

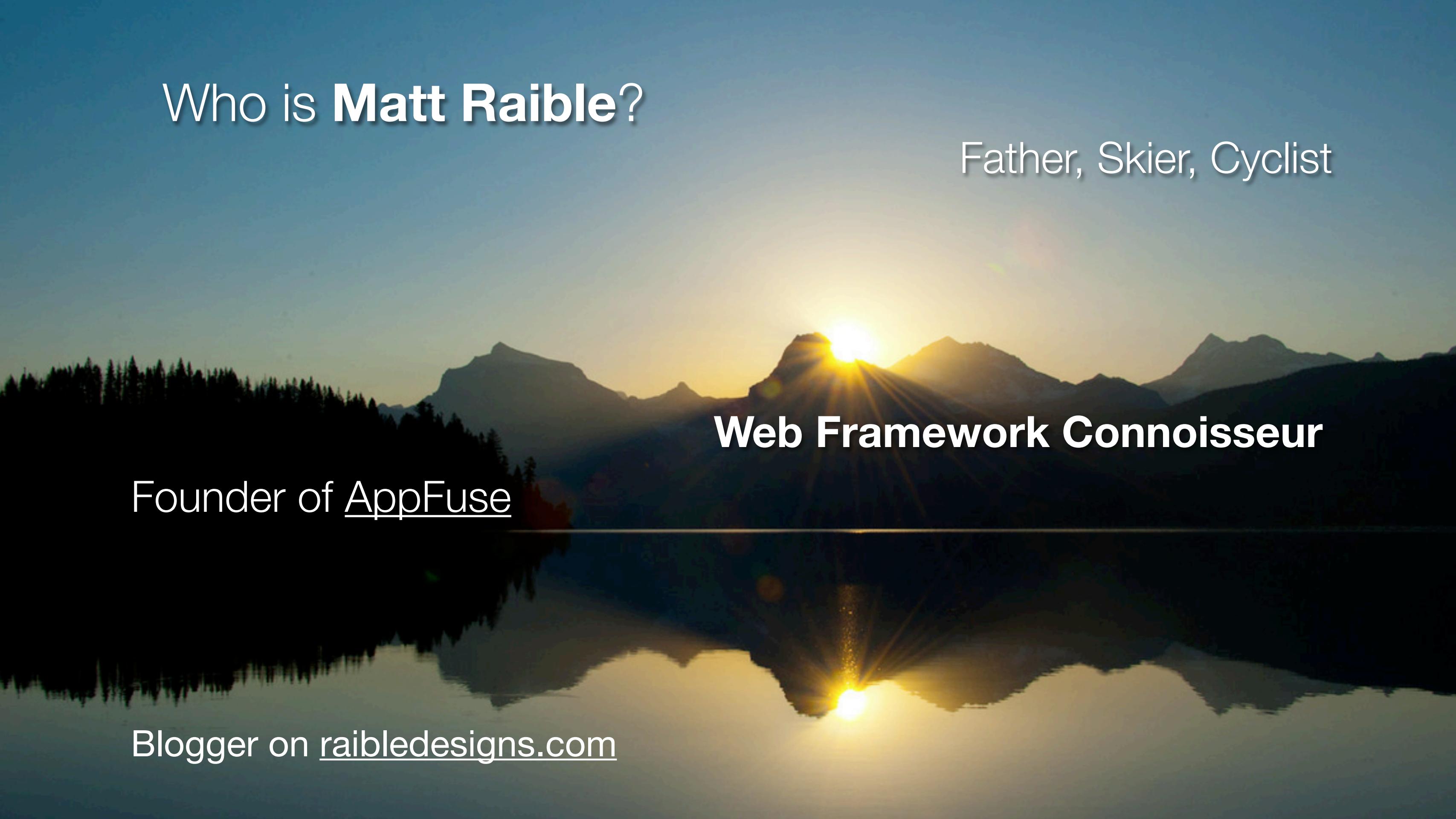
The Modern Java Web Developer

Matt Raible • <http://raibledesigns.com>



Photos by [Trish McGinity](#) and [Art Escobado](#)

 JavaOne™ 2013

The background of the slide features a serene landscape at sunset. In the foreground, a calm lake reflects the warm orange and yellow hues of the setting sun. Beyond the lake, a range of mountains is visible, their peaks silhouetted against the bright sky. The overall atmosphere is peaceful and inspiring.

Who is **Matt Raible**?

Father, Skier, Cyclist

Web Framework Connoisseur

Founder of AppFuse

Blogger on raibledesigns.com

How about YOU?

Have you developed a Struts 1 application? Used PHP?

Have you every written CSS from scratch?

Why do you hate JavaScript?

What's your favorite JavaScript framework?



by Art Escobedo

Topic Inspiration

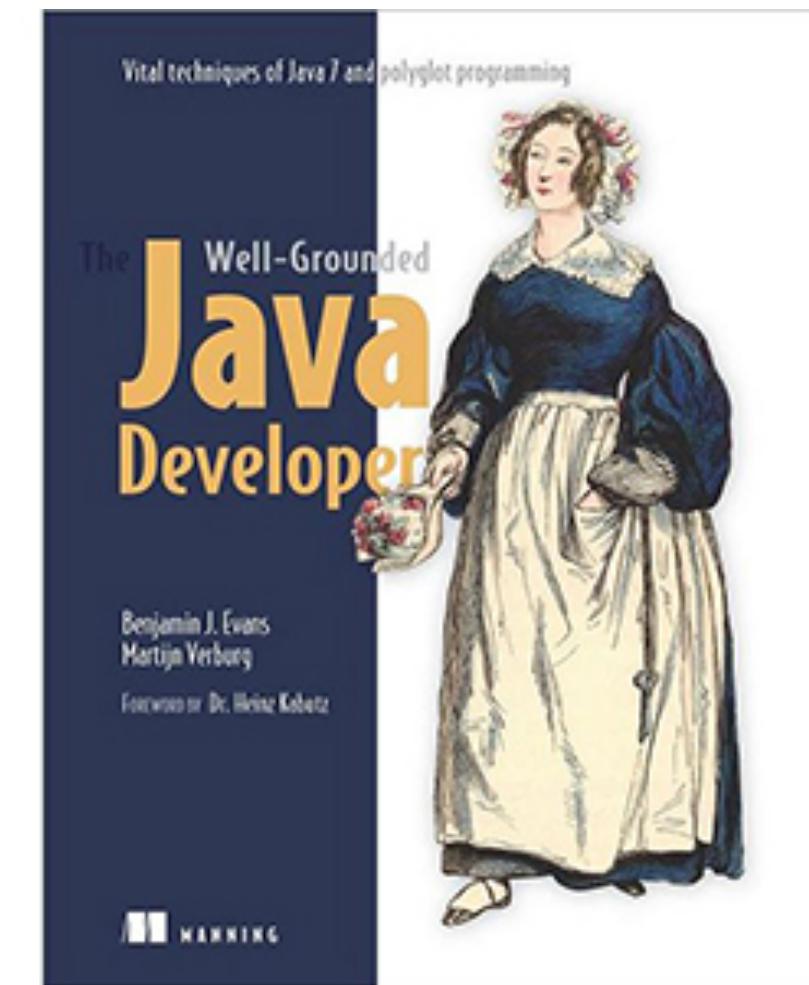
Inspired by Ben Evans's and Martijn Verburg's **The Well-Grounded Java Developer**

