

- Load sample project (syntastic toggle)
- Get water
- Slow the heck down

Modern UI Development With Node.js



NETFLIX



Sir Tim Berners-Lee
writes HTML, URI, and
HTTP

Worlds first web page
launched

CS
Re



1990



1991

e Brehm coins the
m “isomorphic”
<https://medium.com/isomorphic-javascript-future-web-apps/>



Michael Jackson says
“universal” is better

<https://medium.com/@mjackson/universal-javascript-4761051b7ae9>

2011

2015

<code>

Step 9



Use. Modules.

**An IIFE is not a
module!**

ES6 or CommonJS

ES6 Modules

ECMA spec
browsers will (eventually) implement
strict mode default
named exports and imports

ES6 Modules

spec had been under churn
browser support coming along slowly
transpolation can get messy
cannot conditionally load them

CommonJS

Node.js default module system
A single syntax for all the modules
conditional loading

CommonJS

No strict mode default
different syntax for browser and node
conditional loading is a double edge sword

The code you
write **today** is
the code that will
be copied **tomorrow.**



ES6

ES2015

What is the first thing developers do
when they start coding in a new
project?

⌘ + C

+ C

+ V

Babel now!

classes

ui components as classes

```
1 var uiMessages = {  
2     getMessages: function(){  
3         return ...  
4     },  
5     addMessage: function( message ){  
6         ...  
7     },  
8     deleteMessage: function( messageID ){  
9         ...  
10    }  
11};  
12  
13 var messageBusMixin = {  
14     addEvent: function(){  
15         ...  
16     },  
17     getMessages: function (){  
18         ...  
19     }  
20};  
21  
22 var myComponent = React.createClass( {  
23     mixins: [ uiMessages, messageBusMixin ]  
24 } );
```

```
1 var uiMessages = {  
2     getMessages: function(){  
3         return ...  
4     },  
5     addMessage: function( message ){  
6         ...  
7     },  
8     deleteMessage: function( messageID ){  
9         ...  
10    }  
11};  
12  
13 var messageBusMixin = {  
14     addEvent: function(){  
15         ...  
16     },  
17     getMessages: function (){  
18         ...  
19     }  
20};  
21  
22 var myComponent = React.createClass( {  
23     mixins: [ uiMessages, messageBusMixin ]  
24 } );
```

```
1 var uiMessages = {  
2   getMessages: function(){  
3     return ...  
4   },  
5   addMessage: function( message ){  
6     ...  
7   },  
8   deleteMessage: function( messageID ){  
9     ...  
10  }  
11};
```

These aren't even mixins!

```
12 var messageBusMixin = {  
13   addEvent: function(){  
14     ...  
15   },  
16   getMessages: function (){  
17     ...  
18   }  
19 };  
20 };  
21  
22 var myComponent = React.createClass( {  
23   mixins: [ uiMessages, messageBusMixin ]  
24 } );
```

```
1 var uiMessages = {  
2   getMessages: function(){  
3     return ...  
4   },  
5   addMessage: function( message ){  
6     ...  
7   },  
8   deleteMessage: function( messageID ){  
9     ...  
10  }  
11};
```

They should be modules!

```
12 var messageBusMixin = {  
13   addEvent: function(){  
14     ...  
15   },  
16   getMessages: function (){  
17     ...  
18   }  
19 };  
20 };  
21  
22 var myComponent = React.createClass( {  
23   mixins: [ uiMessages, messageBusMixin ]  
24 } );
```

UI components are modules



Browse ▾

Kids

DVD

Search



My Account

MEMBERSHIP & BILLING

ryan.anklam@gmail.com

[Change email](#)

Password: *****

[Change password](#)

Phone: (414) 349-7603

[Email preferences](#)**VISA** **** *4451[Update payment info](#)[Billing details](#)Your **\$52.01** credit will be applied to your next bill.

PLAN DETAILS

2 Screens + HD[Change plan](#)

Your plan price is guaranteed through at least May 9, 2016.

