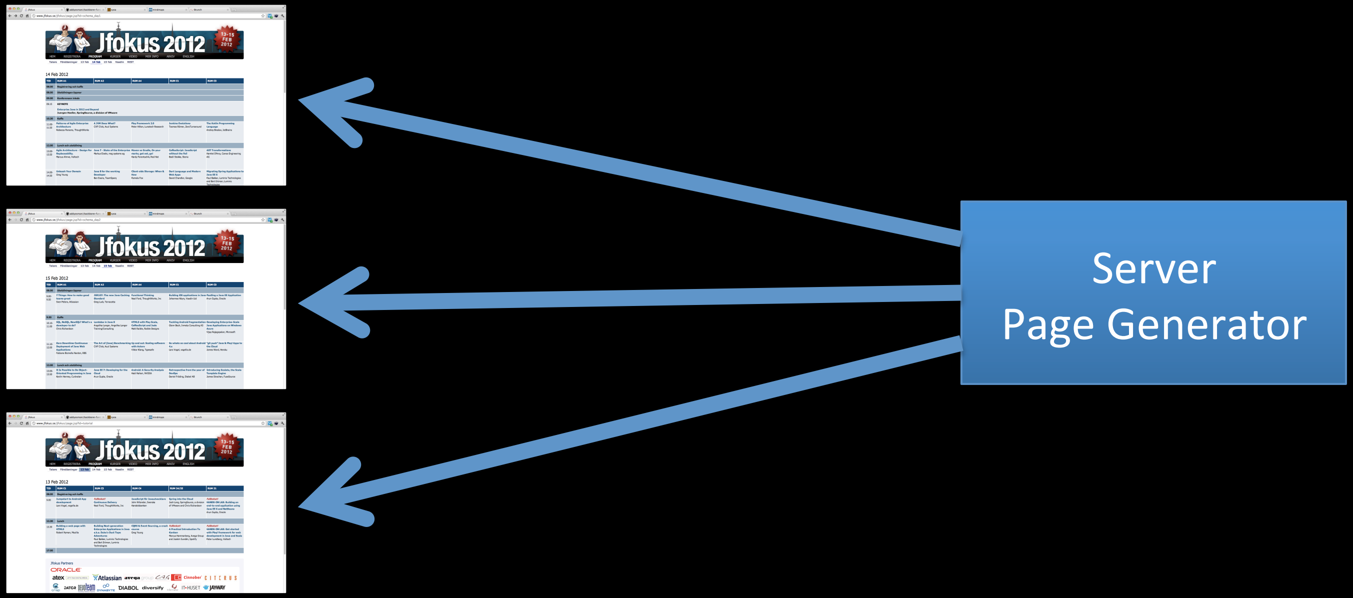
JavaScript Bonanza

The modern developer story

Björn Ekengren
@ekengren
Diversify
Javascriptpatterns.org

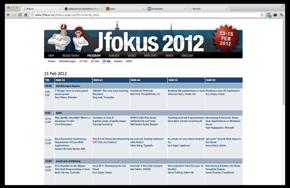
Turrican
MT.

