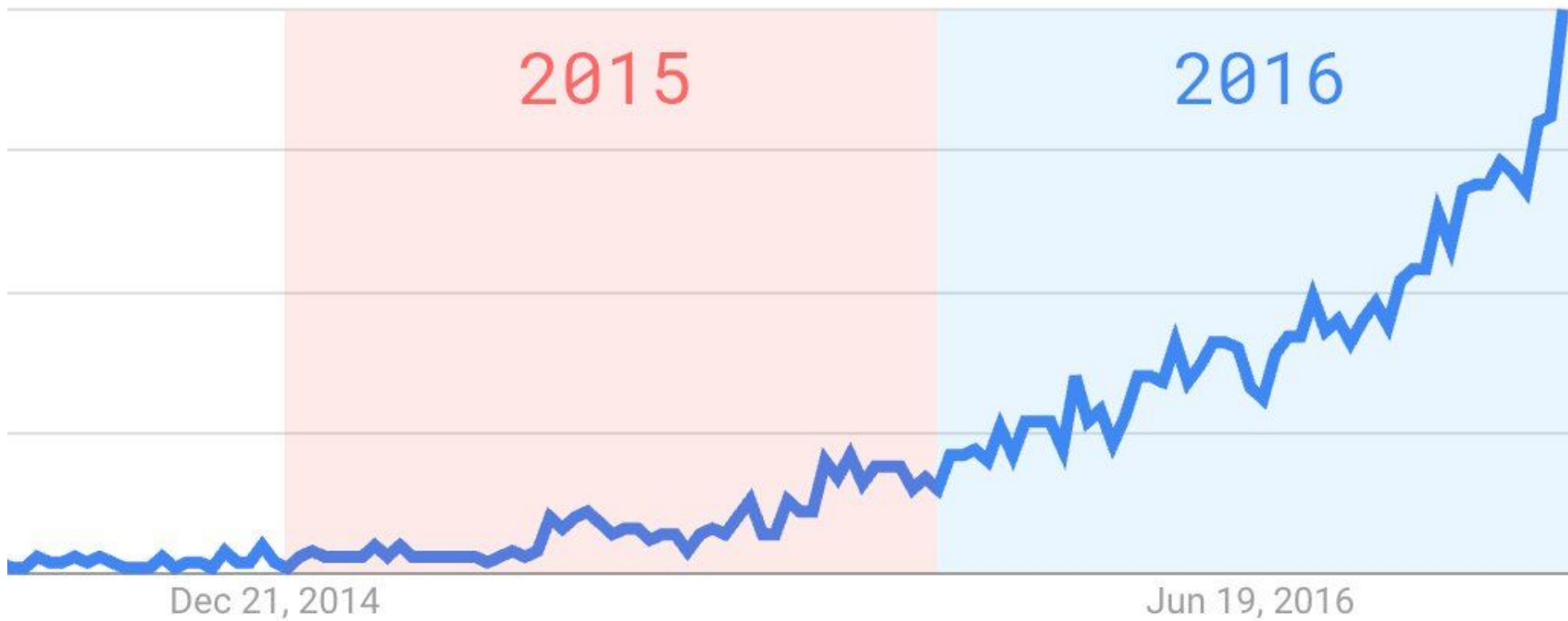


Vue.js

One of the frontend trends in 2016

Google Trends for "vuejs + vue.js"