SETTINGS

[Parental controls](#)[Test participation](#)[Activate a device](#)[Sign out of all devices](#)

MY PROFILE



Master

[Manage profiles](#)[Language](#)[Viewing activity](#)[Playback settings](#)[Taste preferences](#)[Subtitle appearance](#)[Ratings](#)[Order in My List](#)[Reviews](#)

GIFTS & OFFERS

[Redeem](#)[Where to buy gift cards](#)



Browse ▾

Kids

DVD

Search



My Account

MEMBERSHIP & BILLING

[Cancel membership](#)

ryan.anklam@gmail.com

Password: *****

Phone: (414) 349-7603

[Change email](#)[Change password](#)[Email preferences](#)**VISA** **** *4451[Update payment info](#)[Billing details](#)Your **\$52.01** credit will be applied to your next bill.

PLAN DETAILS

2 Screens + HD[Change plan](#)

Your plan price is guaranteed through at least May 9, 2016.

SETTINGS

[Parental controls](#)[Test participation](#)[Activate a device](#)[Sign out of all devices](#)

MY PROFILE

Master

[Manage profiles](#)[Language](#)[Viewing activity](#)[Playback settings](#)[Taste preferences](#)[Subtitle appearance](#)[Ratings](#)[Order in My List](#)[Reviews](#)

GIFTS & OFFERS

[Redeem](#)[Where to buy gift cards](#)



Browse ▾

Kids

DVD

Search



My Account

MEMBERSHIP & BILLING

[Cancel membership](#)

ryan.anklam@gmail.com

Password: *****

Phone: (414) 349-7603

VISA **** *4451

Your \$52.01 credit will be applied to your next bill.

[Change email](#)[Change password](#)[Email preferences](#)[Update payment info](#)[Billing details](#)

PLAN DETAILS

2 Screens -

[Change plan](#)

Your plan price is guaranteed through at least May 9, 2016.

No DVD plan

[Add DVD plan](#)

SETTINGS

[Parental controls](#)[Test participation](#)[Activate a device](#)[Sign out of all devices](#)

MY PROFILE



Master

[Manage profiles](#)[Language](#)[Playback settings](#)[Subtitle appearance](#)[Order in My List](#)[Viewing activity](#)[Taste preferences](#)[Ratings](#)[Reviews](#)

GIFTS & OFFERS

 Enter code or pin[Redeem](#)[Where to buy gift cards](#)

<build>

The build pipeline is the foundation modern UI projects are built on.



Your build chain is your development lifeline.

Treat it accordingly

Each step is a module.

Each module does one thing
(demo)

Rule of thumb

If you have to **require** more than one **plugin** you might want to create another module.

Tip!

Save your globs in a separate JSON file and require them into your build modules.

A photograph of a factory assembly line. Numerous yellow robotic arms are positioned above several cars, likely performing welding or painting tasks. The factory is filled with complex machinery, conveyor belts, and structural steel. The lighting is bright, typical of an industrial setting.

Use your build
tool to automate
mundane tasks.

testing



just do it

- Test as you code

- Test as you code
- Test before you push



Test as you code



Test before you push



Test in your CI Server

Tip!

Make testing easy and people will do it.

Easy is...

fast

automated

easy to bootstrap

having a culture of testing

Code Quality

eslint

Why lint?

catch common errors

catch oops moments (console.log, debugger)

why eslint?

Style checker

Pluggable

Future friendly: ES6, JSX

configure rules [ignore, warn, error]