Server generated content



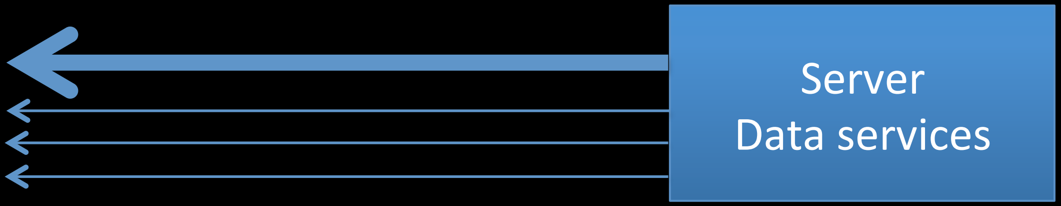
Server
Page Generator

Application generated content



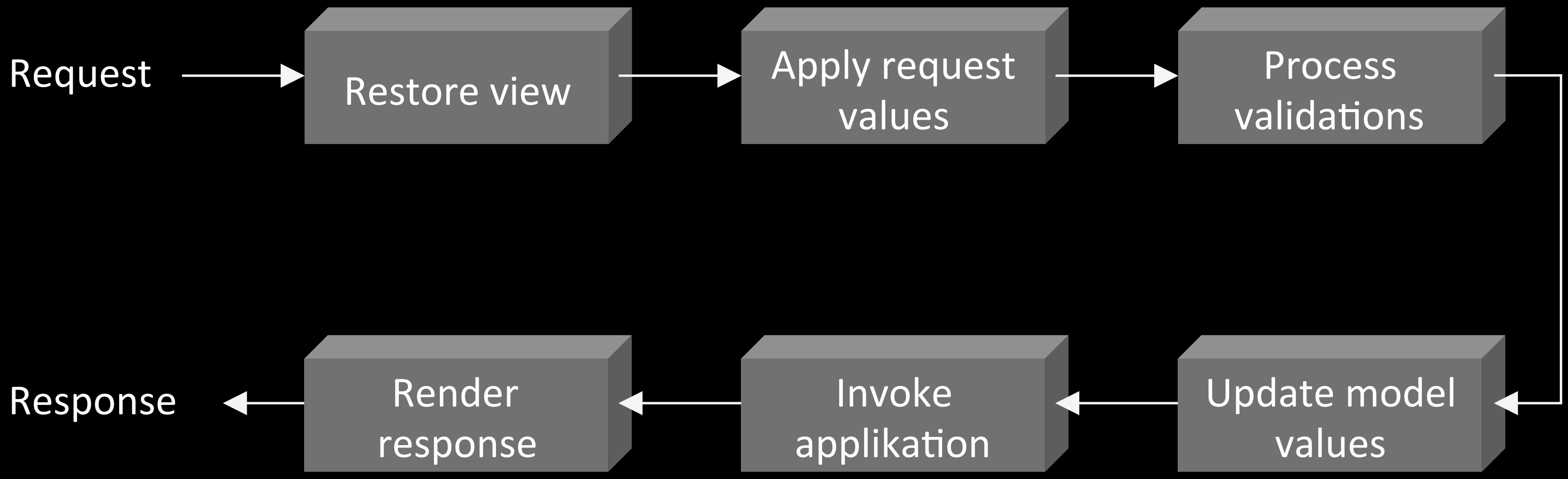
15 Feb 2012

TID	RUM A1	RUM A2	RUM A4	RUM C1	RUM C3
08.00	Utställningen öppnar				
9.00-9.50	7 Things: How to make good teams great Sven Peters, Atlassian	JSR107: The new Java Caching Standard Greg Luck, Terracotta	Functional Thinking Neal Ford, ThoughtWorks, Inc	Building iOS applications in Java Johannes Häyry, Vaadin Ltd	PaaSing a Java EE Application Arun Gupta, Oracle
9.50	Kaffe				
10.10-11.00	SQL, NoSQL, NewSQL? What's a developer to do? Chris Richardson	Lambdas in Java 8 Angelika Langer, Angelika Langer Training/Consulting	HTML5 with Play Scala, CoffeeScript and Jade Matt Raible, Raible Designs	Tackling Android fragmentation Glenn Bech, Inmeta Consulting AS	Developing Enterprise-Scale Java Applications on Windows Azure Vijay Rajagopalan, Microsoft
11.10-12.00	Zero Downtime Continuous Deployment of Java Web Applications Fabiane Bizinella Nardon, RBS	The Art of (Java) Benchmarking Cliff Click, Azul Systems	Up and out: Scaling software with Actors Viktor Klang, Typesafe	So whats so cool about Android 4.x Lars Vogel, vogella.de	"git push" Java & Play! Apps to the Cloud James Ward, Heroku
12.00	Lunch och utställning				
13.00-13.50	It Is Possible to Do Object-Oriented Programming in Java Kevlin Henney, Curbralan	Java EE 7: Developing for the Cloud Arun Gupta, Oracle	Android: A Security Analysis Hadi Nahari, NVIDIA	Retrospective from the year of DevOps Daniel Fröding, Diabol AB	Introducing Scalate, the Scala Template Engine James Strachan, FuseSource



Server
Data services

JSF Request Lifecycle



Request

Restore view

Apply request values

Process validations

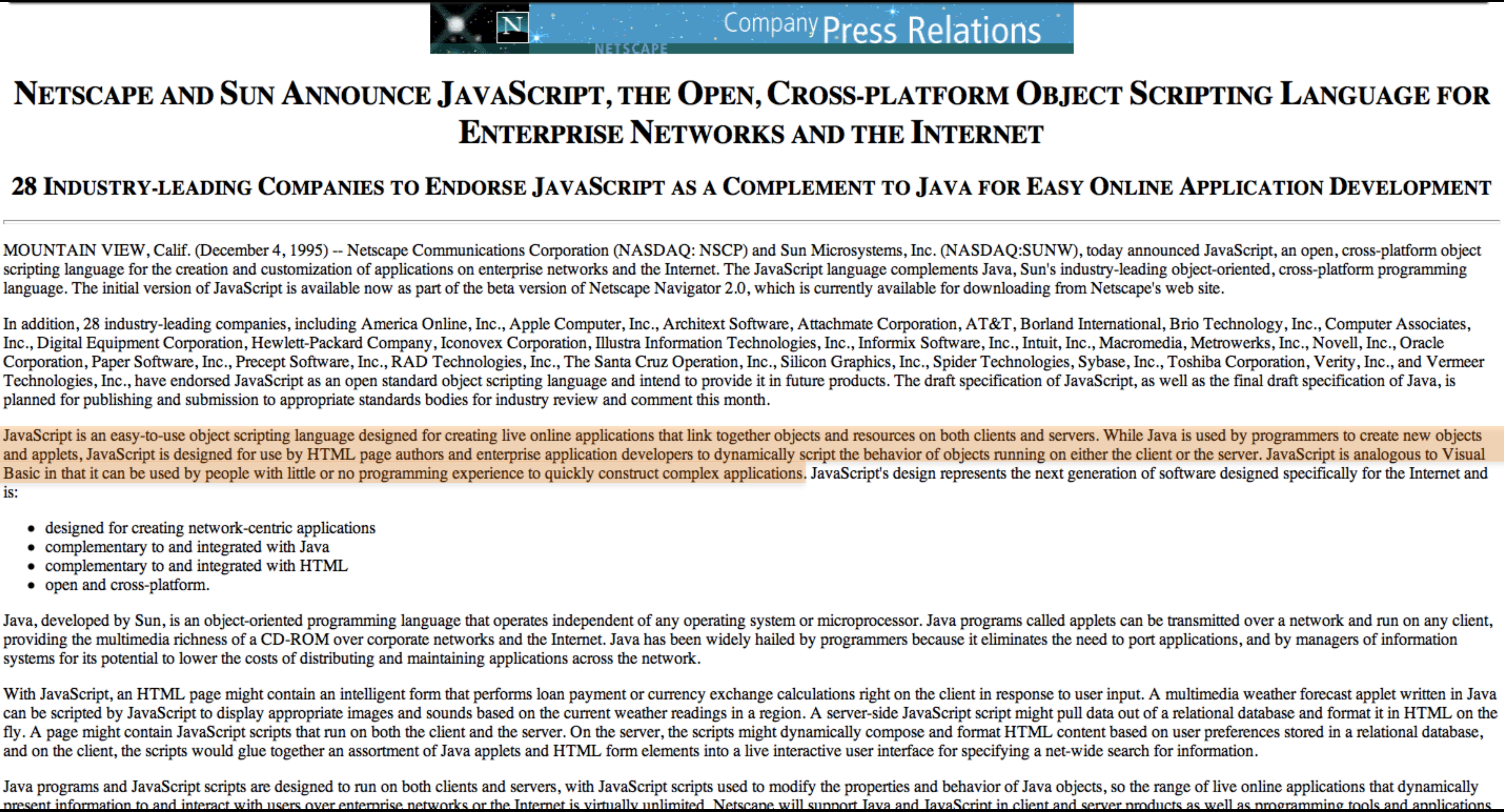
Update model values

Invoke applikation

Render response

Response

History



Netscape and Sun Announce JavaScript, the Open, Cross-platform Object Scripting Language for Enterprise Networks and the Internet

28 Industry-leading Companies to Endorse JavaScript as a Complement to Java for Easy Online Application Development

MOUNTAIN VIEW, Calif. (December 4, 1995) -- Netscape Communications Corporation (NASDAQ: NSCP) and Sun Microsystems, Inc. (NASDAQ:SUNW), today announced JavaScript, an open, cross-platform object scripting language for the creation and customization of applications on enterprise networks and the Internet. The JavaScript language complements Java, Sun's industry-leading object-oriented, cross-platform programming language. The initial version of JavaScript is available now as part of the beta version of Netscape Navigator 2.0, which is currently available for downloading from Netscape's web site.