Project started in 2013



Evan You
Creator of Vue.js

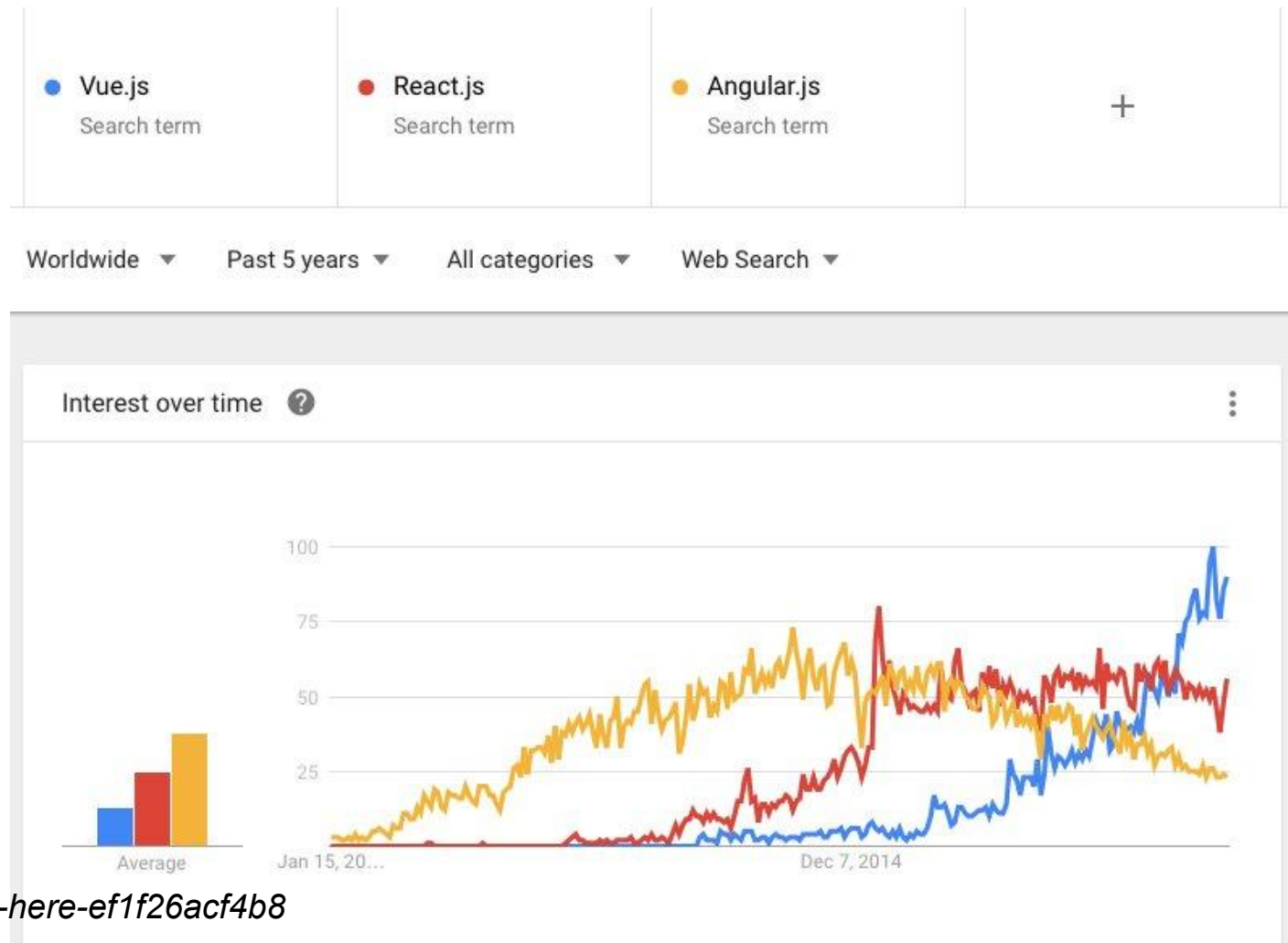
*A progressive, incrementally-adoptable
JavaScript framework for building UI on
the web.*

History

v 1.0 released 10.2015

v 2.0 released in 09.2016
with major changes &
improvements

Current stable: **v 2.1.10**



<https://medium.com/the-vue-point/vue-2-0-is-here-ef1f26acf4b8>



**Oh, new Javascript
framework !**

Developer adoption rate



★ Star

39,367

100s plugins
UI packages, tooling, etc.
Tons of videos and posts

* - As of 12 Jan 2017

Business usage

Alibaba, Baidu, Xiaomi,

Optimizely, Expedia, Nintendo,

DJI, Hunliji, GitLab, etc.

