

JavaScript Bingo

Pete Campbell

pete@sumirolabs.com

@sumirolabs

github.com/campbell

Welcome To The USA!

Welcome To The USA!

(His name is not Roy)

Welcome To The USA!

(His name is not Roy)

(Well played, sir!)

The 90's Were Good To Me

(MicroStrategy)

The 90's Were Good To Me

(MicroStrategy)

(He once gave a 90 minute speech
about his summer vacation)

The 90's Were Good To Me

(MicroStrategy)

(He once gave a 90 minute speech
about his summer vacation)

Ninety

The 90's Were Good To Me

(MicroStrategy)

(He once gave a 90 minute speech
about his summer vacation)

Ninety

Minutes....

MicroStrategy Stock Chart



self.inspect (this.toString?)

- ★ Started in chip design at IBM
- ★ Web developer since 2000
- ★ Ruby since 2008
- ★ Freelance since 2009

Confession

Confession

★ My JavaScript code stinks...

Confession

- ★ My JavaScript code stinks...
- ★ Your code probably stinks too...

Confession

- ★ My JavaScript code stinks...
- ★ Your code probably stinks too...
- ★ But its not your fault...

Confession

- ★ My JavaScript code stinks
- ★ Your code probably stinks too...
- ★ But its not your fault...

Its your parents fault

Parents Just Don't Understand

- ★ JavaScript's parents



Parents Just Don't Understand

- ★ Netscape and Microsoft intentionally disagreed



Parents Just Don't Understand

- ★ Netscape and Microsoft intentionally disagreed

Events bubble
down!



Parents Just Don't Understand

- ★ Netscape and Microsoft intentionally disagreed

You fool! Events
bubble up!



Events bubble
down!



Resolution



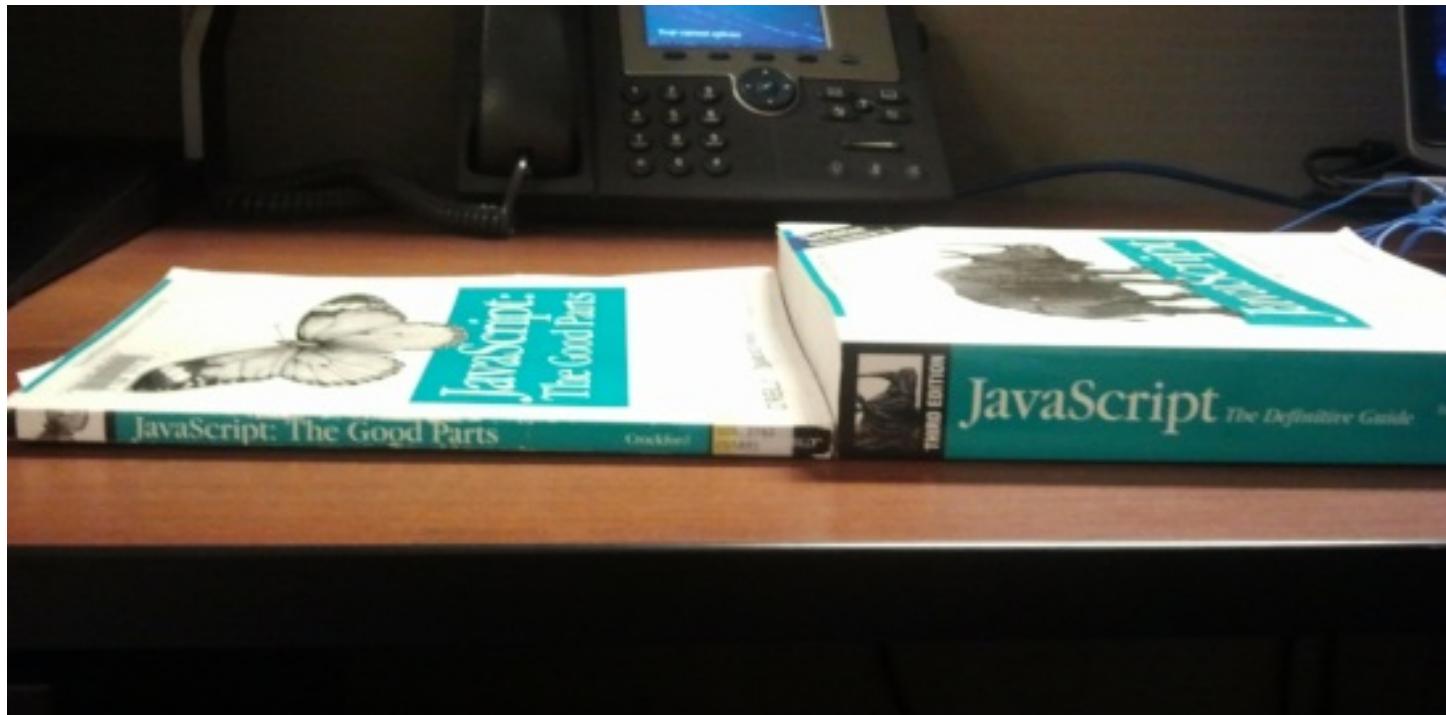
Salvation

Salvation

Douglas Crockford, *JavaScript: The Good Parts*

Salvation

Douglas Crockford, *JavaScript: The Good Parts*



I Believe I Can Fly...