Developing with Java 7

Vital techniques

Polyglot programming on the JVM

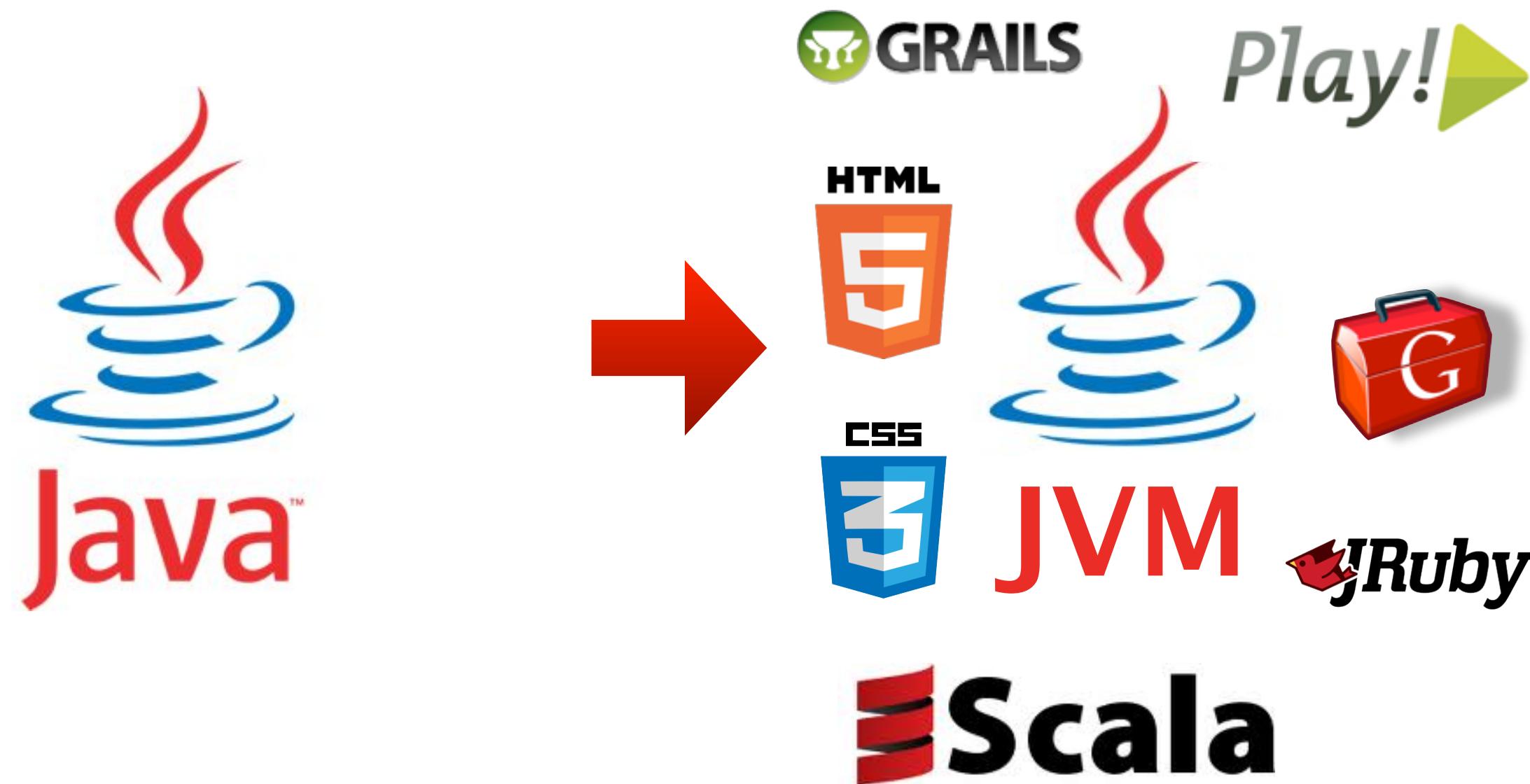
Crafting the polyglot project



Purpose



The Modern Java Web Developer



The Modern **JVM** Web Developer

Starts with Fast Hardware

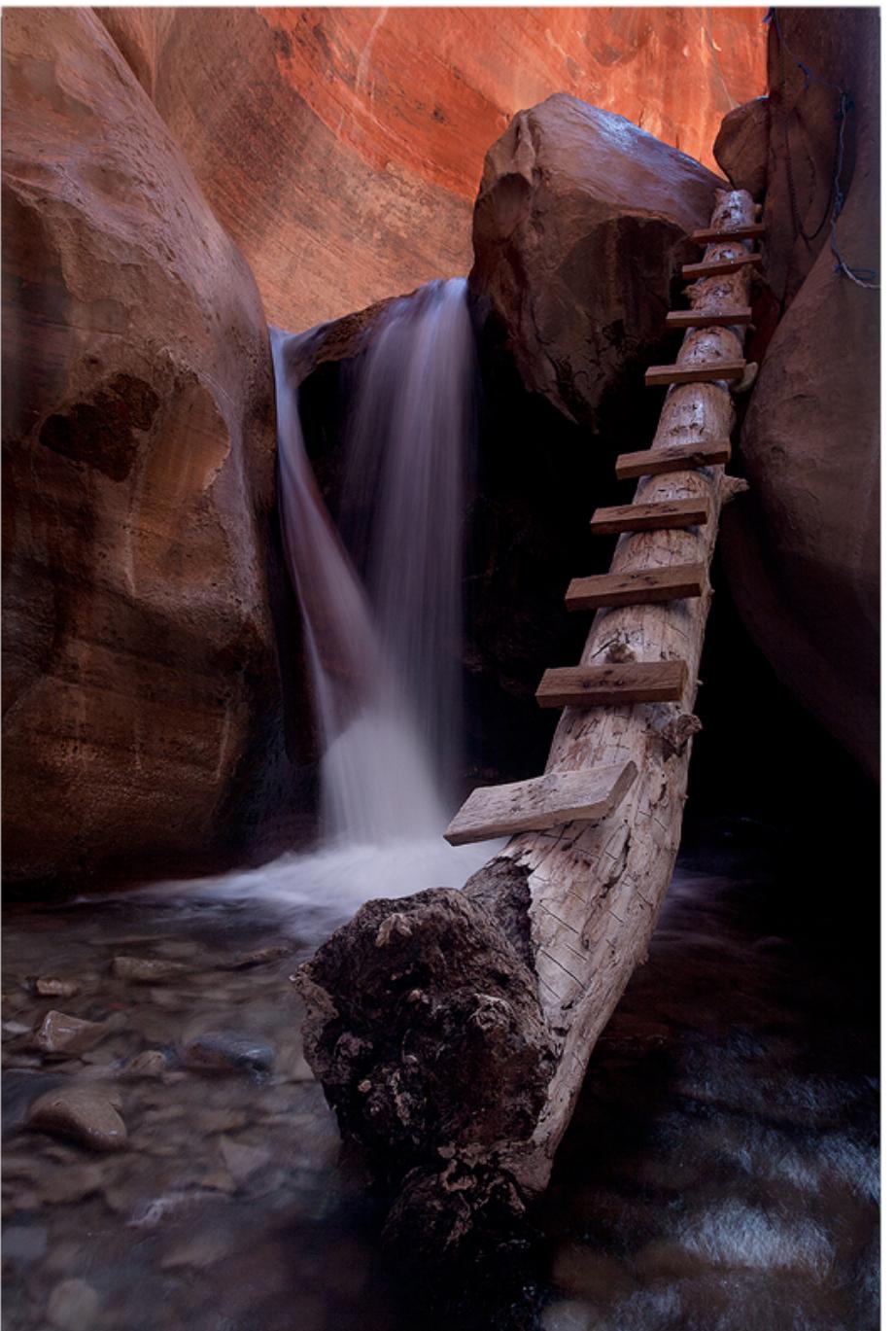
Uses IntelliJ IDEA

Leverages jQuery, HTML5, and CSS3

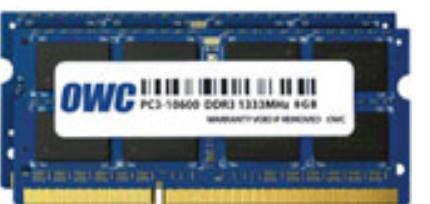
Creates High Performance Web Sites

For Mobile Devices, in the Cloud

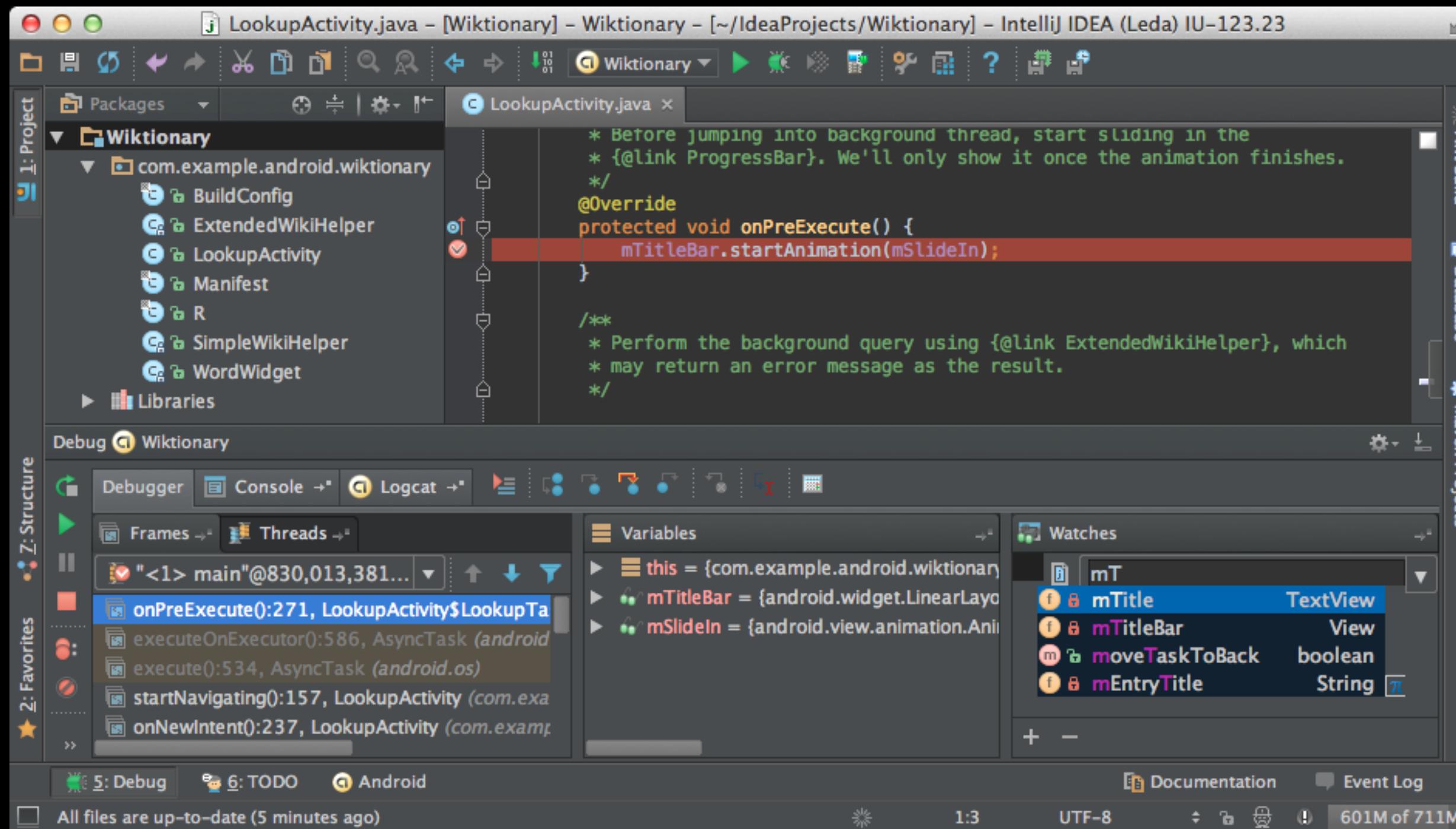
And cares about Security



Fast Hardware

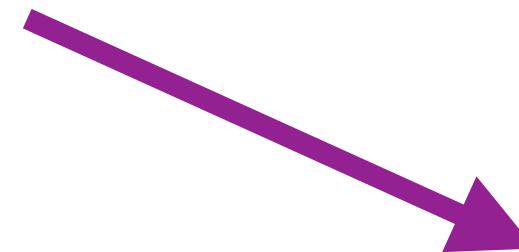


IntelliJ IDEA



Supports Zen Coding*

div#page>div.logo+ul#navigation>li*5>a



```
<div id=page>
  <div class=logo></div>
  <ul id=navigation>
    <li><a href=></a></li>
    <li><a href=></a></li>
    <li><a href=></a></li>
    <li><a href=></a></li>
    <li><a href=></a></li>
  </ul>
</div>
```

* In recent versions it has been renamed to Emmet

Java 7 and 8

Strings in switch statements

Diamond Syntax

Try with resources

Improved exception handling with multi-catch

NIO.2: Path, Files and Asynchronous I/O

```
Path path = FileSystems.getDefault().getPath(logs, access.log);
BufferedReader reader = Files.newBufferedReader(path, StandardCharsets.UTF_8);
```



Java 7 and 8

Parallel Collections

JSR 310 Date and Time API

Functional Interfaces with default method

Lambda Expressions (a.k.a. Closures)

```
// sort a list by lastName
List<Person> persons = ....;
persons.sort((p1, p2) ->
    p1.getLastName().compareTo(p2.getLastName()));
```

Nashorn JavaScript Engine



The Modern JVM Web Developer is aware of...

%s/Hibernate/Spring Data

Servlet 3

@WebServlet, @WebFilter, @WebListener

Asynchronous Servlets

WebApplicationInitializer (to eliminate web.xml)

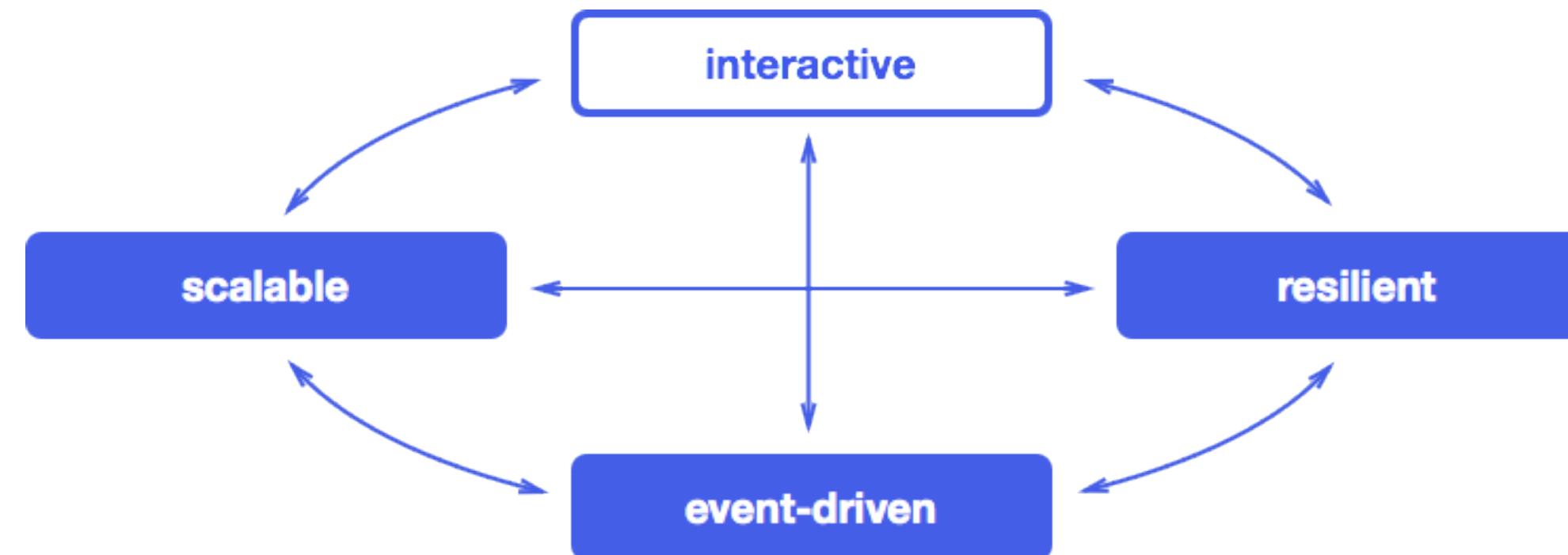
REST and Functional Programming



Reactive Applications

“Users expect millisecond response times and 100% uptime. Data needs are expanding into the petabytes.”