csslint

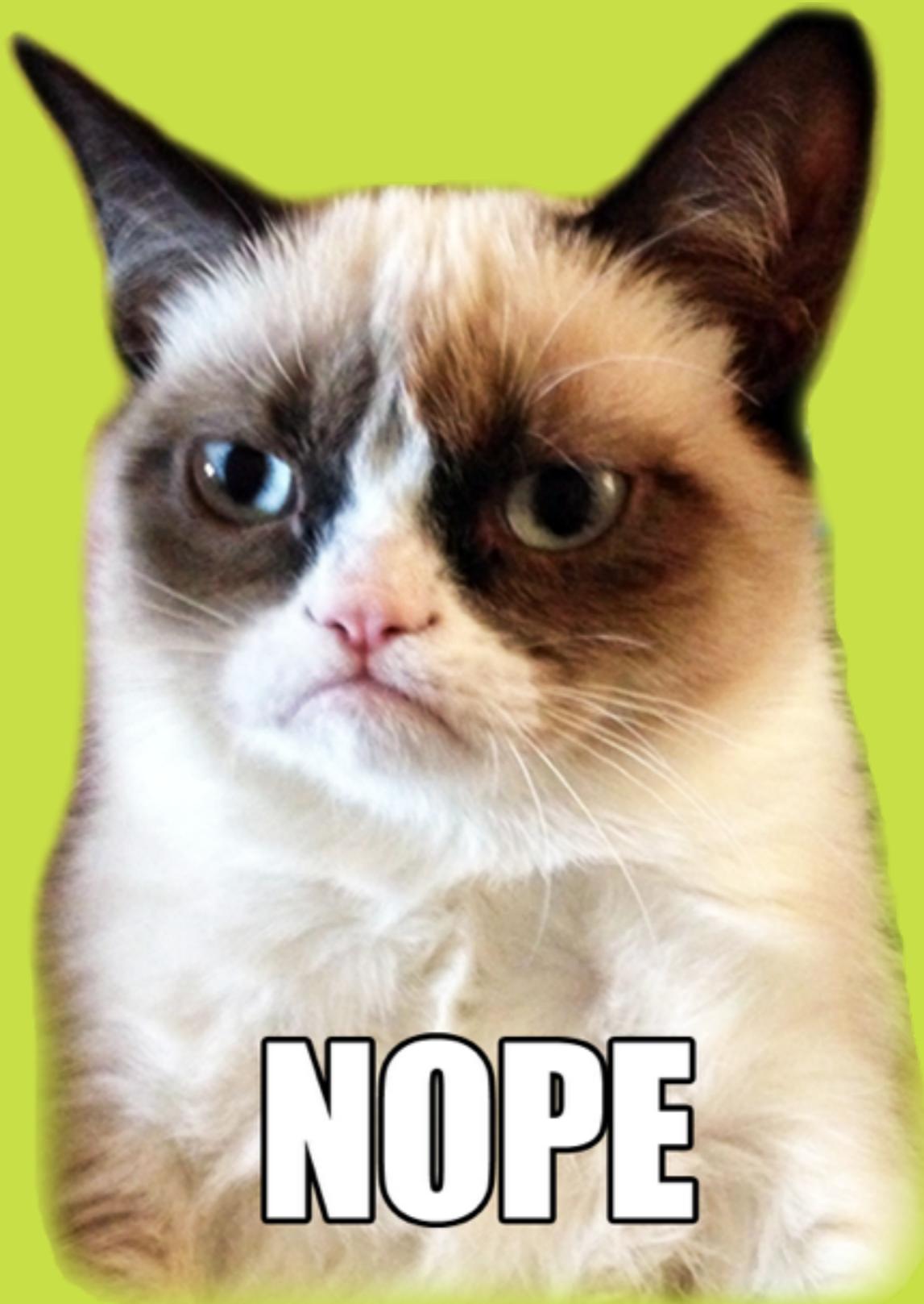
Nesting

is the **worst** thing to happen to CSS.

You should never see this in css!

```
8      }
7      }
6      }
5      }
4      }
3 }
2
```

Don't let
lint errors in your
repos.



How?

push hooks

For GIT

Add these files to your .git directory's hooks folder

- pre-commit
- pre-push
- pre-rebase
- commit-msg
- post-update
- post-checkout
- post-merge