In addition, 28 industry-leading companies, including America Online, Inc., Apple Computer, Inc., Architext Software, Attachmate Corporation, AT&T, Borland International, Brio Technology, Inc., Computer Associates, Inc., Digital Equipment Corporation, Hewlett-Packard Company, Iconovex Corporation, Illustra Information Technologies, Inc., Informix Software, Inc., Intuit, Inc., Macromedia, Metrowerks, Inc., Novell, Inc., Oracle Corporation, Paper Software, Inc., Precept Software, Inc., RAD Technologies, Inc., The Santa Cruz Operation, Inc., Silicon Graphics, Inc., Spider Technologies, Sybase, Inc., Toshiba Corporation, Verity, Inc., and Vermeer Technologies, Inc., have endorsed JavaScript as an open standard object scripting language and intend to provide it in future products. The draft specification of JavaScript, as well as the final draft specification of Java, is planned for publishing and submission to appropriate standards bodies for industry review and comment this month.

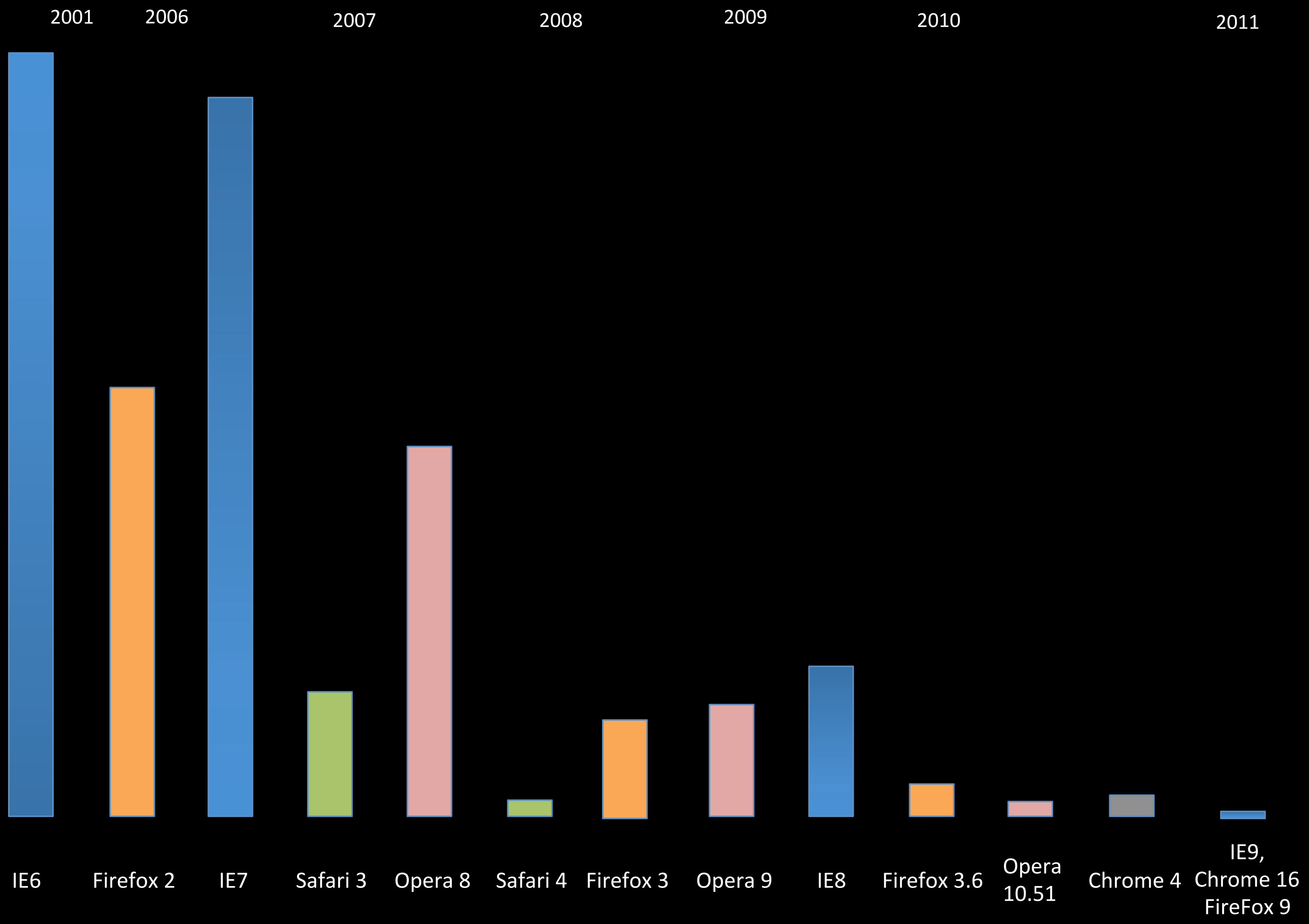
JavaScript is an easy-to-use object scripting language designed for creating live online applications that link together objects and resources on both clients and servers. While Java is used by programmers to create new objects and applets, JavaScript is designed for use by HTML page authors and enterprise application developers to dynamically script the behavior of objects running on either the client or the server. JavaScript is analogous to Visual Basic in that it can be used by people with little or no programming experience to quickly construct complex applications. JavaScript's design represents the next generation of software designed specifically for the Internet and is:

- designed for creating network-centric applications
- complementary to and integrated with Java
- complementary to and integrated with HTML
- open and cross-platform.

Java, developed by Sun, is an object-oriented programming language that operates independent of any operating system or microprocessor. Java programs called applets can be transmitted over a network and run on any client, providing the multimedia richness of a CD-ROM over corporate networks and the Internet. Java has been widely hailed by programmers because it eliminates the need to port applications, and by managers of information systems for its potential to lower the costs of distributing and maintaining applications across the network.

