

FRONT END DEPENDENCY MANAGEMENT

LET'S HUG IT OUT, FRIENDS

Presented to you by [Joe Sepi](#) / [@joe_sepi](#)

I AM WHO I THINK I AM.

- VP of Engineering at [Novus.com](#) where we do financial analysis awesomely
- Previously worked at [The New York Times](#) doing dev, dev mgmt and dev relations
- I also consider myself something of a [punk](#)
- [@joe_sepi](#) on the twitter machine



Rene Auberjonois

WE'RE ALL IN THIS TOGETHER



HISTORY

Dojo, Modules/Transport/C, RunJS, Narwhal, etc...

WHAT ARE THE PROBLEMS WE ARE TRYING TO SOLVE?



```
<script src="..."></script><script src="..."></script><script src="..."></scr  
MyApp.something.whatever.gone.too.far('fail');  
this must load before that but not before
```


So those are the problems we are trying to solve, but what are the benefits of solving these problems?
performance, stability, sanity, reuse,

HOW DO WE GET THERE?

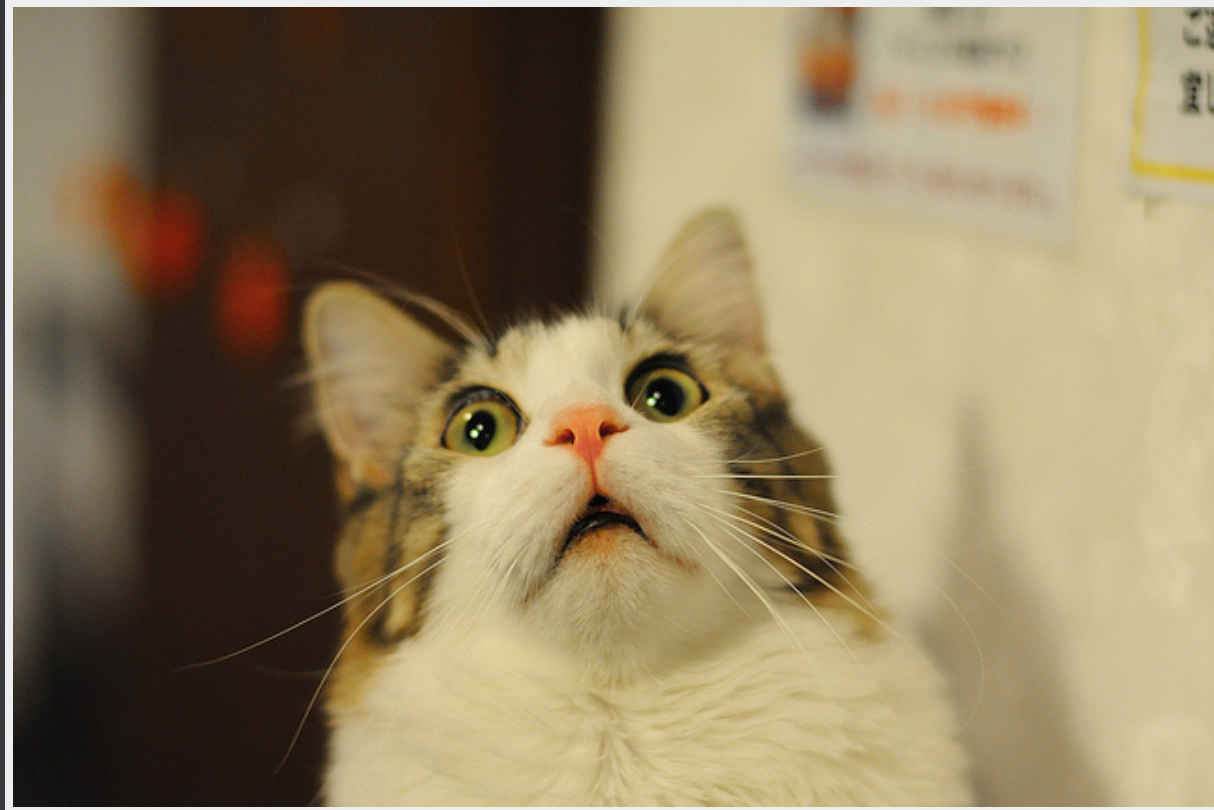


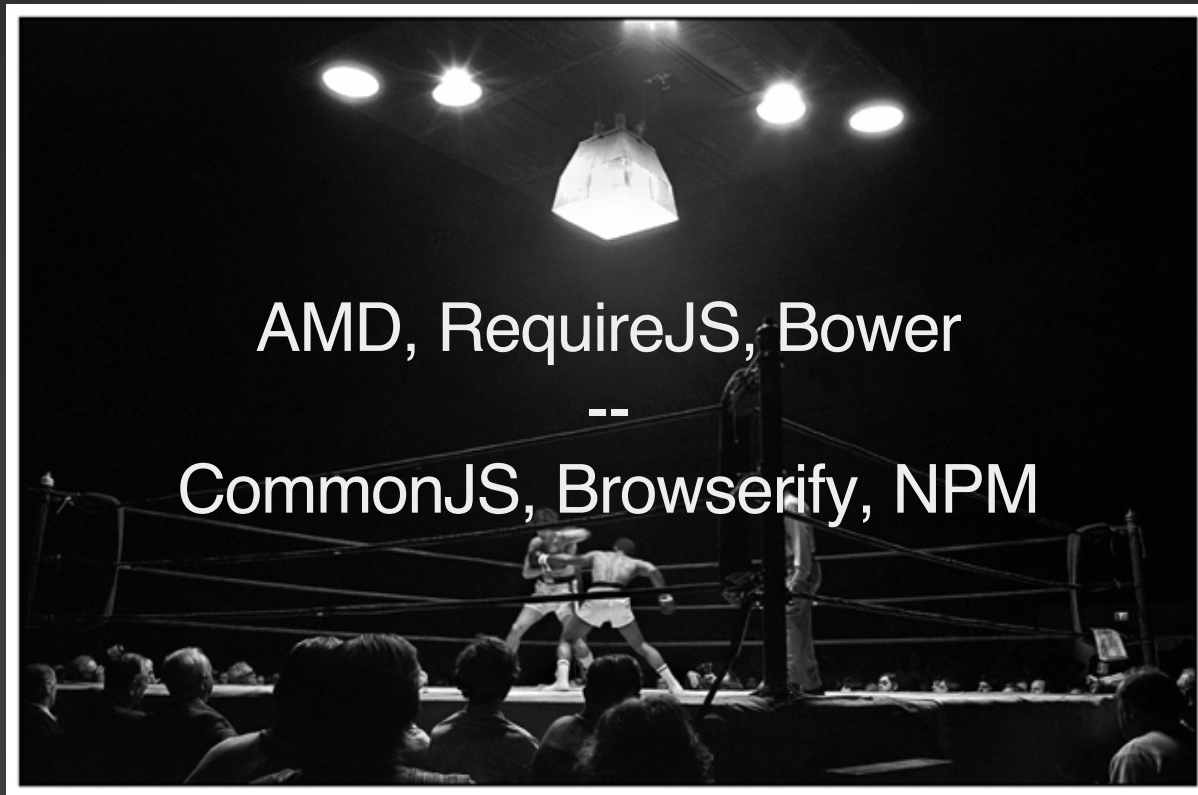
Modules

Tooling

Bonus: Repository

PLEASE DON'T SKIN THE CAT





How do we use these tools to actually solve this problem?

REQUIRE

INDEX.HTML

```
<script data-main="js/main" src="js/require.js"></script>
```

MAIN.JS

```
require(['beep', 'boop'], function(beep, boop) {  
    beep();  
    boop();  
});
```

BEEP.JS

```
define(['robot'], function(robot) {  