- ★ I knew how to write better JavaScript
- ★ In the JavaScript community, there was lots of info about good practices
- ★ This would be just like learning Ruby

I Believe I Can Fly...

- ★ I knew how to write better JavaScript
- ★ In the JavaScript community, there was lots of info about good practices
- ★ This would be just like learning Ruby
- ★ That is a problem...

TMI!

- ★ Too much information!
- ★ Too many libraries, projects, terms
- ★ What is important?
- ★ Why do I care?

Why Are We Here?

- ★ JavaScript jumpstart
- ★ Describe* many JavaScript & related frameworks, libraries, terms, thingys
- ★ Help you figure out what you should be watching in the JS world
- ★ Try to keep you from slipping into a food-coma from lunch

* Guaranteed to be mostly-correct

JavaScript Bingo Rules

- ★ Get a card, cross off a term when I say it (wait for it...)
- ★ When you get 5 in a row, scream *BINGO!* loud enough to wake the person next to you from their food-coma
- ★ Get a prize*
- ★ In case of a tie, there will be a dance-off...

* Warning: Recipient may not consider it a prize

Our Journey

- ★ Server
- ★ Browser
- ★ Libraries
- ★ Neat Stuff
- ★ Bonus Round Dance Party

Server

★ Node.js

- Wins award for the most hype
- “Non-blocking”, so more efficient use of resources
- Powered by V8
- Makes asynchronous, event-driven programming much simpler

★ NPM – Node Package Manager

- Install, manage & publish Node programs

Server

★ Connect

- Middleware HTTP framework (logger, CSRF, cookie & JSON parser...)