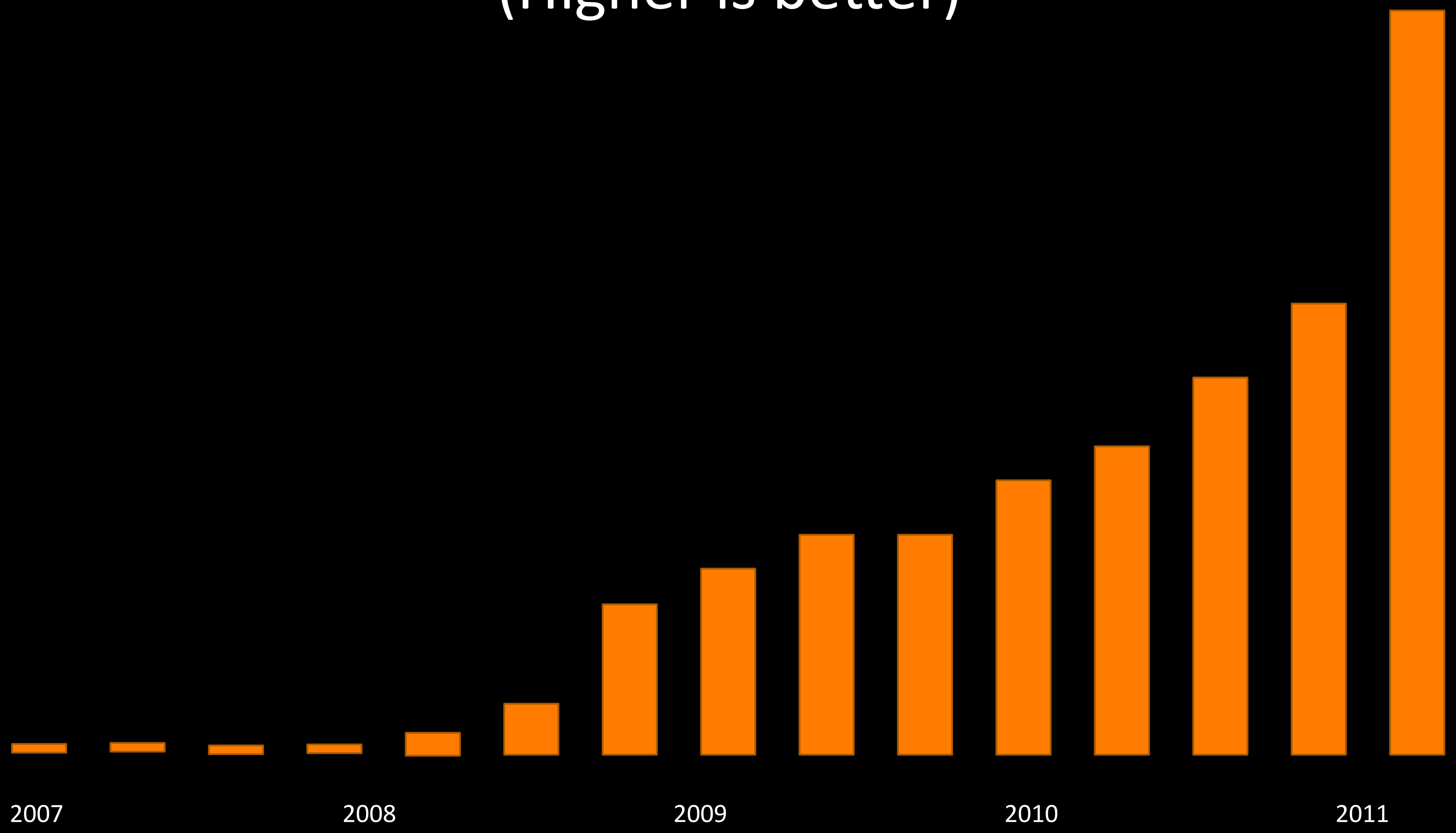
With JavaScript, an HTML page might contain an intelligent form that performs loan payment or currency exchange calculations right on the client in response to user input. A multimedia weather forecast applet written in Java can be scripted by JavaScript to display appropriate images and sounds based on the current weather readings in a region. A server-side JavaScript script might pull data out of a relational database and format it in HTML on the fly. A page might contain JavaScript scripts that run on both the client and the server. On the server, the scripts might dynamically compose and format HTML content based on user preferences stored in a relational database, and on the client, the scripts would glue together an assortment of Java applets and HTML form elements into a live interactive user interface for specifying a net-wide search for information.