Vue.js

selling points

1.

**Incredibly small
footprint**

23k-framework, srsly ?!

GZipped

```
$ gzip -r .  
$ ls -lhS  
123K Jan  4 22:11 ember.1.13.8.min.js.gz  
119K Jan  4 22:11 angular2.0.0-beta.0-all.umd.min.js.gz  
111K Jan  4 22:11 ember.2.2.0.min.js.gz  
111K Jan  4 22:11 angular2.min.js.gz  
51K Jan  4 22:11 angular.1.4.5.min.js.gz  
45K Jan  4 22:11 angular.1.3.2.min.js.gz  
42K Jan  4 22:11 react-with-addons-0.14.5.min.js.gz  
39K Jan  4 22:11 react-0.14.5.min.js.gz  
32K Jan  4 22:11 angular2.0.0-beta.0-Rx.min.js.gz  
1.9K Jan  4 22:11 redux-3.0.5.min.js.gz  
455B Jan  4 22:11 react-dom-0.14.5.min.js.gz  
23K  Oct 13 03:02 vue-2.0.3.min.js.gz
```

<https://gist.github.com/Restuta/cda69e50a853aa64912d>

What utility libs you're using in your current projects?

Vue.js has approx same size as one of those:

- Lodash v.4
- MomentJS (with locales)
- jQuery v.3

2.