<http://www.sitepoint.com/introduction-git-hooks/>

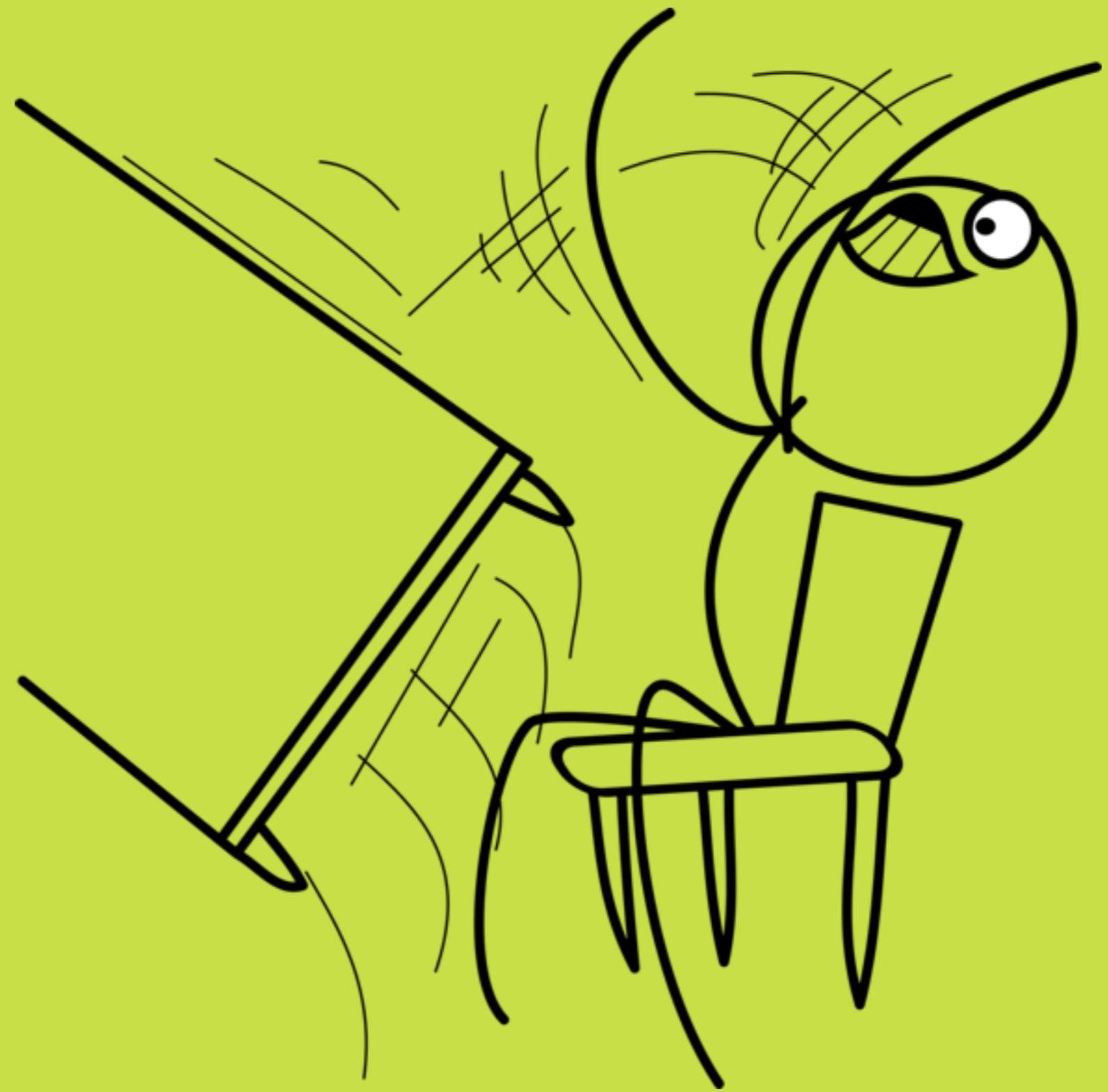
This is your “Nope” button

pre-commit
pre-push
pre-rebase
commit-msg
post-update
post-checkout
post-merge



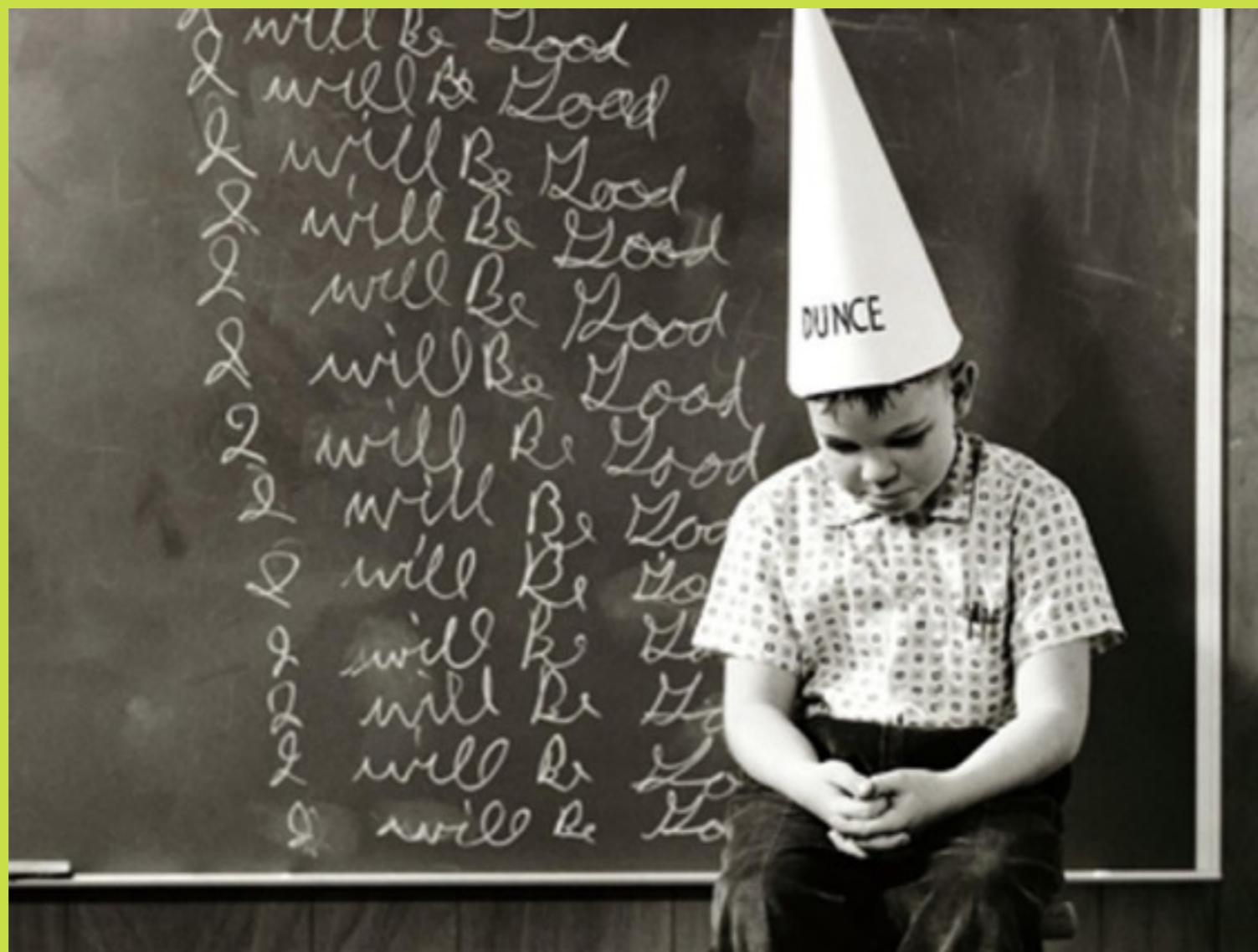
<http://www.sitepoint.com/introduction-git-hooks/>

```
git push --no-verify
```



Continuously Lint & Test

Add your testing and linting
as part of the build process and in your CI tool.



Public shaming is
a great motivator.

Automate Adding Hooks

```
1  'use strict';
2
3
4  module.exports.importTasks = function(gulp) {
5      //-----
6      // Install git hooks
7      //-----
8      gulp.task('hook', function() {
9          var symlink = require('gulp-symlink');
10
11          return  gulp.src([ './devtools/githooks/*' ])
12              .pipe(symlink('.git/hooks/'));
13      });
14  };
```

<serve>

- business logic, maybe some sql

- business logic, maybe some sql
- php/jsp/asp/erb/etc

- 
- business logic, maybe some sql
 - php/jsp/asp/erb/etc
 - run some JS on the client

- RESTful API's

- RESTful API's
- JS template on the server

- 
- RESTful API's
 - JS template on the server
 - interactive JS on the client

There is a large cost to context switching.