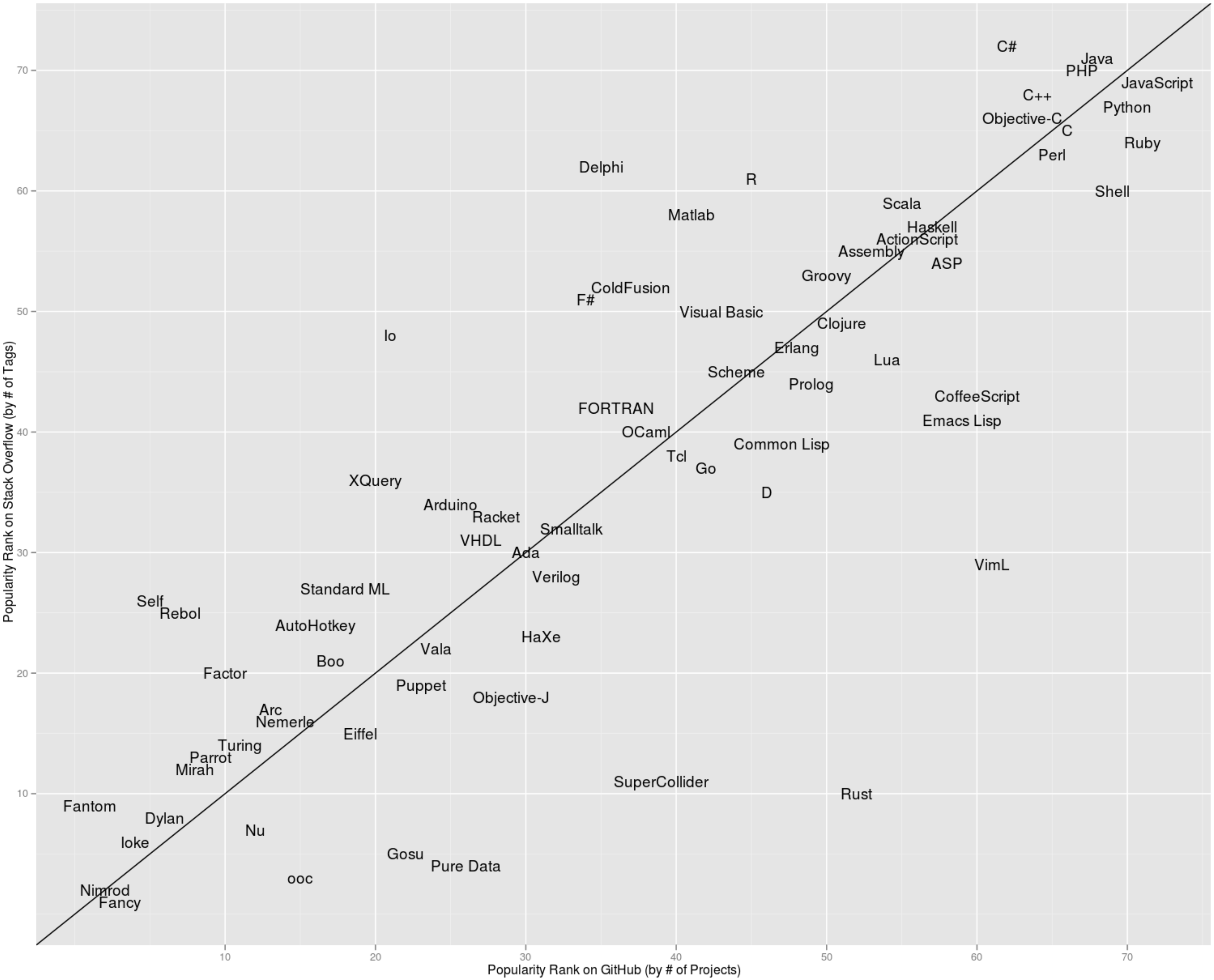
Java programs and JavaScript scripts are designed to run on both clients and servers, with JavaScript scripts used to modify the properties and behavior of Java objects, so the range of live online applications that dynamically present information to and interact with users over enterprise networks or the Internet is virtually unlimited. Netscape will support Java and JavaScript in client and server products as well as programming tools and applications


2001
2006
2007
2008
2009
2010
2011
IE6
Firefox 2
IE7
Safari 3
Opera 8
Safari 4
Firefox 3
Opera 9
IE8
Firefox 3.6
Opera 10.51
Chrome 4
IE9, Chrome 16 FireFox 9

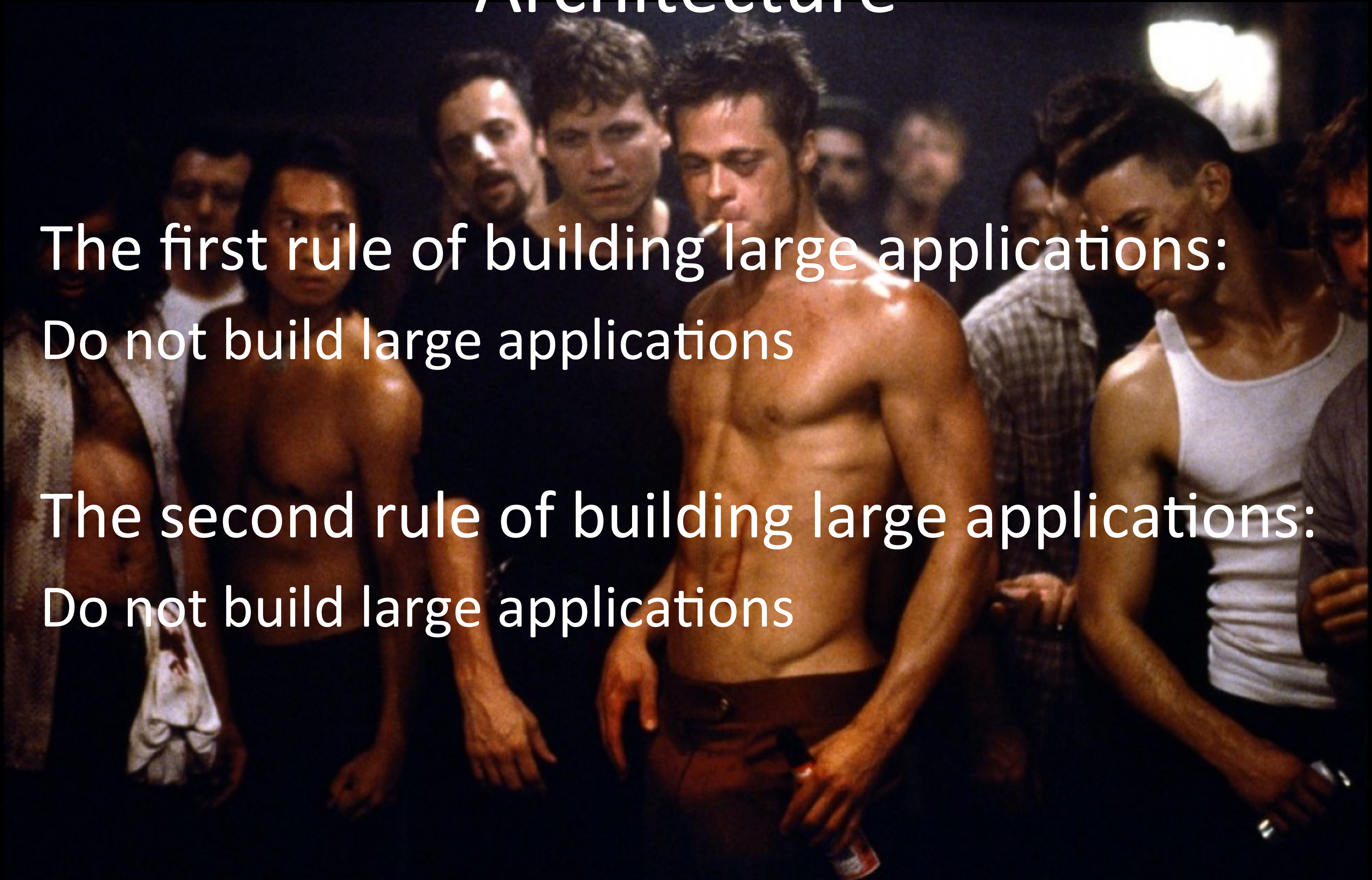
JavaScript Performance over time

(Higher is better)