The Reactive Manifesto



Key Building Blocks

Observable Models

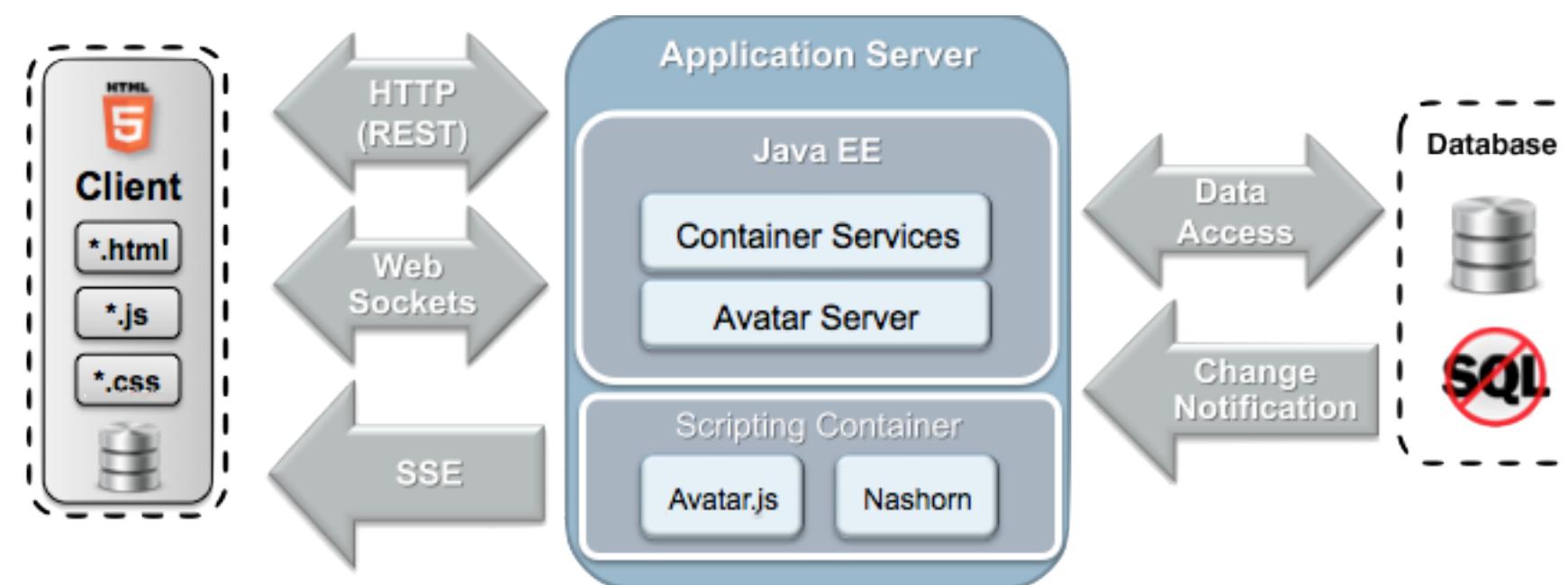
Event Streams

Stateful Clients



Avatar?

avatar.java.net



Scala

“Scala is like the dragon in Avatar. It will try to kill you, but if you master it, you can fly great distances with it and have a wonderful time.”

-- Venkat Subramaniam



Scala Basics

`def` starts a method

variables are started with `var` or `val`

variables are defined with name:type

semicolons are not required

```
import play.mvc.Http

trait Scalate {

  def render(args: (Symbol, Any)*) = {
    val template = Http.Request.current().action.replace(".", "/")
    ScalateTemplate(template).render(args: _*)
  }
}
```



Scala vs. Java

```
public class Car {  
    private final int year;  
    private int miles;  
  
    public int getYear() { return year; }  
    public int getMiles() { return miles; }  
    public void setMiles(int theMiles) { miles = theMiles; }  
  
    public Car(int theYear, int theMiles) {  
        year = theYear;  
        miles = theMiles;  
    }  
}
```



Scala vs. Java

```
class Car(val year: Int, var miles: Int)
```

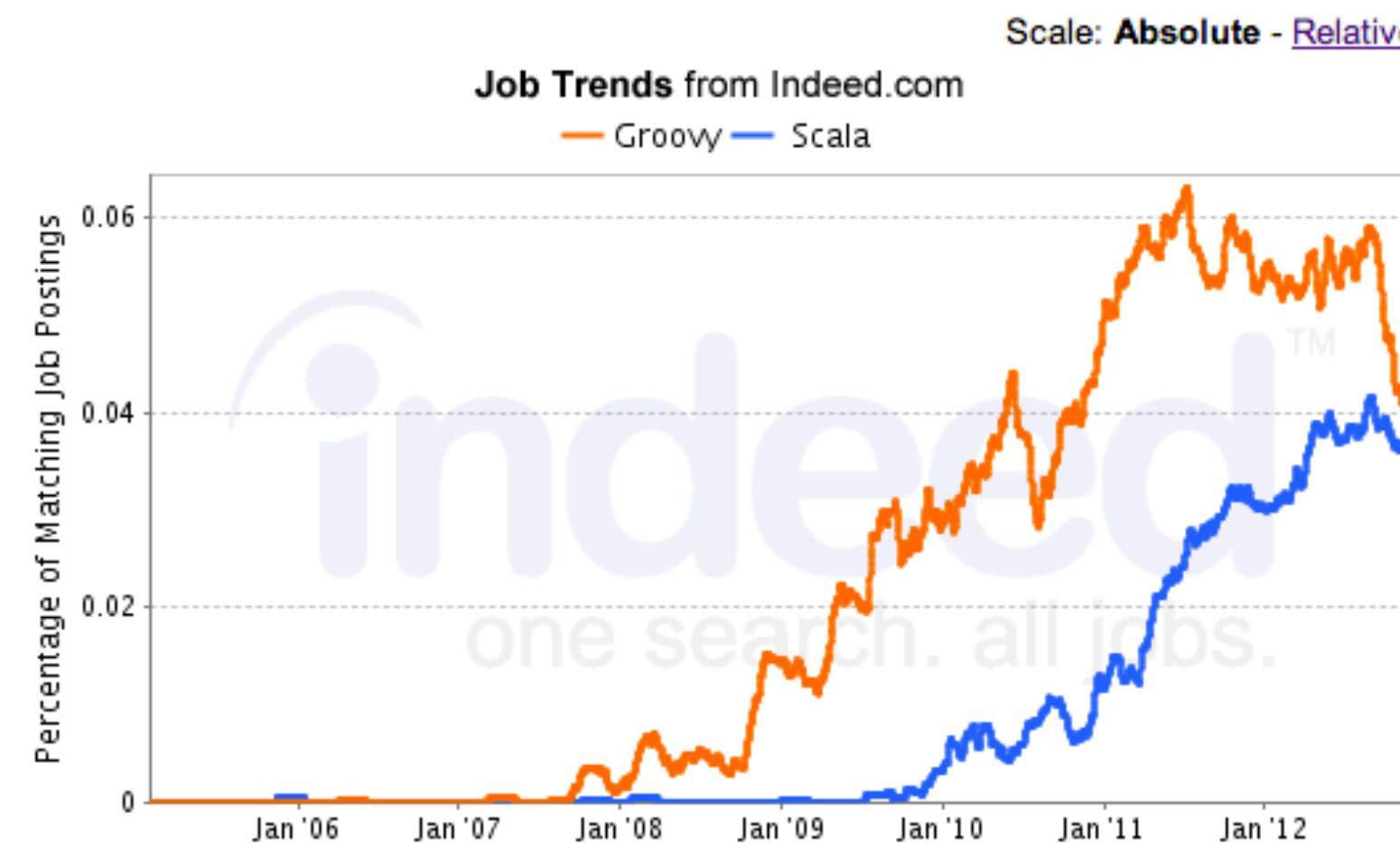


What about Groovy?



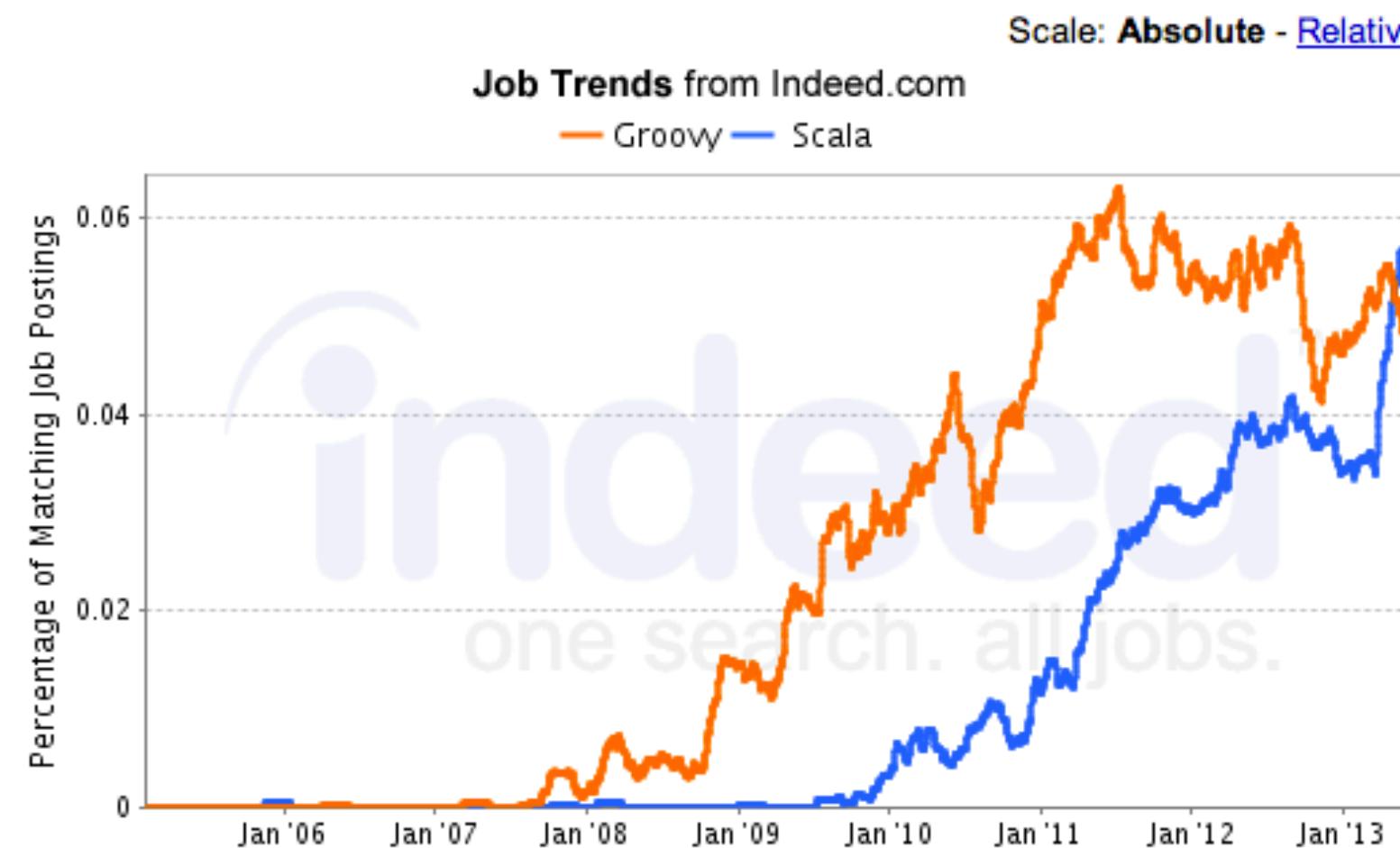
Groovy is still hot...