★ Express

- Lightweight web development framework
- Inspired by Sinatra

★ Geddy, Locomotive, RailwayJS

- Frameworks that look like Rails

JavaScript Engines

★ V8

- From Google
- Compiles to machine-code

★ SpiderMonkey

- From Mozilla
- Compiles to byte code
- Hot-spots compiled to machine code

★ TraceMonkey

- Compiles to machine code

JavaScript Engines

★ Rhino

- Mozilla
- Java implementation, intended for embedding

★ PhantomJS

- Headless WebKit with a JS API
- Useful for testing

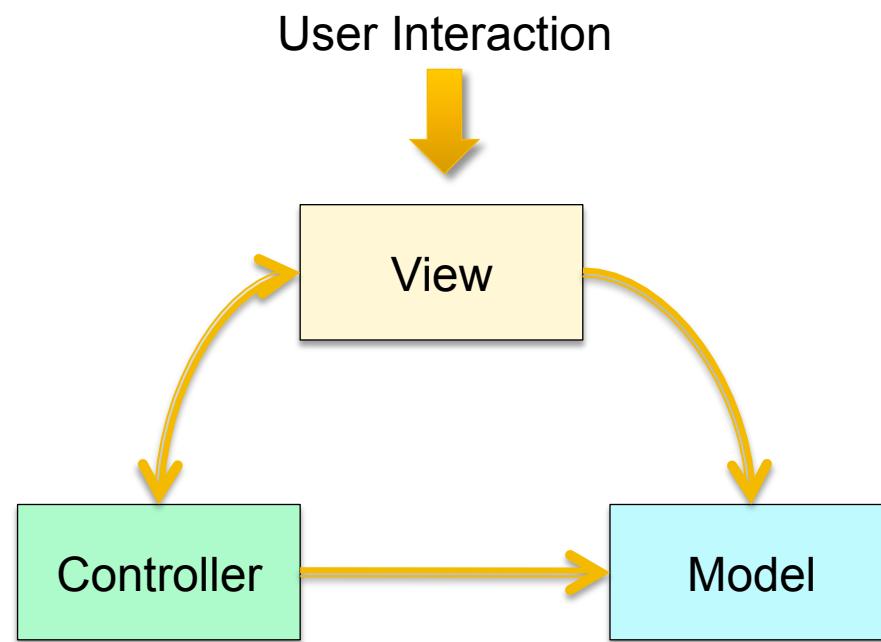
Browser

- ★ Design Patterns
- ★ Frameworks
- ★ HTML5

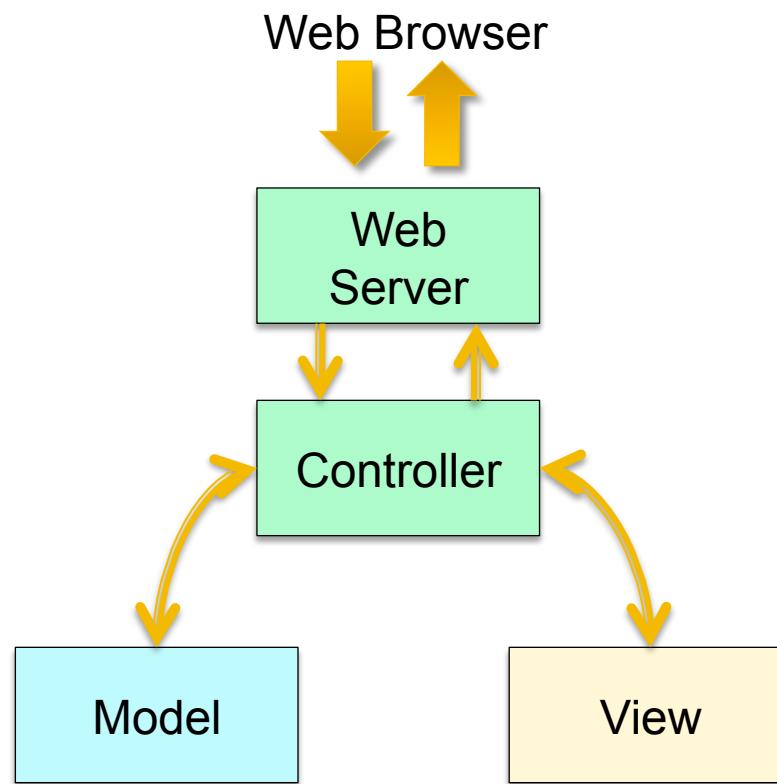
Frameworks <3 Design Patterns

- ★ Over 20 JavaScript frameworks...
(21...22...23...)
- ★ Very hard to compare at a glance
- ★ Design patterns help us understand