C#
Java
PHP
JavaScript
C++
Python
Objective-C
C
Ruby
Perl
Delphi
R
Shell
Scala
Matlab
Haskell
ActionScript
Assembly
ASP
Groovy
ColdFusion
F#
Visual Basic
Clojure
Io
Erlang
Lua
Scheme
Prolog
CoffeeScript
FORTRAN
Emacs Lisp
OCaml
Common Lisp
Tcl
Go
XQuery
D
Arduino
Racket
Smalltalk
VHDL
Ada
VimL
Verilog
Standard ML
Self
Rebol
AutoHotkey
HaXe
Vala
Boo
Factor
Puppet
Objective-J
Arc
Nemerle
Eiffel
Turing
Parrot
Mirah
SuperCollider
Rust
Fantom
Dylan
Nu
Ioke
Gosu
Pure Data
ooc
Nimrod
Fancy
Popularity Rank on Stack Overflow (by # of Tags)
70
60
50
40
30
20
10
Popularity Rank on GitHub (by # of Projects)
10
20
30
40
50
60
70

Architecture



The first rule of building large applications:

Do not build large applications

The second rule of building large applications:

Do not build large applications

Architecture cont…

- Basic framework for working with the DOM
- Patterns
- MVC
- Module communication
- Templating
- Dependency management
- Test
- Building

DOM Frameworks

- jQuery
- Dojo
- Prototype
- YUI
- MooTools
- ExtJS
- Cappucino
- QooxDoo
- FuseJS

Patterns

- MVC
- Observer (pubsub)
- Module
- Façade
- Mediator

MVC

- Backbone.js
- JavaScriptMVC
- Spine.js
- EmberJS
- Serenade
- KnockoutJS (MVVM)
- Sammy.js

Backbone

```
var Photo = Backbone.Model.extend({

    // Default attributes for the photo
    defaults: {
        src: "placeholder.jpg",
        caption: "A default image",
        viewed: false
    },

    // Ensure that each photo created has an `src`.
    initialize: function() {
        this.set({"src": this.defaults.src});
    }
});
```

Module Communication

- jsSignals
- PubSubJS
- Ben Almans pub/sub
- Peter Higgins pub/sub
- Custom Events

jsSignals

```
//store local reference for brevity
var Signal = signals.Signal;

//custom object that dispatch signals
var myObject = {
    started : new Signal(),
    stopped : new Signal()
};

function onStarted(param1, param2){
    alert(param1 + param2);
}

myObject.started.add(onStarted); //add listener
myObject.started.dispatch('foo', 'bar'); //dispatch
```

Templating

- Underscore
- jQuery templating
- Mustache
- Handlebars
- Dust.js
- ICanHaz.js
- PURE
- MicroTemplates
- Closure Templates
- jQuery.view

Underscore

```
var photoTemplate =
‘<li class="photo”>
    <h2><%= caption %></h2>
    <img class="source" src="<%= src %>"/>
    <div class="meta-data”>
        <%= metadata %>
    </div>
</li>’
```

```
photoEl.innerHTML = _.template('photoTemplate',
                               {src: photoModel.getSrc()});
```

Dependency Management

- AMD – Asynchronous Module Definition

- RequireJS
- curl.js
- StealJS
- JSL Script Loader
- Bootstrap

AMD

```
define(
     ‘pnyxtr’,
    [foo, bar],
    function(foo, bar){
        var module = {};
        module.doFoo = foo.doFooStuff();
        module.doBar = bar.doBarStuff();
        return module;
    }
};
```

id

dependencies

factory

RequireJS

```
require([”pnyxtr", ”logger”], function($) {
    //the pnyxtr and logger plugins
    //have been loaded.
    $(function() {
        logger.log(pnyxtr.doFoo());
    });
});
```

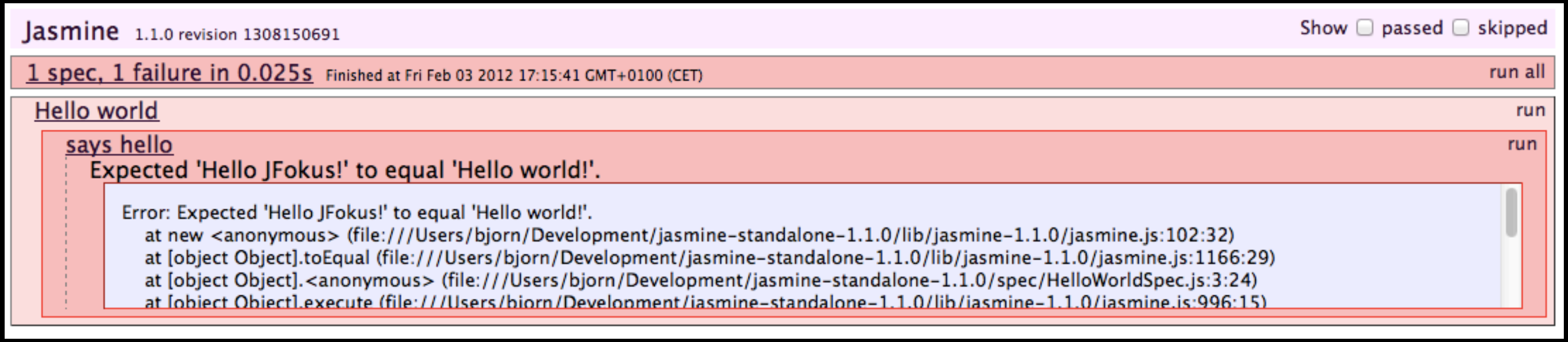
Test

- Qunit
- Jasmine
- FireUnit
- Crosscheck
- JSSpec
- jsTestDriver
- WebDriver
- FuncUnit
- YUI Test
- Selenium
- Google JS Test
- Mocha