Groovy, Scala Job Trends



But sliding to Scala

Groovy, Scala Job Trends



Learning Scala

Scala for the Impatient - Cay Horstmann

Programming in Scala, 2nd Edition - Martin Odersky, Lex Spoon, and Bill Venners

Functional Programming Principles in Scala

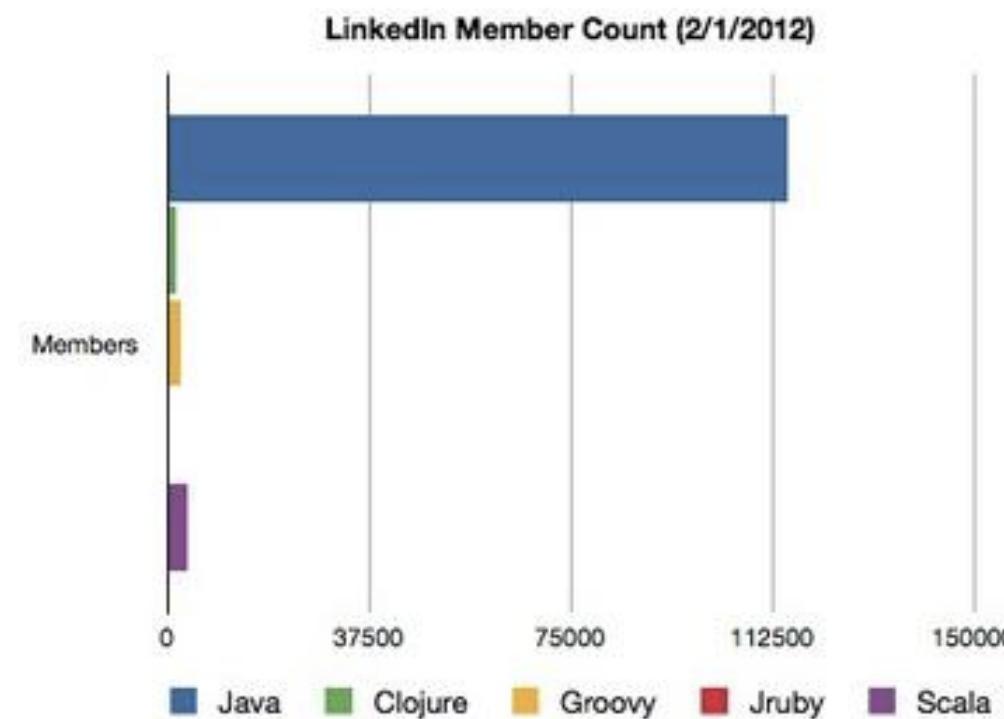
September 16th 2013 (7 weeks long)



Signup at <https://www.coursera.org/course/progfun>

The Java Language

“Java remains – in spite of the fragmented programming language landscape – a viable, growing language.”



<http://redmonk.com/sogrady/2012/02/08/language-rankings-2-2012/>

and ... it's still the most popular!

FEBRUARY 08, 2013

Java retakes the lead in language popularity

Recent security problems aside, Java moves past C in Tiobe's language popularity index thanks to its ties to Android development

By [Paul Krill](#) | [InfoWorld](#)

[Follow @pjkrill](#)

[Print](#) | [27 Comments](#) [77](#) [More](#)

Despite recent headlines about its security woes, Java has returned to the top spot in a monthly assessment of the popularity of programming languages, [10 months after being supplanted by the C language](#).

The Tiobe Programming Community Index again has Java as the most popular language. The company says Java's ties to Android mobile development are the reason for its recent good fortune. "Boosted by the success of Android phones, Java has gained most market share of all languages last half a year," Tiobe said.



The Services Developer

The screenshot shows a Mac OS X desktop with a window titled "InfoQ: How to GET a Cup of C". The URL in the address bar is "www.infoq.com/articles/webber-rest-workflow". The page content is an article titled "How to GET a Cup of Coffee" by Jim Webber, Savas Parastatidis & Ian Robinson, posted on Oct 02, 2008. The article discusses the role of the Web as a primary design philosophy for application integration. The page includes a sidebar for QCon events and a footer with various navigation links.

Facilitating the spread of knowledge and innovation in enterprise software development

InfoQ

En | 中文 | 日本語 | Br

562,907 Jan unique visitors

Development Architecture & Design Process & Practices Operations & Infrastructure Enterprise Architecture

Mobile HTML5 JavaScript Cloud Agile SOA Agile Techniques NoSQL Cloud Security

Article

How to GET a Cup of Coffee

Posted by Jim Webber, Savas Parastatidis & Ian Robinson on Oct 02, 2008

Sections Enterprise Architecture, Architecture & Design Topics Workflow / BPM, Business Process Management, SOA, REST, Business, Enterprise Architecture, Web services, Architecture, Design Guideline, Patterns

Share |

We are used to building distributed systems on top of large middleware platforms like those implementing CORBA, the Web Services protocols stack, J2EE, etc. In this article, we take a different approach, treating the protocols and document formats that make the Web tick as an application platform, which can be accessed through lightweight middleware. We showcase the role of the Web in application integration scenarios through a simple customer-service interaction scenario. In this article, we use the Web as our primary design philosophy to distil and share some of the thinking in our forthcoming book "GET /connected - Web-based integration" (working title).

Introduction

The integration domain as we know it is changing. The influence of the Web and the trend towards more agile practices are challenging our notions of what constitutes good integration. Instead of being a specialist activity conducted in the void between systems – or even worse, an afterthought – integration is now an everyday part of successful solutions.

Yet, the impact of the Web is still widely misunderstood and underestimated in enterprise computing. Even those who are Web-savvy often struggle to understand that the Web isn't about middleware solutions supporting XML over HTTP, nor is it a crude RPC mechanism. This is a shame because the Web has much more value than simple point-to-point connectivity; it is in fact a robust integration platform.

In this article we'll showcase some interesting uses of the Web, treating it as a pliant and robust platform for doing very cool things with enterprise systems. And there is nothing that typifies enterprise software more than workflows...