Model-View-Controller

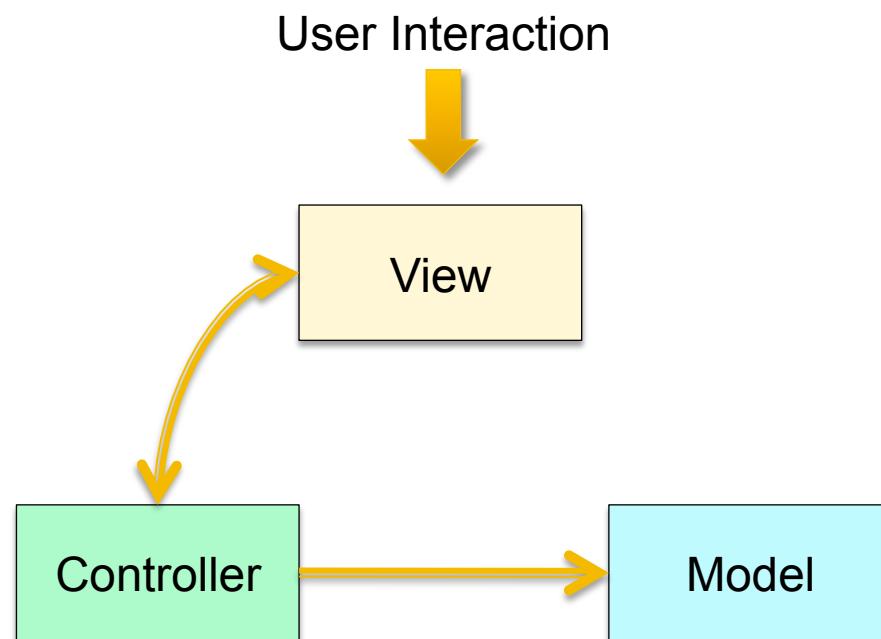


Model2



Model-View-Presenter

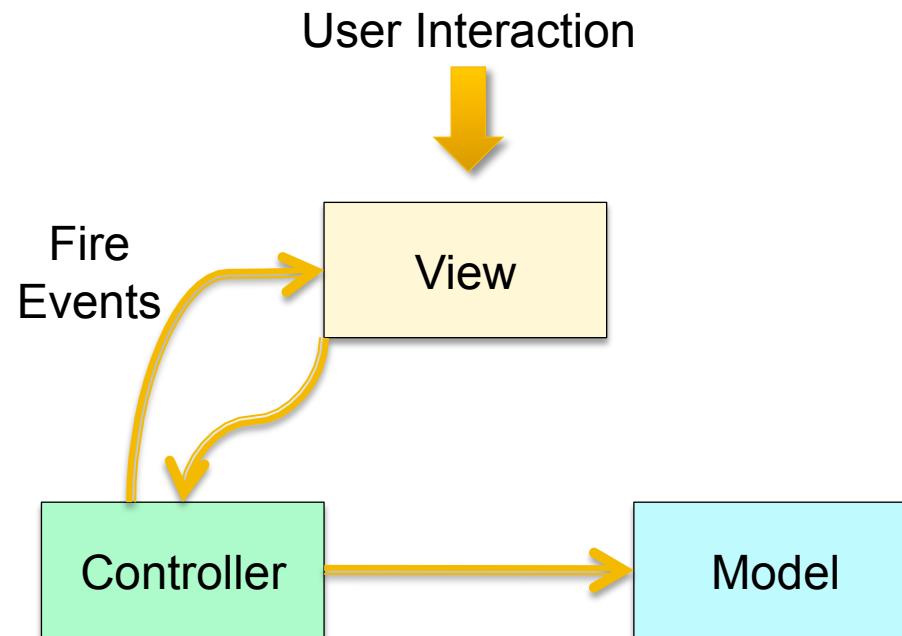
- ★ No View-Model Connection
- ★ Presenter updates the View via an API



MVVM

★ Model-View-ViewModel

- Almost identical to MVP
- Tight binding between view-model and model



Frameworks

★ Backbone (Jeremy Ashkenas)

- Lightweight (600 LOC)
- More of a library than a framework
- Very flexible, very DIY (e.g. templating)

★ Ember.JS (Yehuda Katz)

- Second-coming of Sproutcore
- Not lightweight (6000LOC?)
- Complete opinionated solution, harder to customize

Frameworks

★ JavaScriptMVC

- More mature,
jQueryMX

★ KnockoutJS

- MVVM, awkward
syntax

★ AngularJS

- Similar to Knockout,
easier syntax, more
testable

★ ExtJS4

- Sencha, lots of UI
widgets

★ Batman.js

- CoffeeScript, from
Shopify, developer
'happiness'

★ Spine.js

- Lightweight (500LOC),
CoffeeScript

Templates

★ MustacheJS

- Logic-less templates

★ HandlebarsJS

- Mustache with better options for larger projects

★ Jade

- Node templates, can be compiled for the browser
- HAML-ish

★ LiquidJS

- Port of Liquid safe templates from Shopify

Problem

- ★ We've got a framework
- ★ We've got templates
- ★ We've got 106 miles to Chicago, a full tank of gas, its dark out...oh wait....
- ★ We've got lotsa files...how do we get them to the client?

Script Loaders & Packagers

★ LabJS

- Dependency management, non-blocking

★ ControlJS

- Minimal documentation

★ Script.JS

- Lightweight

★ StealJS

- Component of JavascriptMVC
- Dependency, package management, more...

★ Ender

- Client-side packager, command-line, like NPM for browsers

New Problem

- ★ We've got lots of files...
- ★ We've got lots of code...
- ★ Now we've got lots of name collisions

Module Pattern

- ★ Use a closure to hide your privates
- ★ Expose only what you want

```
var MyModule = function() {  
    var private_counter = 0;  
    return {  
        count: function() {  
            private_counter += 1;  
            return "Count is " + private_counter;  
        }  
    }  
  
    m = MyModule();  
    m.count #=> "Count is 1"  
    m.count #=> "Count is 2"
```

Module Pattern Warz!

★ CommonJS (actually CommonJS Modules)

- Aimed at server-side applications, doesn't support asynchronous loading, dependency resolution
- API exported via a variable called 'exports'
- Modules imported using 'require'

Parent.js

```
exports.says = function() { return 'Hi Mom!' }
```

Child.js

```
var child = require('parent.js')
child.says() //=> 'Hi Mom'
```

AMD (not that AMD)

★ Asynchronous Module Definition

- Aimed at the browser, provides dependency resolution
- Can load other types of resources (images, CSS...)
- ‘define’ function specifies name, dependencies, callback
- RequireJS, CurlJS

```
var get_started= function(mod1, mod2) { ... }
```

```
define('my_module', ['mod_x', 'mod_y'], get_started);
```

HTML5

★ Modernizr

- Feature detection, conditional loading

★ Polyfills

- JavaScript shims for missing features

Graphics

★ RaphaelJS

- SVG graphics, very responsive, great animations, sparse documentation

★ Paper.js

- Vector graphics scripting language using Canvas

★ FabricJS, EaselJS

- Libraries for working with Canvas

★ Processing.JS

- Port of the Processing visual programming language
- 3D, shading, lighting

CSS

★ Why CSS in JavaScript?