    var speak = function () {  
        robot('beep');  
    };  
    return speak;  
});
```

ROBOT.JS

```
define(function() {  
    return function (s) {  
        return console.log(s.toUpperCase() + '!!');  
    };  
});
```

BROWSERIFY

IN TERMINAL

```
$ browserify main.js > bundle.js
```

INDEX.HTML

```
<script src="js/bundle.js"></script>
```

MAIN.JS

```
var beep = require('beep');  
var boop = require('boop');  
beep();  
boop();
```

BEEP.JS

```
var robot = require('robot');  
var speak = function () {  
  robot('beep');  
};  
module.exports = speak;
```

ROBOT.JS

```
module.exports = function (s) {  
  return console.log(s.toUpperCase() + '!');  
};
```

BEEP + ROBOT?? - USING MODULE.EXPORTS OBJECT

```
var robot = require('robot');
var speak = function () {
  robot('beep');
};
var dance = function () {
  $('body').append('');
};
module.exports = {
  speak : speak,
  dance : dance
}
```

BEEP + ROBOT?? - USING EXPORTS.WHATEVS

```
var robot = require('robot');
exports.speak = function () {
  robot('beep');
};
exports.dance = function () {
  $('body').append('');
};
```

BEEP + ROBOT?? - USING REQ WITH HYBRID PATTERN

```
define(function(require, exports, module) {  
  var robot = require('robot');  
  exports.speak = function () {  
    robot('beep');  
  };  
  exports.dance = function () {  
    $('body').append('');  
  };  
});
```