Fast APIs



JAX-RS

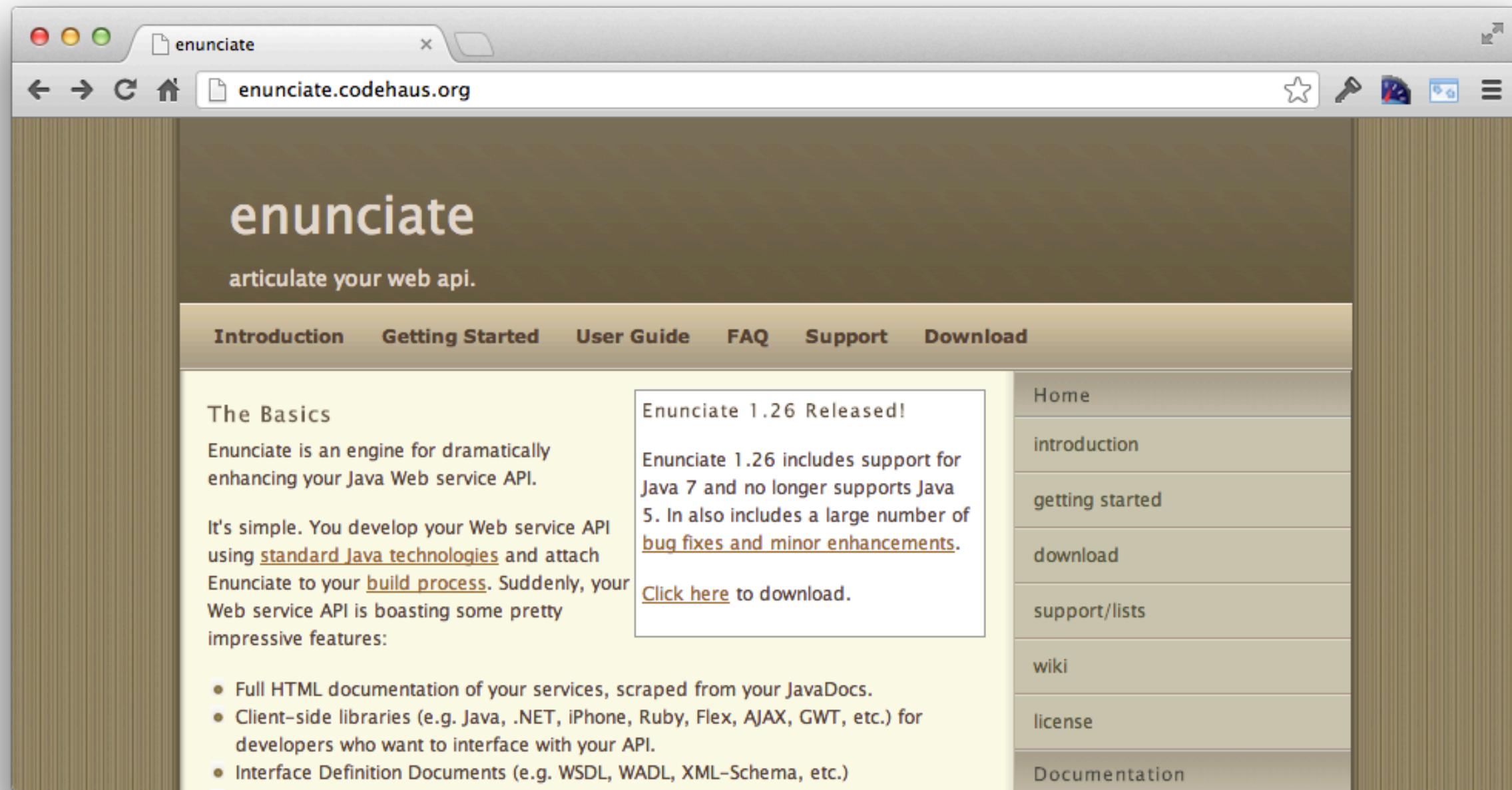


Dropwizard

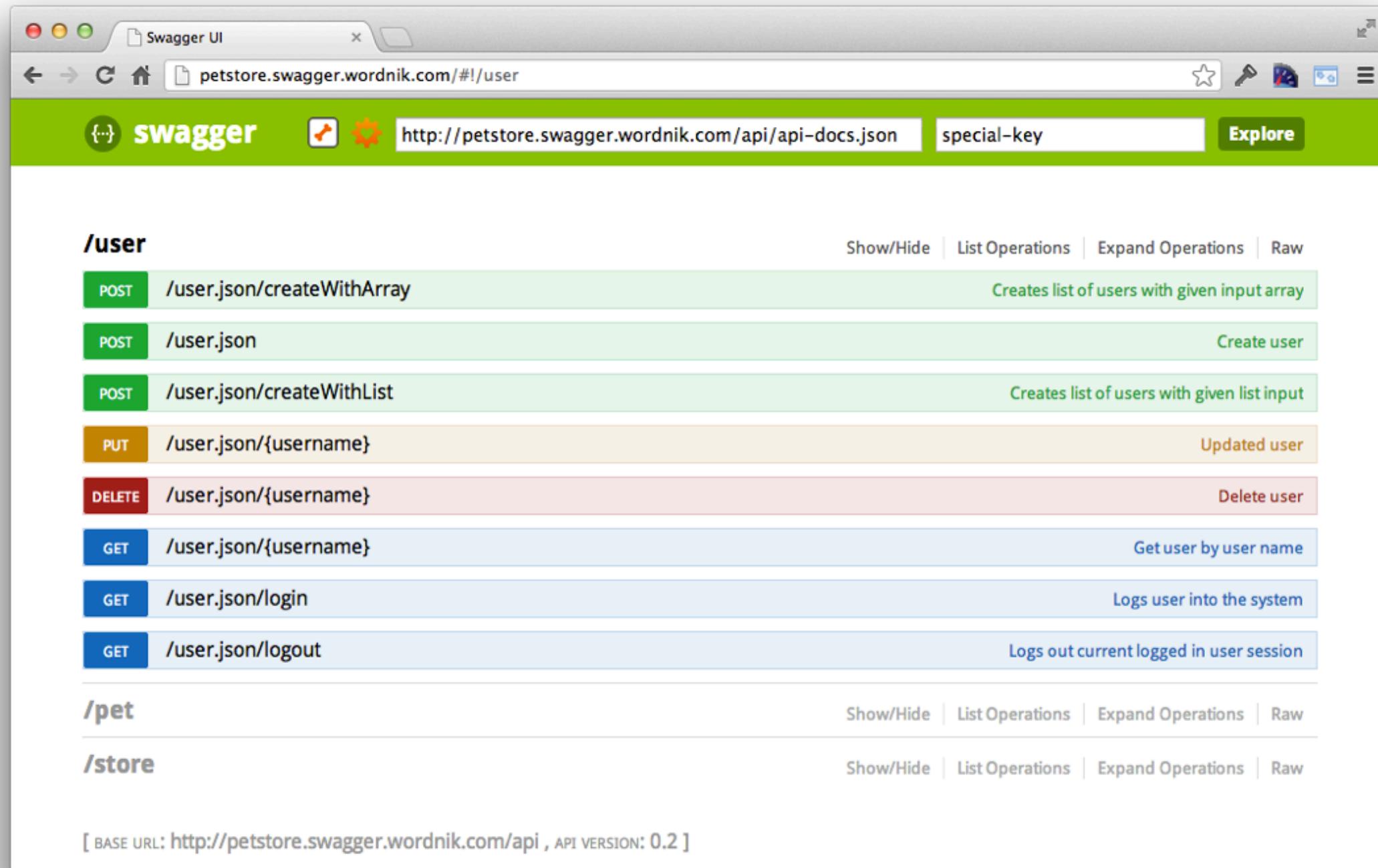


AppFuse

Document Your API



Document Your API



But if want to remain a **Web** Developer...



Modern Principles

The screenshot shows a web browser window with the title bar "Modern Principles in Web De" and the URL "blogs.atlassian.com/2012/01/modern-principles-in-web-development/". The page itself is the Atlassian Blogs interface, featuring a dark blue header with the "Atlassian Blogs" logo and a search bar. The main content area has a white background and displays the following information:

Modern Principles in Web Development

By Rich Manalang, Developer Advocate
About Developer
On January 18, 2012

11 Tweet 162 Like 8

I've been kickstarting a bunch of small web apps lately. It seems like every time I start a new project, there's always something new that causes me to adjust my development principles. I thought it might be good to take a snapshot of what's "in" today. I like to think of web development phases starting from idea to delivery... all of it backed by strong principles of how to build great apps.

The following are my core web development principles today:

- Designing for mobile first (even if you're not building a mobile app)
- Build only single page apps
- Create and use your own REST API
- "Sex sells" applies to web apps

Browser Tools

Firebug for FireFox



Chrome Developer Tools



Elements & Console

Settings

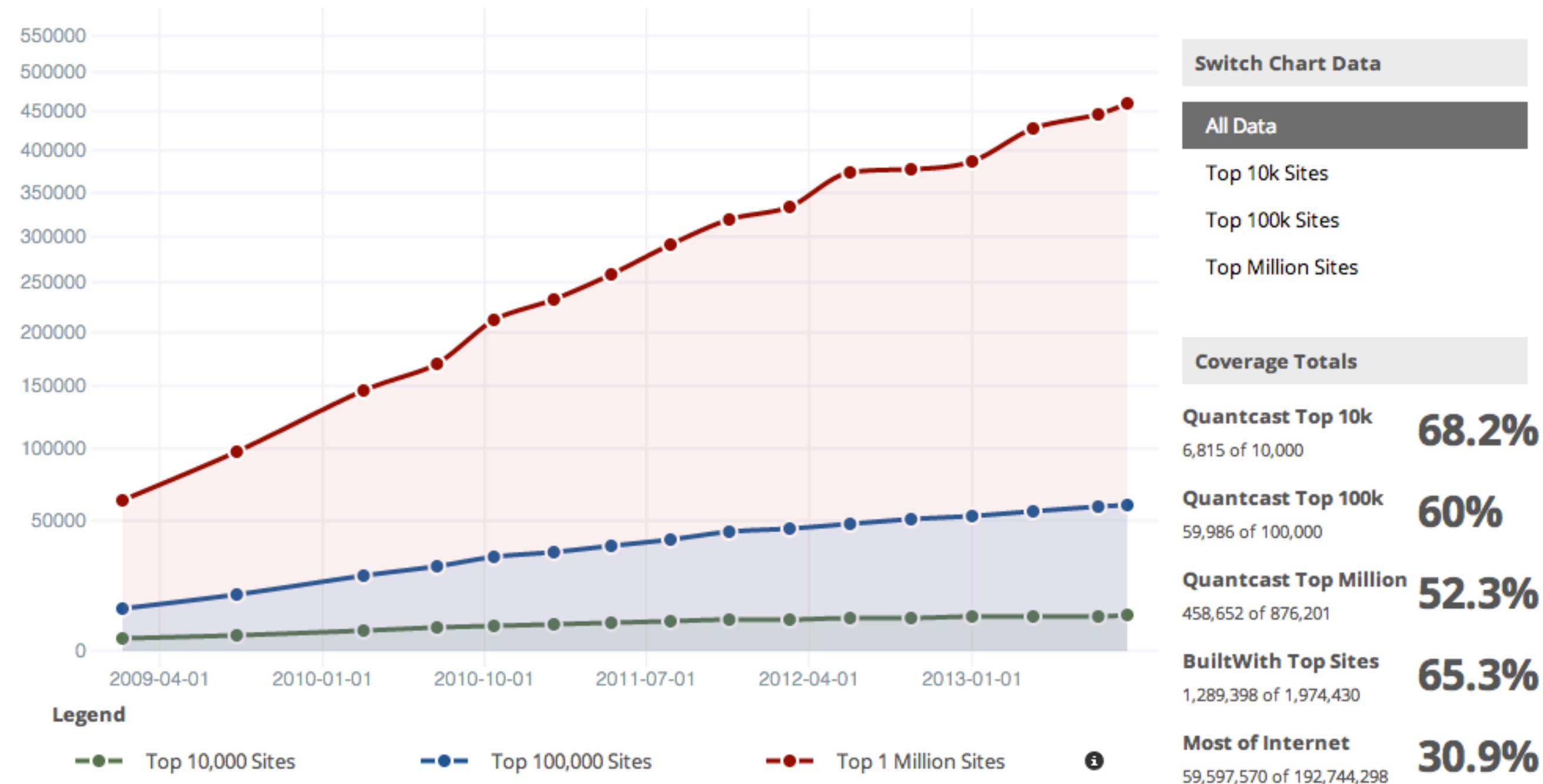
PageSpeed Insights

<http://www.igvita.com/slides/2012/devtools-tips-and-tricks>

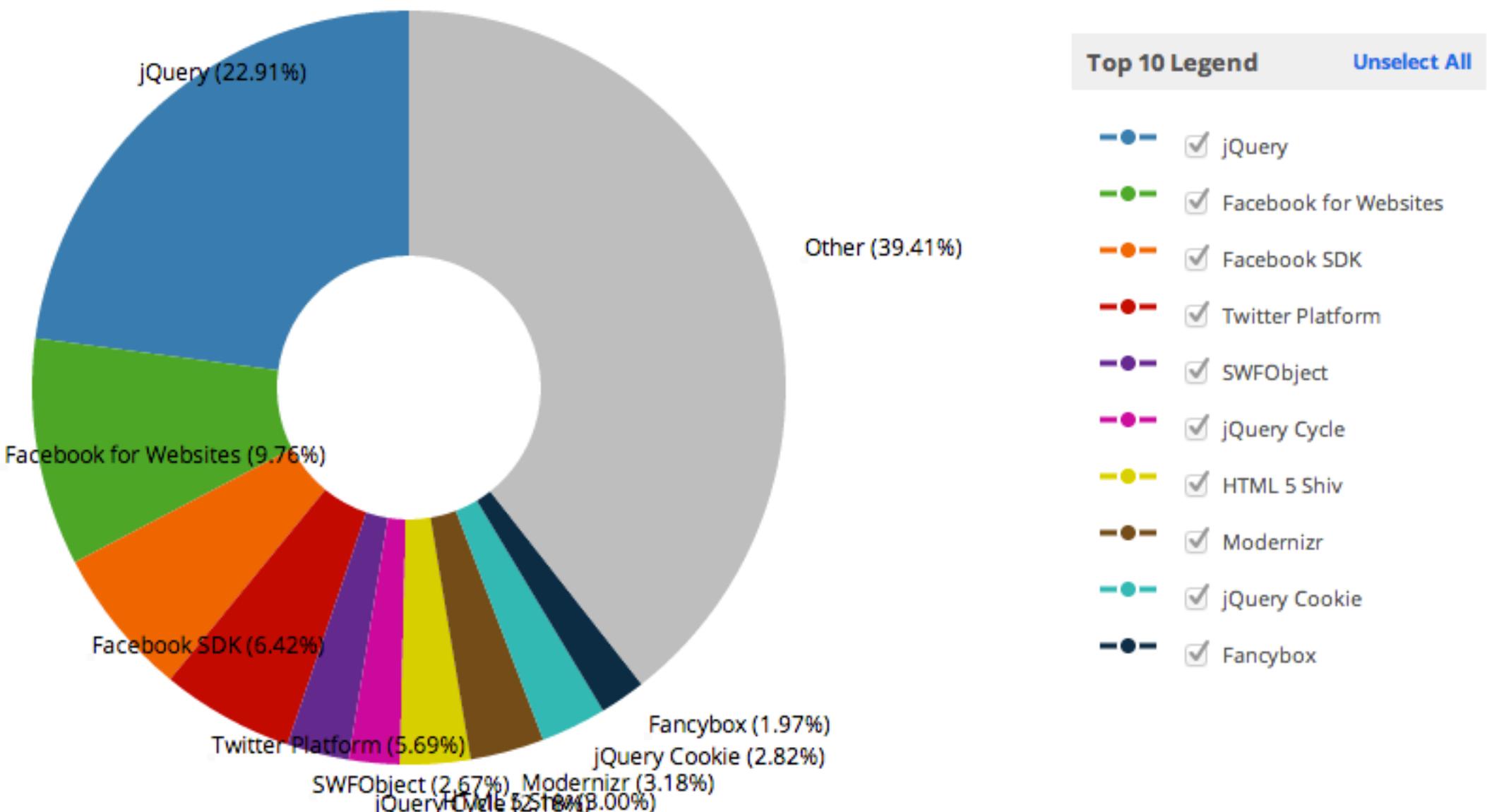
jQuery

<http://trends.builtwith.com/javascript/jQuery>

Websites using jQuery



JavaScript Distribution in Top 10,000 Sites



jQuery

jQuery CDN at <http://code.jquery.com/>

`$(document).ready`



`$(document).on('click', '#selector', function() {})`

jQuery UI for Dialogs, Sliders, Calendars

`$.ajax` and `$('#div').load(url)`



JavaScript

The Good Parts

Lambda

Dynamic Objects

Loose Typing

Object Literals



JavaScript Tips

```
"use strict";  
  
JSON.parse( text ); // converts JSON string to JavaScript Object  
JSON.stringify( obj ); // converts an Object into a serialized JSON string
```