- Node apps can use it natively
- Compiled versions for client

★ Sass.js

★ Stylus

- newer version of Sass.js

★ Less.js

- Port of LessCSS

Libraries

★ UglifyJS

- Parses into an AST that can be manipulated (minify, beautify)
- Node / CommonJS

★ Socket.io

- Realtime sync with fallbacks (web sockets, AJAX, flash, JSONP...)
- Lots of browsers (IE5+)

I <3 Ruby Libraries

★ UnderscoreJS

- 60+ features that will make Rubyists happy (map, select, each...)
- Included with Backbone

★ Classify.JS

- Rubyish modules, classes, inheritance, type-checking

★ Classified.JS

- Rubyish extensions, similar to underscore

Testing

★ Jasmine

- RSpec-like BDD, standalone / CI / CLI, not well integrated with Node

★ Mocha

- Node-based, similar to Jasmine
- Combine with an assertion library (Chai, Should.js, Expect.js)

★ Zombie.js

- Headless browser for testing client-side code
- Fast, lightweight, requires Node (V8)

Game Engines

★ ImpactJS

- commercial, can publish to Apple App store

★ LimeJS

- Box2D physics engine

★ CraftyJS

- Lightweight (14.5kb)

★ Spaceport.io

- Mobile gaming

★ RPG JS

- based on EaselJS Canvas library

★ Entity.js

- Ruby GEM, provides generators

Physics Engines

★ Box2DJS

- Port of Box2D

★ Bullet.js

- Port of Bullet 3D

★ Ammo.js

- Port of Bullet 3D via Emscripten

★ CannonJS

- 3D, lightweight

★ Microphysics.js

- very lightweight (500LOC)

★ Chipmunk.js

- Port of Chipmunk 2D

3D Graphics

★ Three.js

- Renders to Canvas, WebGL, SVG

★ C3DL (Canvas 3D Library)

- WebGL, provides math, scene & object classes

★ CopperLicht

- Imports a lot of 3D file formats

★ J3D

- Can use Unity3D models*

3D Graphics

- ★ Three.js
 - Renders to Canvas, WebGL
- ★ C3DL (Canvas 3D Library)
 - WebGL, provides math API
- ★ CopperLicht
 - Imports a lot of 3D file formats
- ★ J3D
 - Can use Unity3D models*



Free Unity3D License
<http://store.unity3d.com>

Transpilers

- ★ CoffeeScript

- ★ Pyjamas

- ★ Dart

- “Hello World” in only
17,259 LOC

- ★ GWT

- ★ ClojureScript

- ★ Emscripten

- LLVM to JS

Way too many to list here, Google ‘javascript transpilers’

Transpilers

- ★ CoffeeScript

- ★ Pyjamas

- ★ Dart

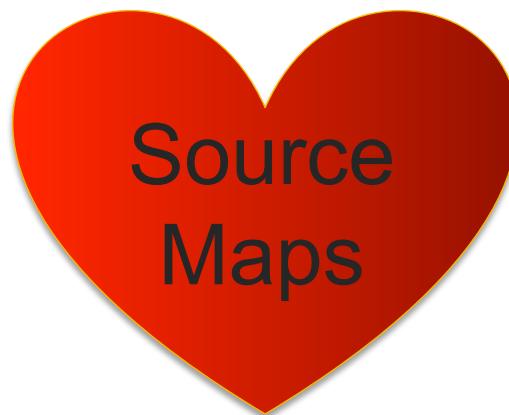
- “Hello World” in only
17,259 LOC

- ★ GWT

- ★ ClojureScript

- ★ Emscripten

- LLVM to JS



Other Thingys

★ Google Closure Tools

- JS optimizer, large library, templates

★ NaCl

- Google's Native Client, C++ in the browser

★ Haxe, Monkey

- Multiplatform languages, compilable to JS, Flash, C++ , Java (and then to iOS, Android...)
- Adds non-native features (type checking, classes, packages for JS)

How To Keep Up

- ★ JavaScript Weekly
- ★ @jsgoodies
- ★ @JavaScriptDaily
- ★ Javascript Jabber

Wake Up!

pete@sumirolabs.com

@sumirolabs

github.com/campbell/presentations

Thanks for
the prizes!



CustomInk
Design T-shirts Online