Jasmine

```
function helloWorld() {
        return "Hello JFokus!";
}
```

```
describe("Hello world", function() {
        it("says hello", function() {
                expect(helloWorld()).toEqual("Hello world!");
        });
});
```



```
1 spec, 1 failure in 0.025s Finished at Fri Feb 03 2012 17:15:41 GMT+0100 (CET)
Hello world
  says hello
    Expected 'Hello JFokus!' to equal 'Hello world!'.
      Error: Expected 'Hello JFokus!' to equal 'Hello world!'.
          at new <anonymous> (file:///Users/bjorn/Development/jasmine-standalone-1.1.0/lib/jasmine-1.1.0/jasmine.js:102:32)
          at [object Object].toEqual (file:///Users/bjorn/Development/jasmine-standalone-1.1.0/lib/jasmine-1.1.0/jasmine.js:1166:29)
          at [object Object].<anonymous> (file:///Users/bjorn/Development/jasmine-standalone-1.1.0/spec/HelloWorldSpec.js:3:24)
          at [object Object].execute (file:///Users/bjorn/Development/jasmine-standalone-1.1.0/lib/jasmine-1.1.0/jasmine.js:996:15)
```

Building

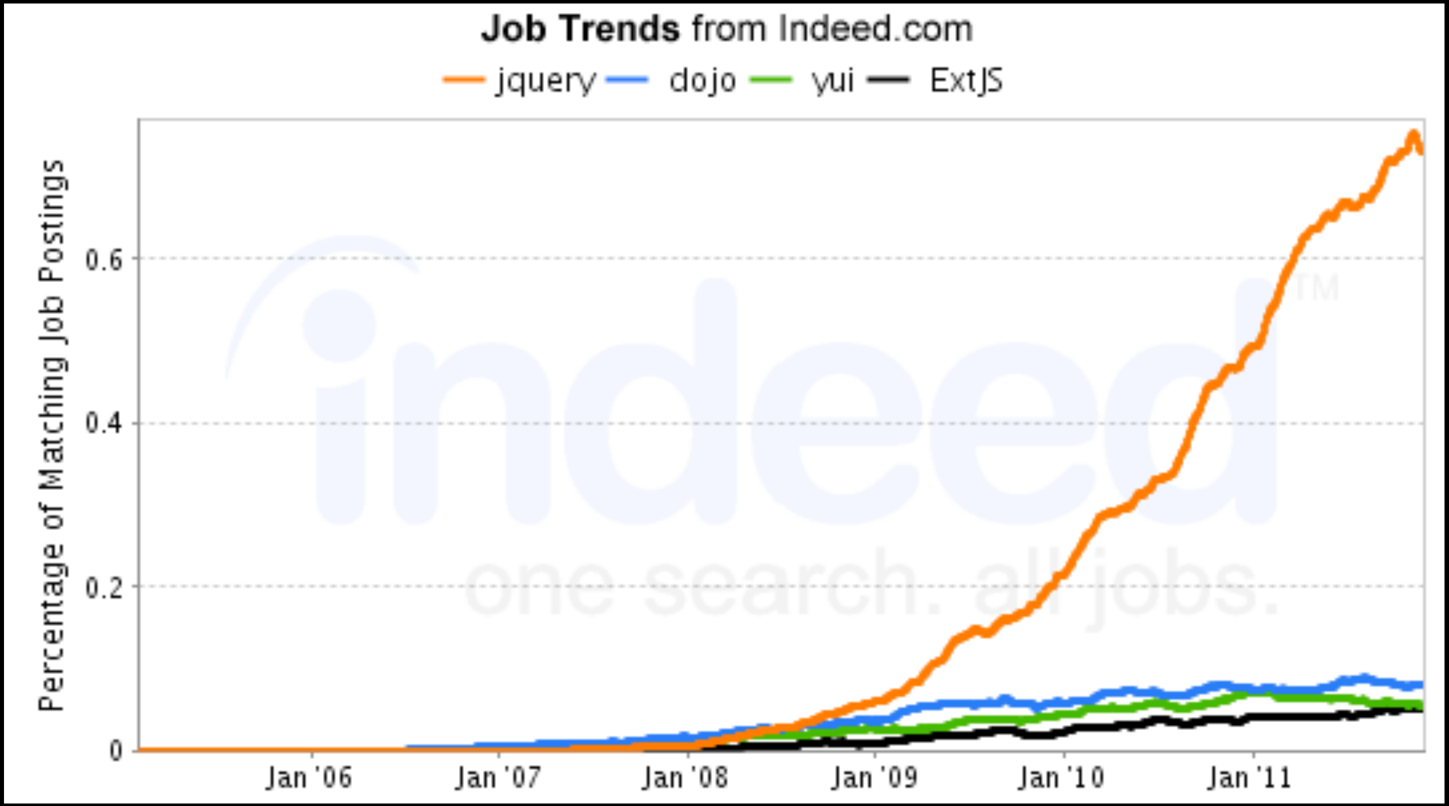
- Jake
- Smasher
- YUI Compressor
- Sprockets
- Closure Compiler
- UglifyJS
- ShrinkSafe
- JSMin
- Grunt.js