<http://ejohn.org/blog/ecmascript-5-strict-mode-json-and-more/>

JavaScript Programming Patterns

The Old-School Way

Singleton

Module Pattern

Revealing Module Pattern

Custom Objects

Lazy Function Definition



<http://www.klauskomenda.com/code/javascript-programming-patterns/>

Revealing Module Pattern

```
01 var myRevealingModule = function () {  
02  
03     var privateVar = "Ben Cherry",  
04         publicVar  = "Hey there!";  
05  
06     function privateFunction() {  
07         console.log( "Name:" + privateVar );  
08     }  
09  
10     function publicSetName( strName ) {  
11         privateVar = strName;  
12     }  
13  
14     function publicGetName() {  
15         privateFunction();  
16     }  
17  
18     // Reveal public pointers to  
19     // private functions and properties  
20  
21     return {  
22         setName: publicSetName,  
23         greeting: publicVar,  
24         getName: publicGetName  
25     };  
26  
27 }();  
28  
29 myRevealingModule.setName( "Paul Kinlan" );
```

CoffeeScript

JS → Coffee Coffee → JS

```
/* Type here! */

(function ($) {
  $.fn.highlight = function () {
    $(this).css({ color: 'red', background: 'yellow' });
    $(this).fadeIn();
  };
})(jQuery);

((($) =>
  $.fn.highlight = ->
    $(this).css
      color: "red"
      background: "yellow"

    $(this).fadeIn()
  ) jQuery
```

Fork me on GitHub



MORE INFO ↓

AngularJS

A JavaScript MVW Framework

From Google, MIT Licensed

Data-binding, Controllers, Dependency Injection

Localization, Components, Testable

Angular-seed for Scaffolding

Great Documentation and Community



AngularJS Basics

The Basics

index.html

```
1. <!doctype html>
2. <html ng-app>
3.   <head>
4.     <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.0.8/
angular.min.js"></script>
5.   </head>
6.   <body>
7.     <div>
8.       <label>Name:</label>
9.       <input type="text" ng-model="yourName" placeholder="Enter a name here">
10.      <hr>
11.      <h1>Hello {{yourName}}!</h1>
12.    </div>
13.  </body>
14. </html>
```

Choosing a JavaScript MVC Framework

Simplified by one man: **Addy Osmani**

Journey Through The JavaScript MVC Jungle

Learning JavaScript Design Patterns





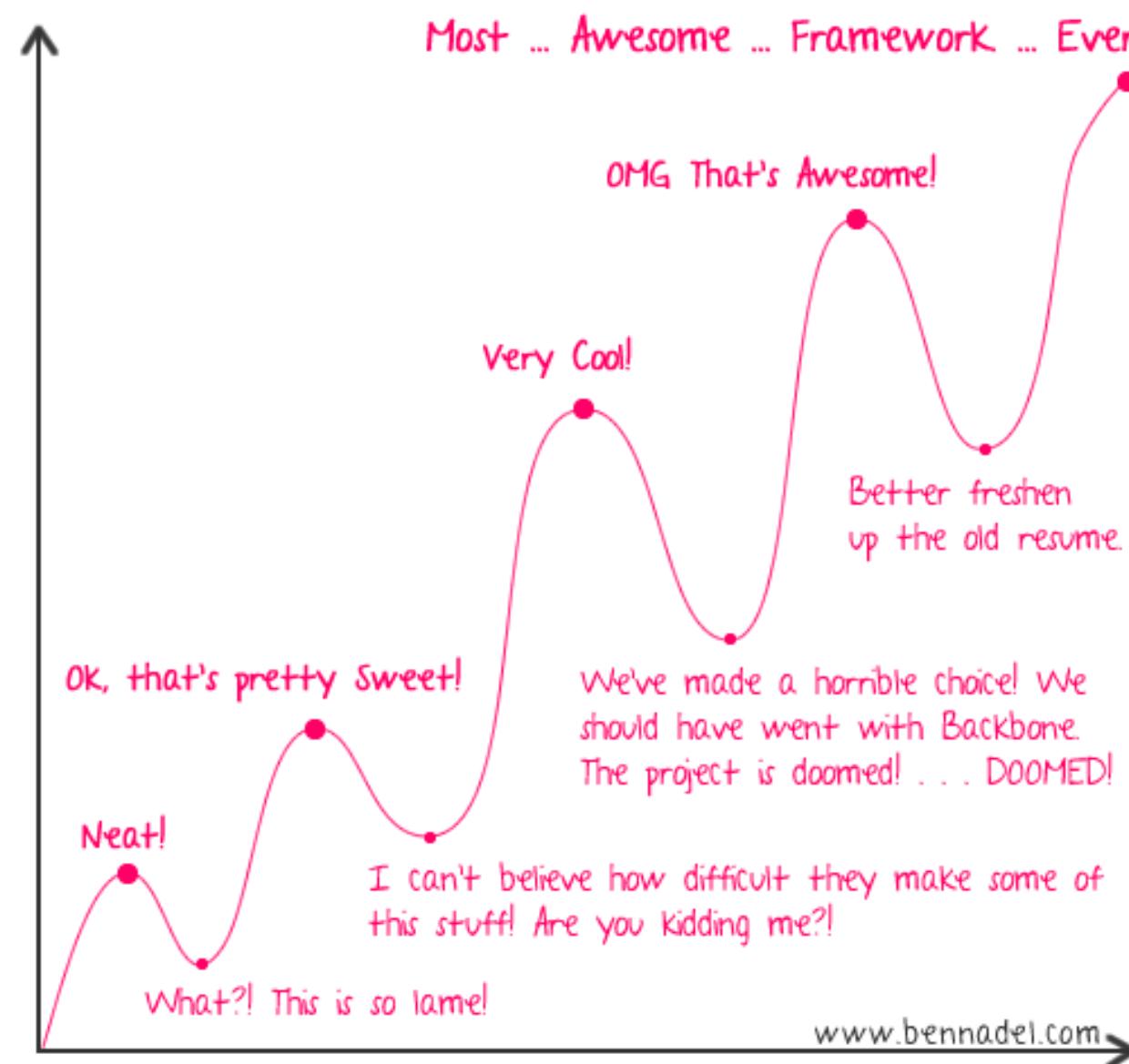
TodoMVC

Helping you **select** an MV* framework

[Download \(1.0.1\)](#) [View project on GitHub](#)



Or Just Choose AngularJS



My Feelings About AngularJS Over Time

<http://bit.ly/UICDZi>

My Angular JS Experience

Taleo Business Edition – Premium

Administration Help View Logout

Pipeline Requisitions Candidates Accounts Contacts Employees OnBoard/OffBoard Reports Reviews Compensation Users TBE Apps Position Control

Switch to My View

Interviews List 25 My View "Active Candidates" 38 My View "My Tasks" 12 Awaiting My Approval: Reqs 02

Awaiting My Approval: Goals 03 My View "My Requisition" 12 My Active Direct Reports 13 My Active Subordinates 19

My View "Positions" 19 Awaiting My Approval: Roles 09 Awaiting My Approval: Offers 01 Onboarding Employees 13

My Onboarding Activities 21 Offboarding Employees 06 My View "Employee Goals" 11 My View "My Goals" 02

Show More ▾

My Tasks

Awaiting My Approval: Goals 3 My View "My Tasks" Overdue 24 Awaiting My Approval: Requisitions 9 Awaiting My Approval: Offers 1 Awaiting My Approval: Plans 1 Awaiting My Approval: Reviews 2

Show More ▾

Charts

My View "Positions" by Department (Total: 19)

5.263%	36.84%	57.89%
--------	--------	--------

My View "My Goals" by status (Total: 2)

100.0%

Favorite Reports

Candidate Days per Status by Requisition - by Department
Report on the actual and average number of days candidates spend in each requisition-specific status grouped by department.

Candidate Interviews and Feedback By Requisition
Report on all candidate interviews by interview date, location, region, department, or interviewer.

Candidate Interviews and Feedback by Requisition - Current User Owner
Report on all candidate interviews filtered by owner.

Candidate Main Status Summary
Report on the number of candidates and their relationship status with our company.

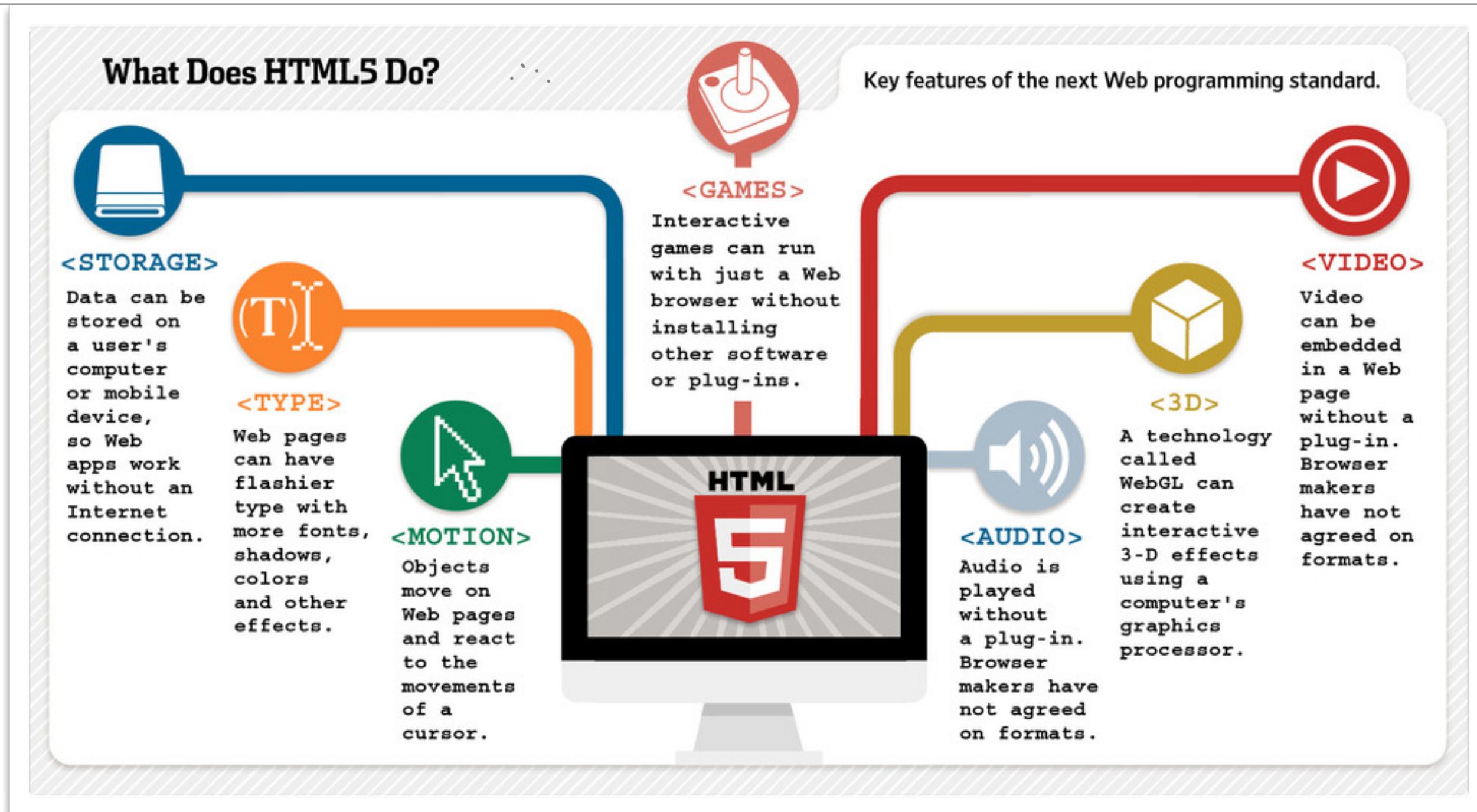
Candidate Pipeline by Requisition - by Department
Report on the total number of candidates who have progressed through each requisition-specific candidate status grouped by department.