Proceed with Caution!

Let node do what it's good at, nothing more.

Don't block

Don't implement business logic

fire & forget, take advantage of the message queue

Make calls to async services
Compose a template
Send the response

**Some things have to be
synchronous.**

Keep as many of those at startup as you can.

Be aware of the “hot” path!

Code that will run with **EVERY** request.

Keep your functional programming
off the hot path.

But what if?

- RESTful API's

- RESTful API's
- JavaScript

Isomorphic!

Unimorphic!

Isomorual!

Universal!

Don't let the demos fool you!

Universal JS is **hard!**

Calling a template with some data to
produce HTML is easy

De-hydrating data in Node and
re-hydrating client in the browser is
HARD.

Getting your data can
be **really hard.**

Don't

```
1  function isomorphicFunction (){
2      var data;
3      if( typeof window === 'undefined' ){
4          data = window.browserContext;
5      }
6      else{
7          data = require( './singletons/data' );
8      }
9  }
```

Do

```
1 function contextResolver(){  
2     ...  
3 }  
4  
5 var context = contextResolver();  
6  
7 function isomorphicFunction ( ctx ){  
8     var data = ctx.data;  
9 }
```

Use meta data to tell other modules what data the UI needs.

In React, statics are perfect for this.

```
var MyAccount = React.createClass( {  
  
    statics : {  
        __meta__ : {  
            stores : [ 'accountInfoStore', 'serviceInfoStore' ]  
        }  
    },
```

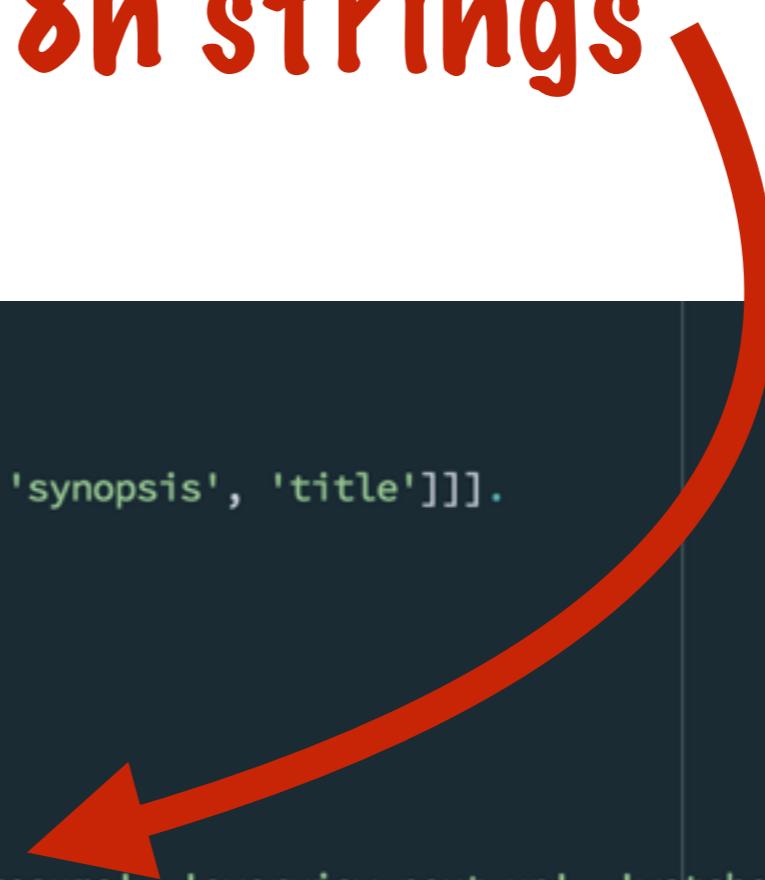
Bind Falkor paths

```
statics: {  
  getPaths: function () {  
    var paths = [['current', ['summary', 'synopsis', 'title']]].concat(Watched.getPaths());  
  
    return paths;  
  },  
  __meta__: {  
    strings: [{  
      bundle: BUNDLE, ids: ['overview.resume', 'overview.next.up', 'watched.title.overview']  
    }]  
  }  
},
```



