(new Predicate<String>() {  
        public boolean test(String s) { ... }  
    })  
    .consume(s → log.info("consumed string {}", s));
```

Did I mention it's *fast*?



Jon Brisbin @j_brisbin · Feb 20

@ProjectReactor #SpringIntegration #RingBuffer msg batcher bumps #Kryo
handler throughput from 900k to 3.9M msgs/sec #uberfastdata

[Collapse](#)

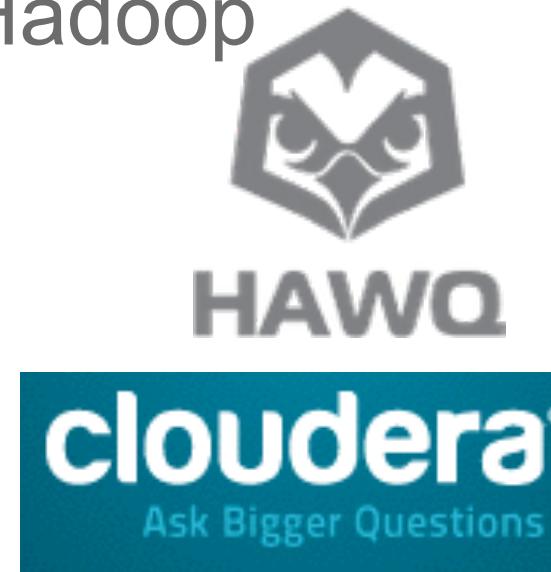
Reply Retweet Favorite More

<https://github.com/reactor/reactor-si-quickstart>

SPRING WORKS WELL WITH



Surviving the Big Data Wild-West with Spring for Hadoop



Getting Data into the System is Hard

@metamarkets founder **Michael E. Driscoll**:



Michael E. Driscoll @medriscoll 21 Sep

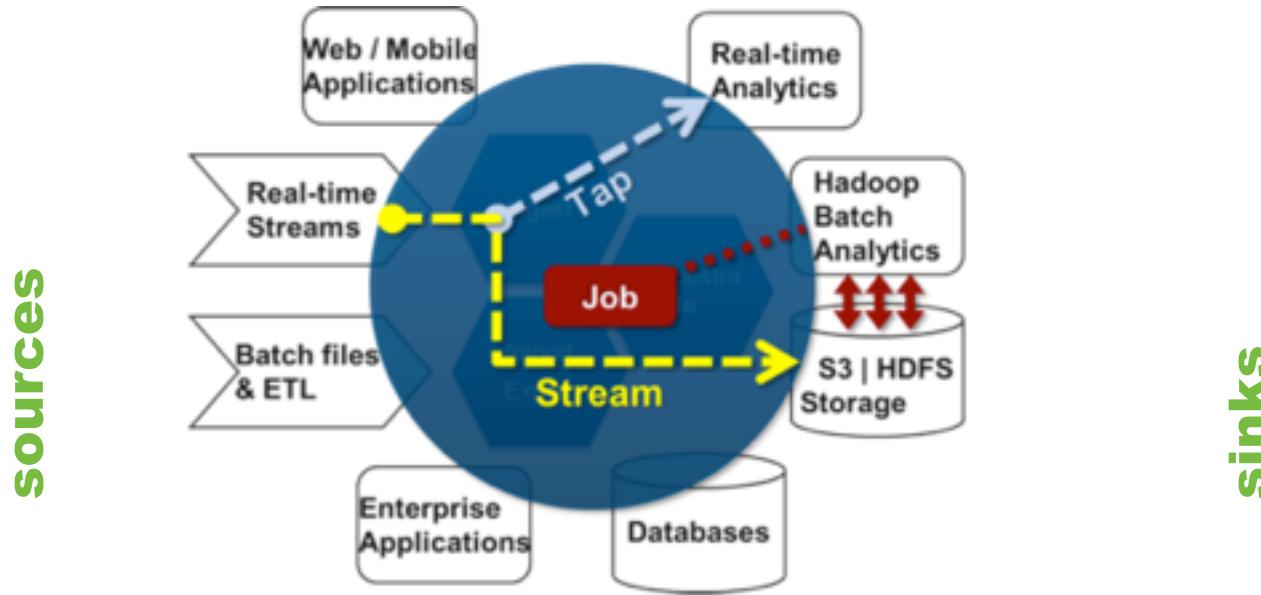
Stream processing is the most innovative nook of (big) data today:
LinkedIn's Samza, Twitter's Summingbird, Yahoo's S4, Google's
Millwheel.