Candidates 13_3
Report on candidates by requisition. Includes the candidates, requisition candidates, requisitions, requisition owners, requisition approvers, sources, employment history, education history, references, certificate and licenses, residence history, tests,

Candidates By Requisition
Report on all candidates by requisition by status, location, region, or department.

Candidates By Requisition - Current User Owner

HTML5



How do you write HTML5?

<!DOCTYPE html>

<article> <aside> <section>
<header> <footer> <nav>
<audio> <canvas> <video>
<datalist> <details>



<applet> <center>
<frame> <frameset>



<https://developer.mozilla.org/en-US/docs/HTML/HTML5>

HTML5 Forms

```
<form>
  <input type=text placeholder=Enter your search terms>
  <input type=submit value=Search/>
</form>
```

```
<form>
  <input type=text autofocus>
  <input type=submit value=Search/>
</form>
```

```
<form>
  <input type=text required>
  <input type=submit value=Search/>
</form>
```

<http://diveintohtml5.info/forms.html>

HTML5 Forms

```
<form>
  <input type=email>
  <input type=submit value=Go>
</form>
```

HTML5 Killer Features



Editable Text



Pitbull
Don't Stop The Party

A screenshot of a music player interface showing a waveform, song title, artist, duration, and genre information. The song is Pitbull's "Don't Stop The Party". It shows a play button, album art, and a progress bar from 1.05 to 3.26. The genre is listed as Pop.

HTML5 Killer Features



CSS3 Secrets

Animated Transitions

```
transform: rotateY(180deg);
```

Rounded Corners

```
border-radius: 8px 8px 0 0;
```

Drop Shadows

```
box-shadow: 2px 2px 4px 4px;
```

Gradient Colors

Styling based on sibling count

More cursors for better usability

<http://lea.verou.me/css3-secrets>



CSS3 Media Queries

```
/* Smartphones (portrait and landscape) ----- */
@media only screen and (min-device-width: 320px)
and (max-device-width: 854px) {
    body {
        padding: 10px;
    }

    textarea {
        width: 90%;
    }
}

/* iPad (portrait and landscape) ----- */
@media only screen and (min-device-width: 768px)
and (max-device-width: 1024px) {
    body {
        padding-top: 50px;
    }
}
```

Cool HTML5 Demos

<http://fff.cmiscm.com>



Sheep

15

about this site • screen saver

share • fullscreen

Bootstrap

Good-looking websites by default

Layouts, navs, pagination, buttons

Mobile First (aka Responsive Design)

Over a dozen Custom jQuery Plugins

Themes from {wrap}bootstrap (Market) and
Bootswatch (OSS)



{less} the dynamic stylesheet language

```
// LESS

.rounded-corners (@radius: 5px) {
  -webkit-border-radius: @radius;
  -moz-border-radius: @radius;
  -ms-border-radius: @radius;
  -o-border-radius: @radius;
  border-radius: @radius;
}

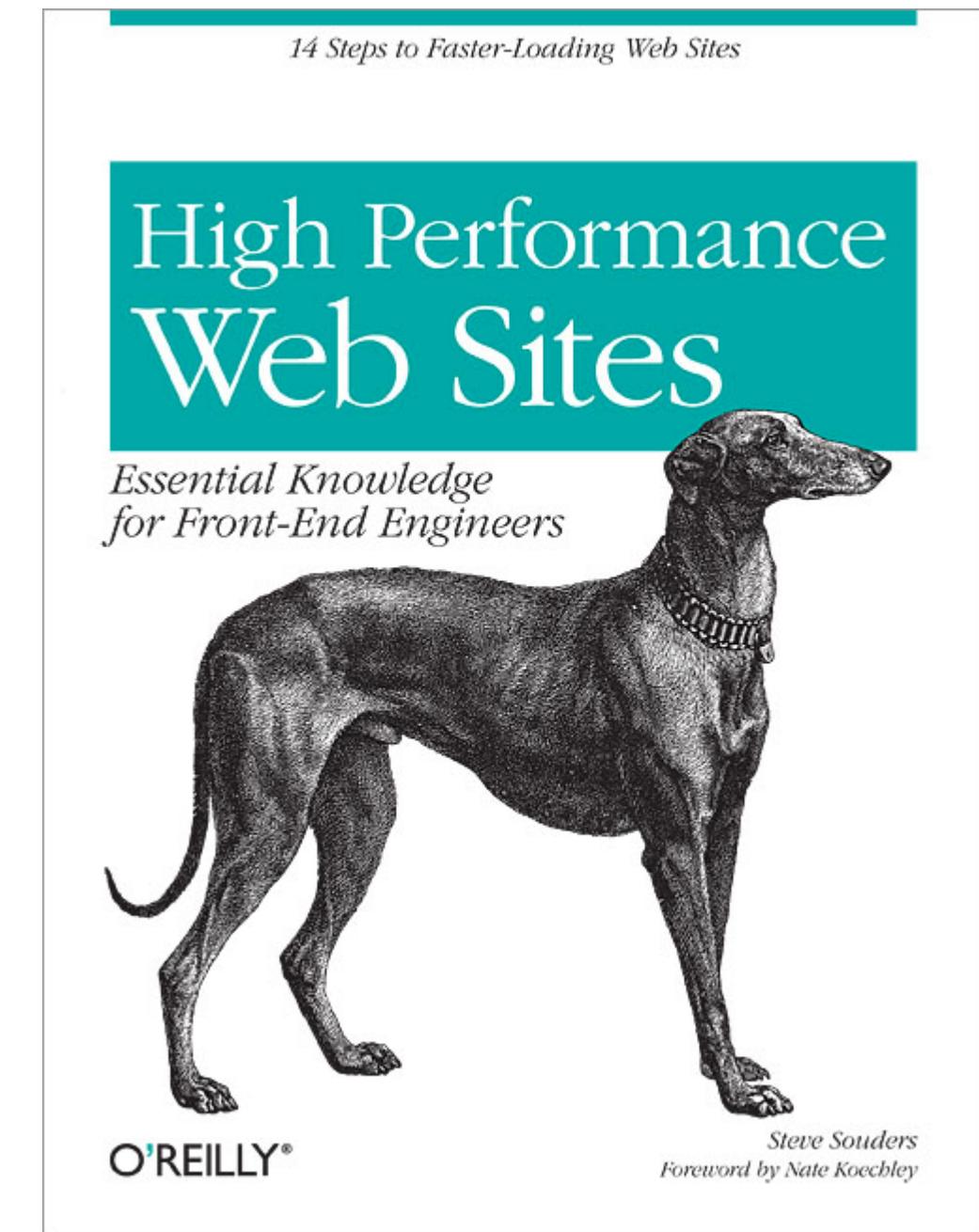
#header {
  .rounded-corners;
}
#footer {
  .rounded-corners(10px);
}

/* Compiled CSS */

#header {
  -webkit-border-radius: 5px;
  -moz-border-radius: 5px;
  -ms-border-radius: 5px;
  -o-border-radius: 5px;
  border-radius: 5px;
}
#footer {
  -webkit-border-radius: 10px;
  -moz-border-radius: 10px;
  -ms-border-radius: 10px;
  -o-border-radius: 10px;
  border-radius: 10px;
}
```

High Performance Web Sites

1. Make Fewer HTTP Requests
2. Use a Content Delivery Network
3. Add Expires Headers
4. Gzip Components
5. Put Stylesheets at the Top
6. Put Scripts at the Bottom
7. Avoid CSS Expressions



High Performance Web Sites

8. Make JavaScript and CSS External
9. Reduce DNS Lookups
10. Minify JavaScript
11. Avoid Redirects
12. Remove Duplicates Scripts
13. Configure ETags
14. Make Ajax Cacheable



My Page Speed Experience

Biggest Problem is HTTP Requests

Minify and Concatenate

GZipping has biggest score gain

Expires Headers for Browser Caching

Use Apache or Nginx

Image Sprites for CSS

<http://spritecow.com> is great



Wro4j

Open Source Java project for optimization of web resources

Provides concatenation and minimization of JS and CSS

Gzip, YUI Compressor, JsHint, JsHint, CssLint, LESS, SASS, CoffeeScript, Dojo Shinksafe



Sidenote: WebJars and UrlRewrite Filter

WebJars: client-side web libraries packages in JARs

<http://webjars.org>

Servlet 3, Play 2, Grails, Dropwizard, Spring MVC, Tapestry and Wicket

UrlRewriteFilter: like Apache's mod_rewrite

<http://tuckey.org/urlrewrite>

Clean URLs and just about everything else

Spring MVC's <default-servlet-handler/> [is your friend](#)



Techniques

Versioning Static Assets with UrlRewriteFilter

http://raibledesigns.com/rd/entry/versioning_static_assets_with_urlrewritefilter

Adding web resource fingerprinting to AppFuse with wro4j

<http://www.operatornew.com/2012/10/adding-web-resource-fingerprinting-to.html>

Improving AppFuse's PageSpeed with Apache

http://raibledesigns.com/rd/entry/improving_appfuse_s_pagespeed_with

But what about nginx?

An open-source, high-performance HTTP server and reverse proxy, as well as an IMAP/POP3 proxy server

Powers Netflix, Wordpress.com, GitHub and Heroku

<http://kevinworthington.com/nginx-for-mac-os-x-mountain-lion-in-2-minutes/>



Apache Gzip and Expires Headers

mod_pagespeed - <https://developers.google.com/speed/pagespeed/mod>

Automatically applies web performance best practices w/o modification

Improving AppFuse's PageSpeed with Apache

Configured mod_deflate, mod_expires and turned on KeepAlive

PageSpeed went from 24 to 96!