Load i18n strings

```
statics: {  
  getPaths: function () {  
    var paths = [['current', ['summary', 'synopsis', 'title']]].  
      concat(Watched.getPaths());  
  
    return paths;  
  },  
  __meta__: {  
    strings: [{  
      bundle: BUNDLE, ids: ['overview.resume', 'overview.next.up', 'watched.title.overview']  
    }]  
  }  
},
```



Webpack & Browserify



```
1  'use strict';
2  var request = require( 'superagent' );
3
4  module.exports = {
5      doLogin: function( error, success ){
6          request.post( 'api/login' )
7              .send( { username : 'username',
8                      password : 'password'
9                  } )
10             .set('Accept', 'application/json')
11             .end( function( err, res ){
12                 if( err ){
13                     return error( err );
14                 }
15
16                     return success( res );
17             } );
18     }
19  };
```

Node code vs Browser code

Client code is bundled, node code isn't.

Why not run bundles in Node?

It only takes a few configuration tweaks

<http://jlongster.com/Backend-Apps-with-Webpack--Part-I>

0) Collect your node_modules

```
9  fs.readdirSync('node_modules')
10 .filter(function(x) {
11     return ['.bin'].indexOf(x) === -1;
12
13 })
14 .forEach(function(mod) {
15     nodeModules[mod] = 'commonjs ' + mod;
16 });
```

Not a “hot” code path

```
9  fs.readdirSync('node_modules')
10 .filter(function(x) {
11   return ['.bin'].indexOf(x) === -1;
12 }
13 })
14 .forEach(function(mod) {
15   nodeModules[mod] = 'commonjs ' + mod;
16 });


```

1) Add another config to webpack.config.js

Export an array

```
18 module.exports = [ ←  
19   {  
20     entry: './server.js',  
21     target: 'node',  
22     ...  
23   },  
24   {  
25     entry: './ui/js/root.js',  
26     target: 'web',  
27     filename: 'root.js'  
28   },  
29   ...
```

Have multiple targets

```
18 module.exports = [
19   {
20     entry: './server.js',
21     target: 'node', ←
22     ...
23   },
24   {
25     entry: './ui/js/root.js',
26     target: 'web', ←
27     filename: 'root.js'
28   },
29   ...
```

2) Tell web pack to ignore your requires for node_modules
AND your CSS

No Node node_modules

```
18 module.exports = [
19   {
20     entry: './server.js',
21     target: 'node',
22     output: {
23       path: path.join(__dirname, 'build', 'server' ),
24       filename: 'server.js'
25     },
26     externals: nodeModules,
27     plugins : [
28       new webpack.NormalModuleReplacementPlugin(/\.css$/, 'node-noop')
29     ],

```

No css

```
18 module.exports = [
19   {
20     entry: './server.js',
21     target: 'node',
22     output: {
23       path: path.join(__dirname, 'build', 'server'),
24       filename: 'server.js'
25     },
26     externals: nodeModules,
27     plugins : [
28       new webpack.NormalModuleReplacementPlugin(/\.css$/, 'node-noop')
29     ],
30   }
31 ];
```

```
$> node build/server.js
```