[Collapse](#) [Reply](#) [Retweeted](#) [Favorite](#) [More](#)

10 RETWEETS	16 FAVORITES
-------------	--------------



Introducing Spring XD



```
xd:> stream create --definition "time | log" --name ticktock
```

DEMO: SPRING FOR HADOOP AND SPRING XD



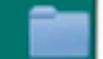
```
~/sqlf          ~/xd-single  
demo-xd/springxd_files/cmd/  
job undeploy batchRetailArc  
Un-deployed Job 'batchRetailArc'  
stream destroy --name batchRetailArchive  
Destroyed stream 'batchRetailArchive'  
job deploy batchRetailArchitect  
Deployed job 'batchRetailArchitect'  
stream create --name batchRetailArchive  
Created new stream 'batchRetailArchive'  
job undeploy batchAnalytic  
Un-deployed Job 'batchAnalytic'  
stream destroy --name batchAnalytic  
Destroyed stream 'batchAnalytic'  
job deploy batchAnalytic  
Deployed job 'batchAnalytic'  
stream create --name batchAnalytic  
Created new stream 'batchAnalytic'  
Script required 0.506 seconds  
xd:>[
```



gpadmin's Home



misc



Pivotal_D



README



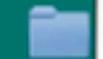
start_piv_hd.sh



Trash



misc



Pivotal_D



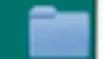
README



start_piv_hd.sh



misc



Pivotal_D



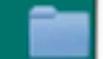
README



start_piv_hd.sh



misc



Pivotal_D



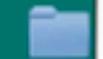
README



start_piv_hd.sh



misc



Pivotal_D



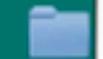
README



start_piv_hd.sh



misc



Pivotal_D



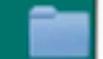
README



start_piv_hd.sh



misc



Pivotal_D



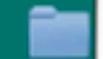
README



start_piv_hd.sh



misc



Pivotal_D



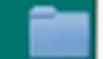
README



start_piv_hd.sh



misc



Pivotal_D



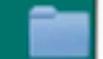
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



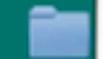
README



start_piv_hd.sh



misc



Pivotal_D



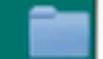
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



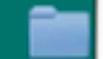
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



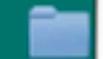
README



start_piv_hd.sh



misc



Pivotal_D



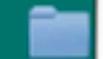
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



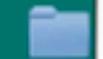
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



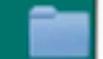
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



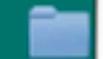
README



start_piv_hd.sh



misc



Pivotal_D



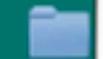
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



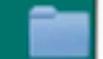
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



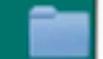