YSlow went from 90 to 98



/etc/httpd/conf.d/deflate.conf

```
<IfModule mod_deflate.c>

    SetOutputFilter DEFLATE

        AddOutputFilterByType DEFLATE text/plain text/html text/xml text/css
application/xml application/xhtml+xml application/rss+xml application/javascript
application/x-javascript

    DeflateCompressionLevel 9

        BrowserMatch ^Mozilla/4 gzip-only-text/html
        BrowserMatch ^Mozilla/4\.0[678] no-gzip
        BrowserMatch \bMSIE !no-gzip !gzip-only-text/html

    DeflateFilterNote Input instream
    DeflateFilterNote Output outstream
    DeflateFilterNote Ratio ratio
    LogFormat '%r %{outstream}n/%{instream}n (%{ratio}n%%)' deflate

</IfModule>
```

/etc/httpd/conf.d/expire.conf

```
<IfModule mod_expires.c>
    ExpiresActive On
    <FilesMatch \.(jpe?g|png|gif|js|css)$>
        ExpiresDefault access plus 1 week
    </FilesMatch>
    ExpiresByType image/jpeg access plus 1 week
    ExpiresByType image/png access plus 1 week
    ExpiresByType image/gif access plus 1 week
    ExpiresByType text/css access plus 1 week
    ExpiresByType application/javascript access plus 1 week
    ExpiresByType application/x-javascript access plus 1 week
</IfModule>
```

Application Architecture

Server to Client

Performance implications

Twitter

Airbnb

Charm

Ajaxified Body with pjax

<https://github.com/defunkt/jquery-pjax>



Mobile Devices

If developing a mobile app with web technologies

Use PhoneGap or Sencha Touch

Otherwise, add a **viewport** meta tag

```
<meta name=viewport content=width=device-width, initial-scale=1.0>
```



Mobile Devices - CSS3 Media Queries

```
/* Smartphones (portrait and landscape) -----  
*/  
@media only screen and (min-device-width: 320px) and  
(max-device-width: 854px) {  
    /* Smartphone rules */  
}  
  
/* iPad (portrait and landscape) ----- */  
@media only screen and (min-device-width: 768px) and  
(max-device-width: 1024px) {  
    /* Tablet rules */  
}
```



Mobile Devices - Hide Address Bar

```
<script type="text/javascript">
  // Hide address bar for smartphones
  /Mobile/.test(navigator.userAgent) && !location.hash
    && setTimeout(function () {
      if (!pageYOffset) window.scrollTo(0, 1);
    }, 1000);
</script>
```

Mobile Devices - Disable Focus Zoom

```
(function(doc) {
    var addEvent = 'addEventListener',
        type = 'gesturestart',
        qsa = 'querySelectorAll',
        scales = [1, 1],
        meta = qsa in doc ? doc[qsa]('meta[name=viewport]') : [];

    function fix() {
        meta.content = 'width=device-width,minimum-scale=' +
                      scales[0] + ',maximum-scale=' + scales[1];
        doc.removeEventListener(type, fix, true);
    }

    if ((meta = meta[meta.length - 1]) && addEvent in doc) {
        fix();
        scales = [.25, 1.6];
        doc[addEvent](type, fix, true);
    }
}(document));
```

The Cloud



A screenshot of a blog post titled "Heroku for Java" by Adam, posted on August 25, 2011. The post announces the public beta of Heroku for Java, which is the fourth official language available on the Cedar stack. It highlights Java's popularity and its strengths as a programming language. To the right of the text is a graphic of a small green seedling growing out of several brown coffee beans.

Heroku for Java
by Adam - Aug 25, 2011

We're pleased to announce the public beta of Heroku for Java. Java is the fourth official language available on the [Cedar](#) stack.

Java is, by [many measures](#), the world's most popular programming language. In addition to its large and diverse developer base, it offers a huge ecosystem of libraries and tools, an extremely well-tuned VM for fast and reliable runtime performance, and an accessible C-like syntax.

But there are also many criticisms commonly leveled against the language. We'll take a closer look at Java's strengths and weaknesses in a moment, but first:

[Heroku for Java in 2 minutes](#)



The Cloud



Supports Ruby, Node.js, Clojure, Java, Python and Scala

```
$ git push heroku master
Counting objects: 47, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (25/25), done.
Writing objects: 100% (47/47), 10.25 KiB, done.
Total 47 (delta 19), reused 42 (delta 17)

-----> Heroku receiving push
-----> Java app detected
-----> Installing OpenJDK 1.6... done
-----> Installing Maven 3.0.3... done
-----> Installing settings.xml... done
-----> executing /app/tmp/repo.git/.cache/.maven/bin/mvn -B -Duser.home=/tmp/build_3k0p14ghrmdz
[INFO] Scanning for projects...
```

The Cloud



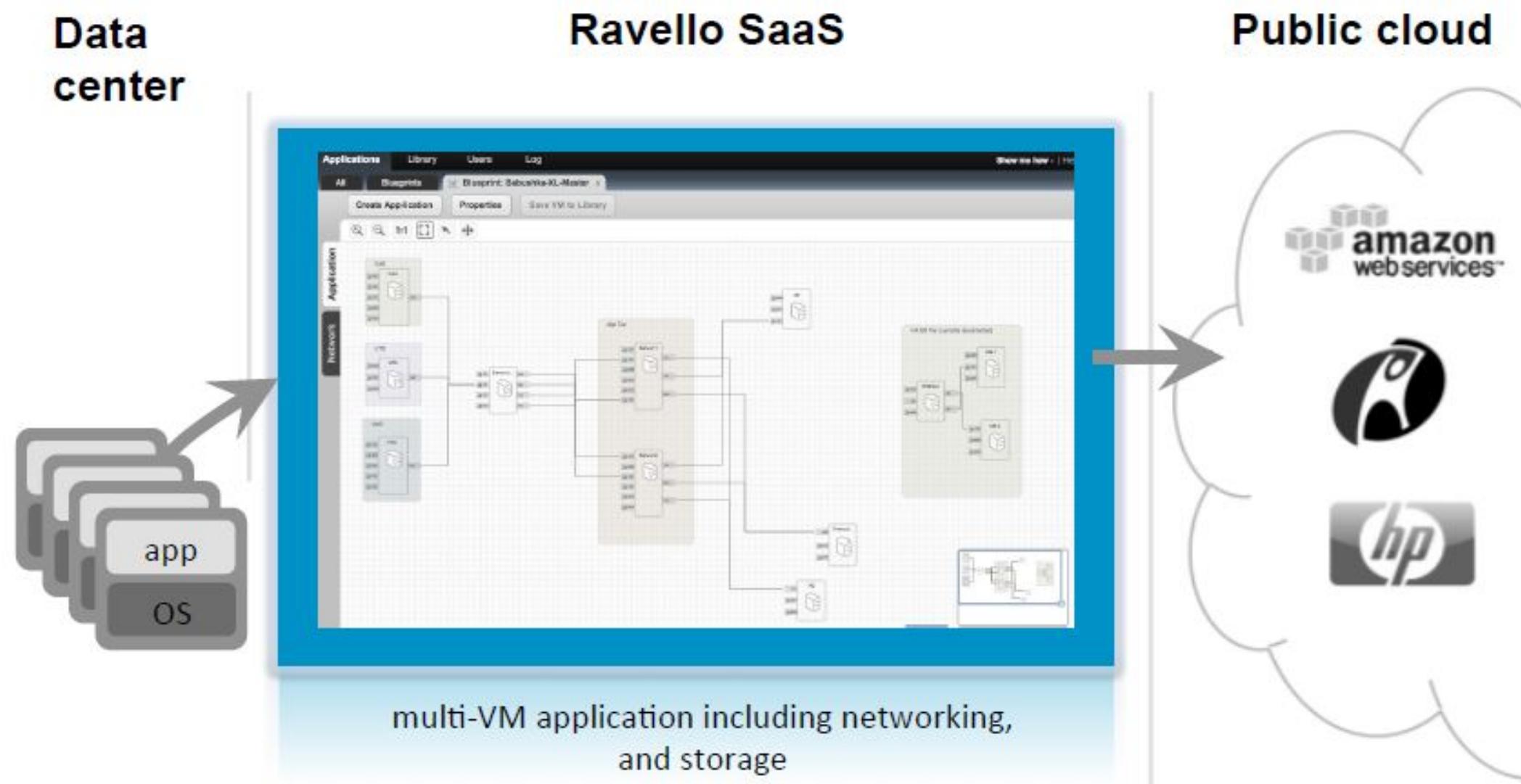
Supports Spring, Grails, Scala, Play, Node.js, Ruby/Rails/Sinatra

Services: MySQL, PostgreSQL, MongoDB, Redis, RabbitMQ

The Cloud



Upload VMs, define app, and deploy



Apache TomEE



Security Matters

Be aware of SQL and Content Injection

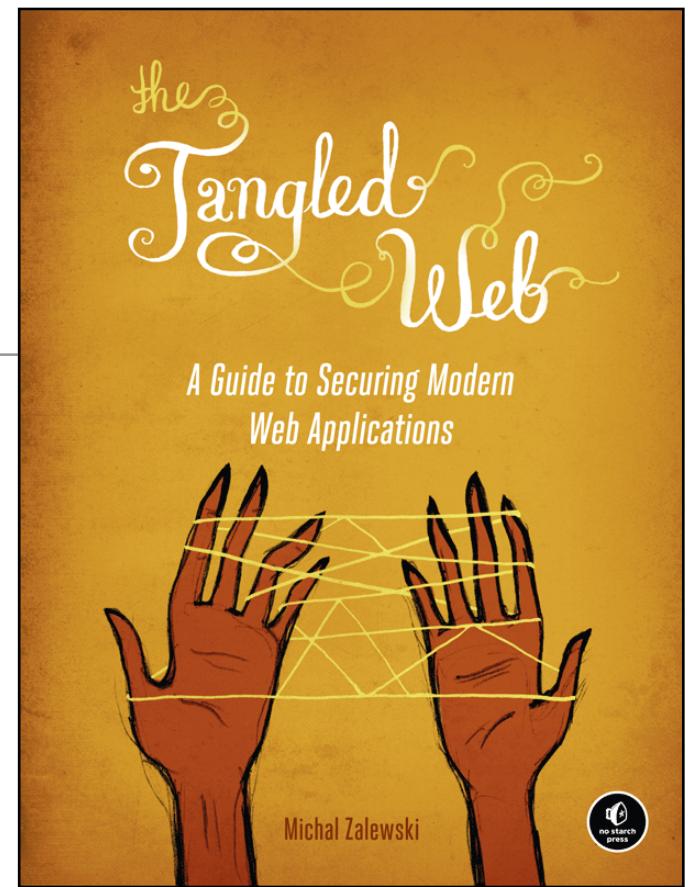
Disable Cross-Site Scripting (XSS)

Don't give too much information in error pages and exceptions

Stop unvalidated redirects and forwards

Always use https if your application has private data

Understand what browsers have to deal with



The Modern **JVM** Web Developer

Starts with Fast Hardware

Uses IntelliJ IDEA



2013 Jolt Awards Coding Tools

Leverages jQuery, HTML5, and CSS3

Creates High Performance Web Sites

For Mobile Devices, in the Cloud

And cares about Security



Staying Modern

Read

Attend Conferences

Submit a talk!

Write

Do

Get Paid ★

Open Source



Questions?

Contact Me!

 <http://raibledesigns.com>

 [@mraible](https://twitter.com/mraible)

Presentations

 <http://slideshare.net/mraible>

Code

 <http://github.com/mraible>