Other

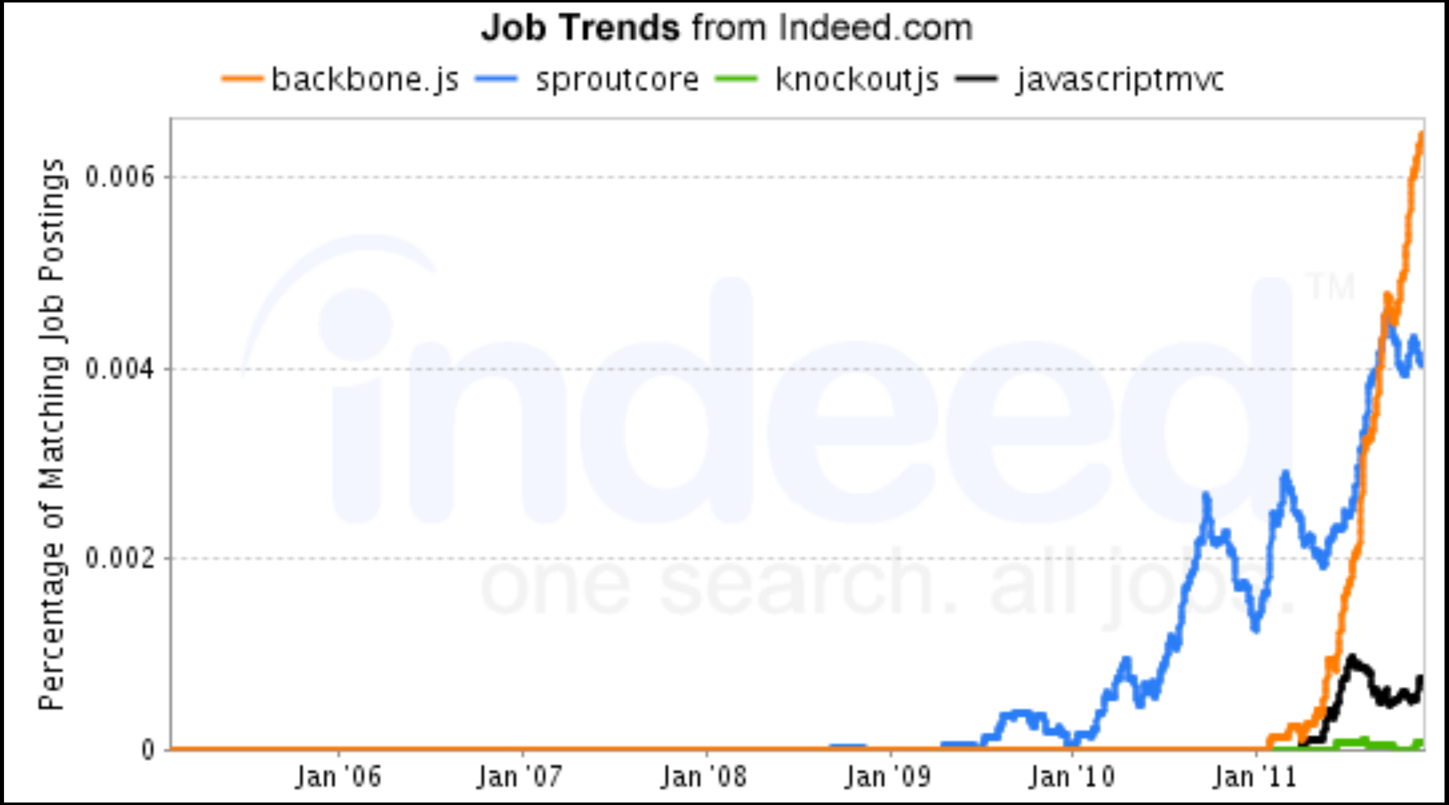
- Feature detection
  - Modernizr
  - Has.js
- LocalStorage
  - Store.js
  - Persist.JS
- CSS
  - Less
  - Sass

What to choose?

What to choose?



What to choose?



What to choose?

- Complete packages
  - Brunch
  - Wakanda
- TodoMVC

Tools

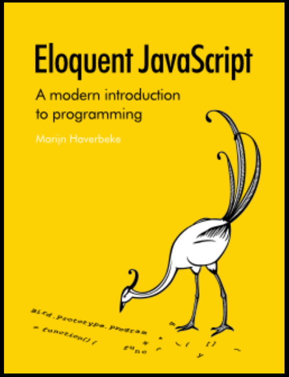
- IDEs
  - Eclipse
  - Netbeans
  - Webstorm
  - Cloud9
  - Textmate
  - Visual Studio
- Browser tools
  - Firebug
  - Chrome developer tools
  - Yslow
- Code analysis
  - JSLint
  - CSSLint
  - CoffeeLint

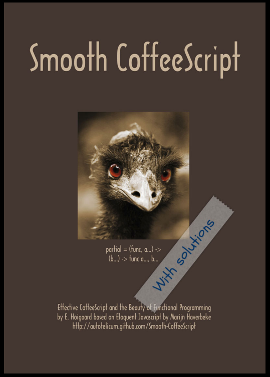
“To err is human, but to really foul things up you need a computer.”

(Paul Ehrlich)

Advice on learning JavaScript Level 1

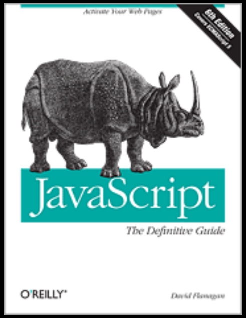
- Get an understanding of
  - Functions
  - Scope
  - 'this'
  - Falsy and truthy values
  - Indexing of arrays vs object attributes
- Get an IDE that support you
- Read




Smooth CoffeeScript
partial = (func, a...) ->
(b...) -> func a..., b...
With solutions
Effective CoffeeScript and the Beauty of Functional Programming
by E. Hoigaard based on Eloquent Javascript by Marijn Haverbeke
http://autotelicum.github.com/Smooth-CoffeeScript

Advice on learning JavaScript Level 2

- Get an understanding of
  - Asynchronous function execution
  - JSON notation
  - Closures
  - Basic patterns
  - AJAX
- Learn a DOM framework
  - jQuery, Dojo, ExtJS, YUI, Mootools …
- Read



Professional
JavaScript
for Web Developers

Advice on learning JavaScript Level 3

- Get an understanding of
  - Prototypical inheritance/prototype chain
  - Partially applied functions
  - Function arguments variable
  - Knowing what tools to use to build robust cross browser code
- Learn a unit test framework
  - Qunit, Jasmine, JSTestDriver, Mocha
- Read

High Performance
JavaScript
O'REILLY
YAHOO! PRESS
Nicholas C. Zakas

Object-Oriented JavaScript
Create scalable, reusable high-quality JavaScript applications, and libraries
Stoyan Stefanov
PACKT

Advice on learning JavaScript Level 4

- Structure code to build large applications
- Learn an MVC framework
  - Backbone, JavaScriptMVC, EmberJS, KnockoutJS
- Read

Build Better Applications with Coding and Design Patterns
JavaScript Patterns
O'REILLY
YAHOO! PRESS
Stoyan Stefanov

Secrets of the JavaScript Ninja
MEAP
John Resig
Bear Bibeault
MANNING

Javascriptpatterns.org/ladder.html

Summary

- Separation of concerns
- Better user experience
- Easier do debug
- Get to know the language

Questions

Cool stuff

- Mindmaps
- JSLinux
- JSNes
- Mugeda
- Mr doob
- Languages that compile to JS
- Ro.me
- Impress.js