README



start_piv_hd.sh



misc



Pivotal_D



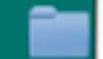
README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



README



start_piv_hd.sh



misc



Pivotal_D



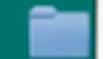
README



start_piv_hd.sh



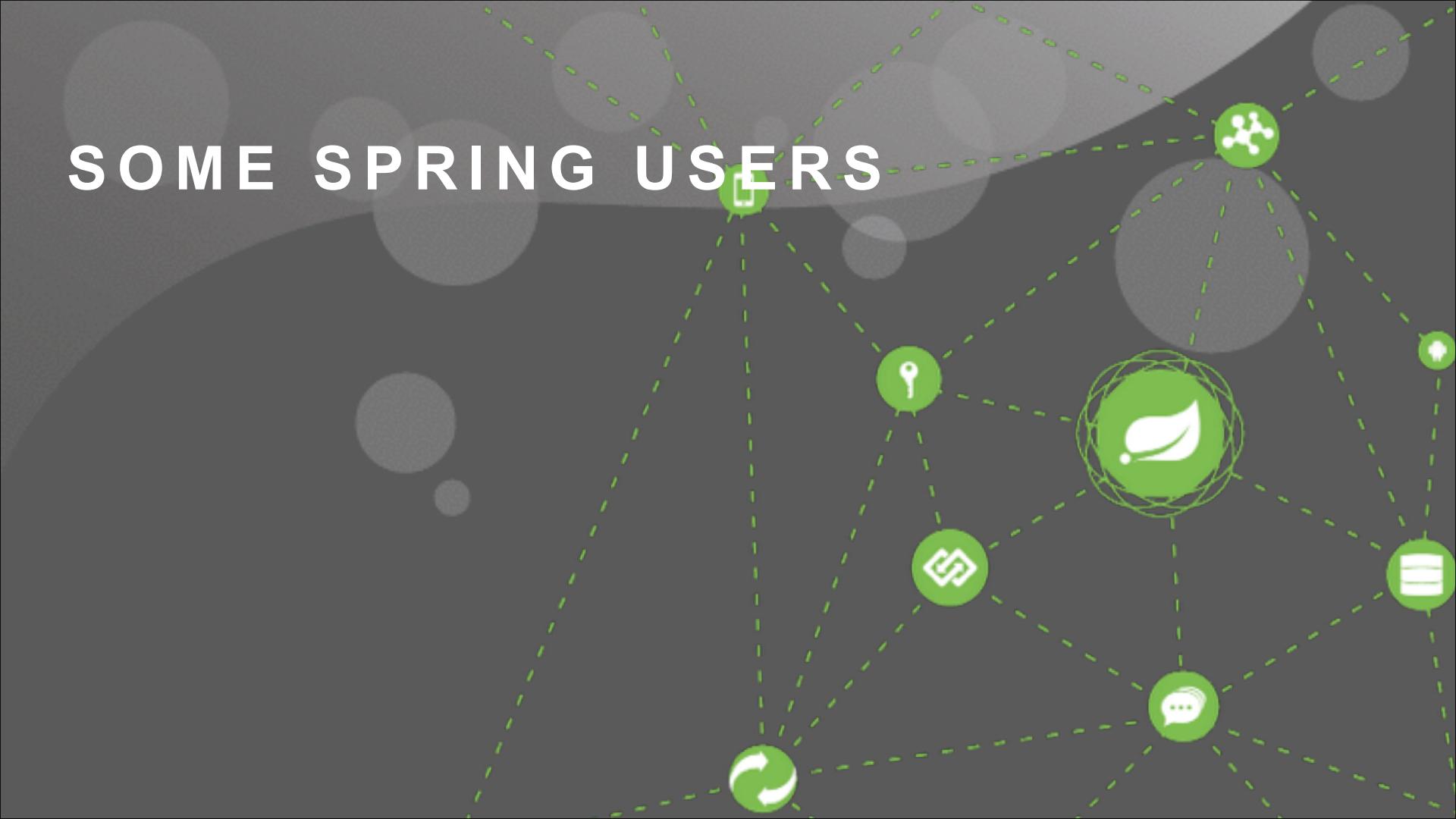
misc



Pivotal_D



SOME SPRING USERS

The background features a dark grey gradient with several semi-transparent grey circles of varying sizes. Overlaid on this are several green circular nodes connected by dashed green lines, forming a network. The nodes contain icons representing different Spring framework modules: a mobile phone, a key, a leaf inside a circular border, a diamond shape, a speech bubble, a document, and a gear.



“ on 11/11/2012 (the “Double Sticks” promotion day), Tmall and Taobao witnessed 147 million user visits, purchases of 30 million people and nearly 100 million paid orders. At 0:00, more than 10 million users were concurrently online

china



InfoQ

Development Architecture & Design Process & Practices Operation Infrastructure

858,214 Sep unique visitors

Mobile HTML5 JavaScript APM Big Data PHP Cloud Agile

You are here: [InfoQ Homepage](#) > [News](#) > How Alibaba Catered To \$3 Billion Sales In A Day

How Alibaba Catered To \$3 Billion Sales In A Day

by [Roopesh Shenoy](#) on Dec 26, 2012 | [3 Discuss](#)

Share

* source: <http://www.infoq.com/news/2012/12/interview-taobao-tmall>

<http://spring.io/blog/2013/03/04/spring-at-china-scale-alibaba-group-alipay-taobao-and-tmall/>

Spring and RabbitMQ: powering India's 1.2 B Person Biometric DB



- 1.2 billion residents
 - ~75% literacy
 - less than 3% pay incomes taxes
 - less than 20% banking
 - 800 million mobile, ~200-300 million migrant workers
- **Government spends about \$25-40 billion on direct attributes**
 - residents have no standard identity
 - most programs plagued with ghost and multiple identities causing leakage of 30-40%

- traded on **JASDAQ**
- largest Java consultancy in Japan
- Spring Batch
 - handles customer profile information and real-time ad matching system
- Spring is agile enabler
 - production control system
 - just-in-time production is really the heart of manufacturing
 - cost decreased 30%



Shigeru Urushibara
President and
Chief Executive
Officer



china

- **1.3 billion people in China**
 - Google is either mostly blocked or - from Hong Kong - very slow
- Baidu has its own Android operating system, cloud services (like GMail, Dogs, etc.)
- Built micro service architecture that builds on top of Spring
- Custom web frameworks using Spring XML namespaces as global integration API

<http://blog.gopivotal.com/products/have-you-seen-spring-lately>



@starbuxman @springcentral
josh.long@springsource.com
josh@joshlong.com
github.com/joshlong
slideshare.net/joshlong