BACKBONE VIEW

```
define([
  'jquery',
  'underscore',
  'backbone',
  'text!templates/example.html'
], function ($, _, Backbone, exampleTemplate) {
  var AppView = Backbone.View.extend({
    el: '#main',
    template: _.template(exampleTemplate),
    render: function () {
      this.$el.html(this.template({}));
      return this;
    }
  });

  return AppView;
});
```

BACKBONE VIEW

```
define(function (require, exports, module) {
  var $          = require('jquery');
  var _          = require('underscore');
  var Backbone   = require('backbone');
  var exampleTemplate = require('text!templates/example.html');

  var AppView = Backbone.View.extend({
    el: '#main',
    template: _.template(exampleTemplate),
    render: function () {
      this.$el.html(this.template({}));
      return this;
    }
  });

  return AppView;
});
```


**HERE ARE THE WAYS IN WHICH EACH
APPROACH SHINES:**

REQUIRE BIG WIN:

ASYNC MODULE LOAD

dive in quickly

debug loaded deps in console

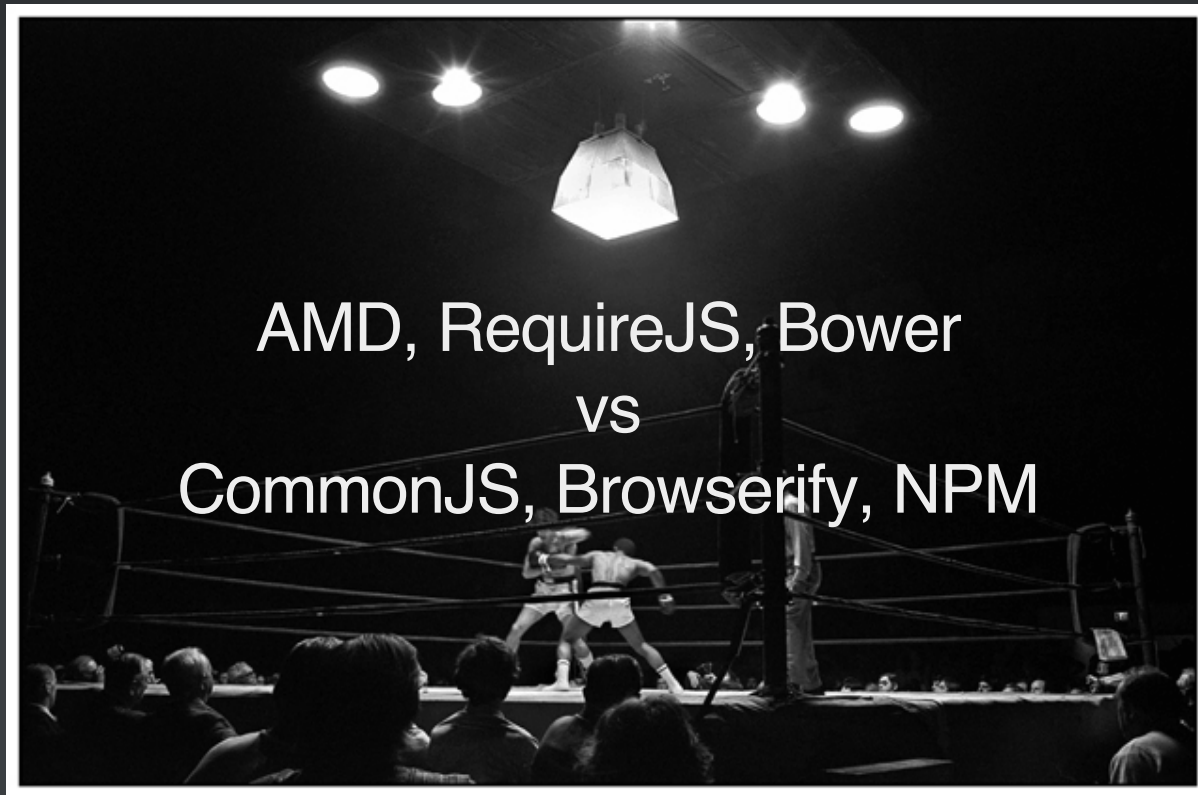
dynamically load code

BROWSERIFY BIG WIN(S):

`npm install [module-name] --save`

server and browser code share

through bundling (and transforms) you can do most anything



WHAT ABOUT ES6 MODULES?

WILL THEY SAVE US FROM OURSELVES AND SOLVE WORLD HUNGER?



I'm glad you asked!

IN CONCLUSION

Do what feels right

Come talk to me about what you think I missed or am missing

THE END

Presented to you by [Joe Sepi](#) / [@joe_sepi](#)