Next level
Stuff

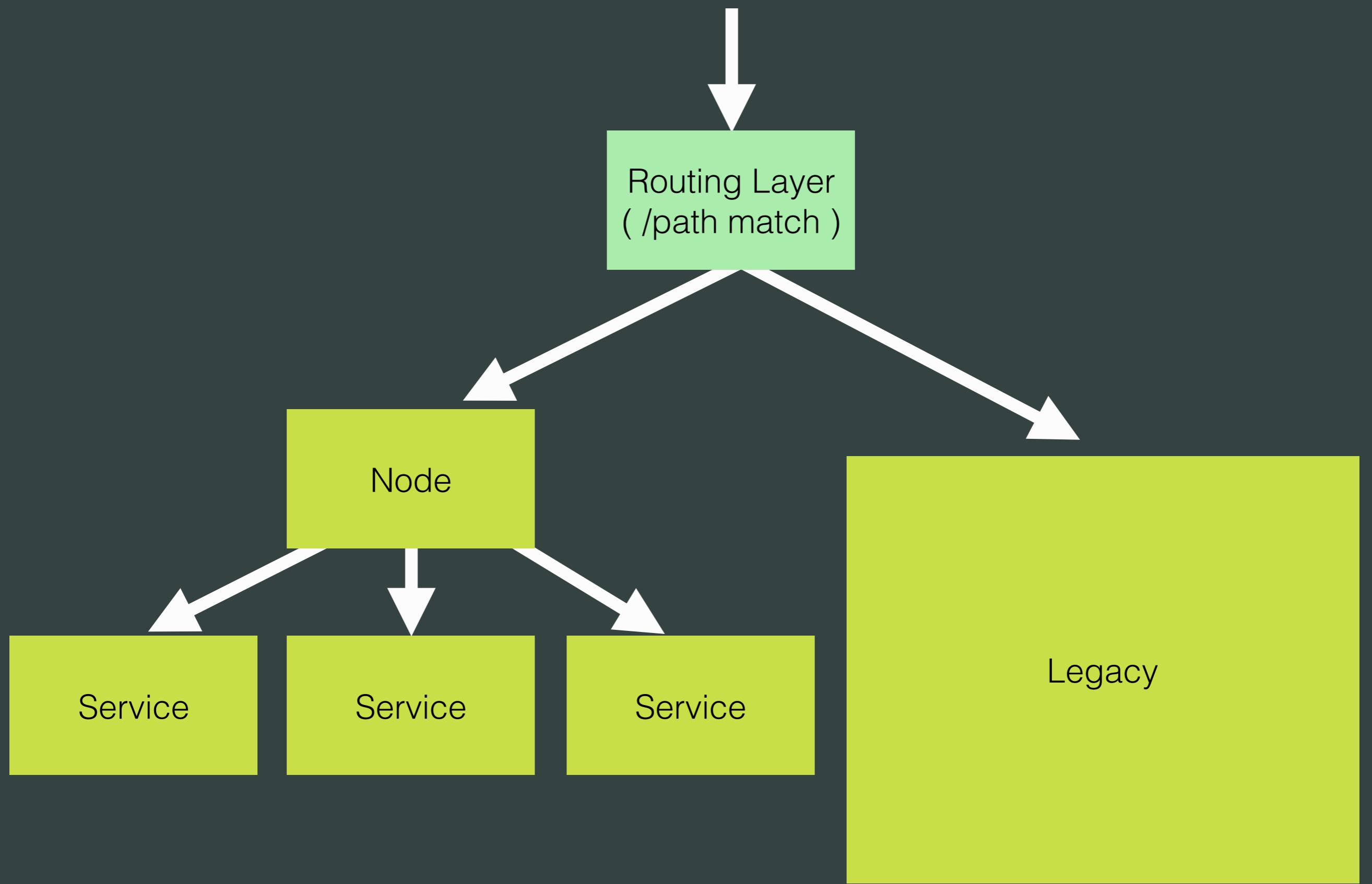
Weback dev server & hot module swapping.

(demo)

Last thoughts...

Start Small When Transitioning

There are lots of learnings to be had from a single page



Iterate on Modules

Modules don't have to be perfect at first
learn & refactor

Keep your code clean

Don't let bad code infect your repository.

Automate

People are lazy. They will do the easy things and ignore the hard things.

References

<http://webdesign.tutsplus.com/articles/a-brief-history-of-the-world-wide-web--webdesign-8710>

<http://www.evolutionoftheweb.com/>

<http://nerds.airbnb.com/isomorphic-javascript-future-web-apps/>

<http://jlongster.com/Backend-Apps-with-Webpack--Part-I>

Questions?

Thank You.

ranklam@netflix.com

ryan.anklam@gmail.com