**It's about developer's
(experience)**

You want the API a human can master

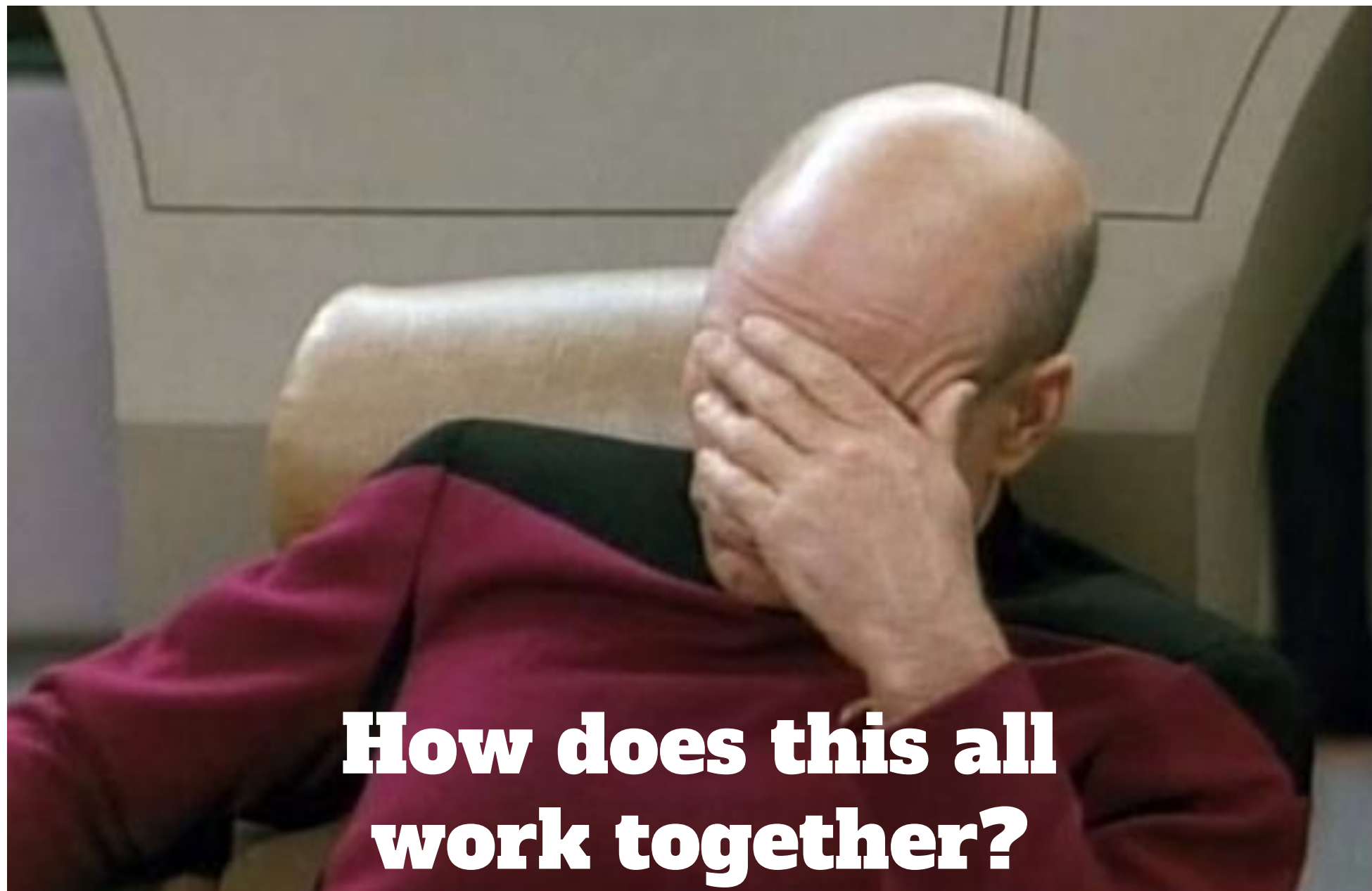
@angular/common

K APP_BASE_HREF	P AsyncPipe	C CommonModule
P CurrencyPipe	P DatePipe	P DecimalPipe
C HashLocationStrategy	P I18nPluralPipe	P I18nSelectPipe
P JsonPipe	C Location	I LocationChangeEvent
I LocationChangeListener	C LocationStrategy	P LowerCasePipe
D NgClass	D NgFor	D NgIf
C NgLocalization	D NgPlural	D NgPluralCase
D NgStyle	D NgSwitch	D NgSwitchCase
D NgSwitchDefault	D NgTemplateOutlet	C PathLocationStrategy
P PercentPipe	C PlatformLocation	P SlicePipe
P UpperCasePipe	K VERSION	

@angular/core

K ANALYZE_FOR_ENTRY_COMPONENTS	K APP_BOOTSTRAP_LISTENER	K APP_ID
K APP_INITIALIZER	K AUTO_STYLE	C AfterContentChecked
C AfterContentInit	C AfterViewChecked	C AfterViewInit
C AnimationAnimateMetadata	C AnimationEntryMetadata	C AnimationGroupMetadata
C AnimationKeyframesSequenceMetadata	C AnimationMetadata	C AnimationPlayer
C AnimationSequenceMetadata	C AnimationStateDeclarationMetadata	C AnimationStateMetadata
C AnimationStateTransitionMetadata	C AnimationStyleMetadata	C AnimationTransitionEvent
C AnimationWithStepsMetadata	C ApplicationInitStatus	C ApplicationModule
C ApplicationRef	I Attribute	K COMPILER_OPTIONS
K CUSTOM_ELEMENTS_SCHEMA	E ChangeDetectionStrategy	C ChangeDetectorRef
F Class	I ClassDefinition	I ClassProvider
C CollectionChangeRecord	C Compiler	C CompilerFactory
T CompilerOptions	@ Component	C ComponentFactory
C ComponentFactoryResolver	C ComponentRef	@ ContentChild
C SimpleChange	I SimpleChanges	@ SkipSelf
C SystemJsNgModuleLoader	C SystemJsNgModuleLoaderConfig	K TRANSLATIONS
K TRANSLATIONS_FORMAT	C TemplateRef	C Testability
C TestabilityRegistry	I TrackByFn	C Type

<https://angular.io/docs/ts/latest/api/>



**How does this all
work together?**

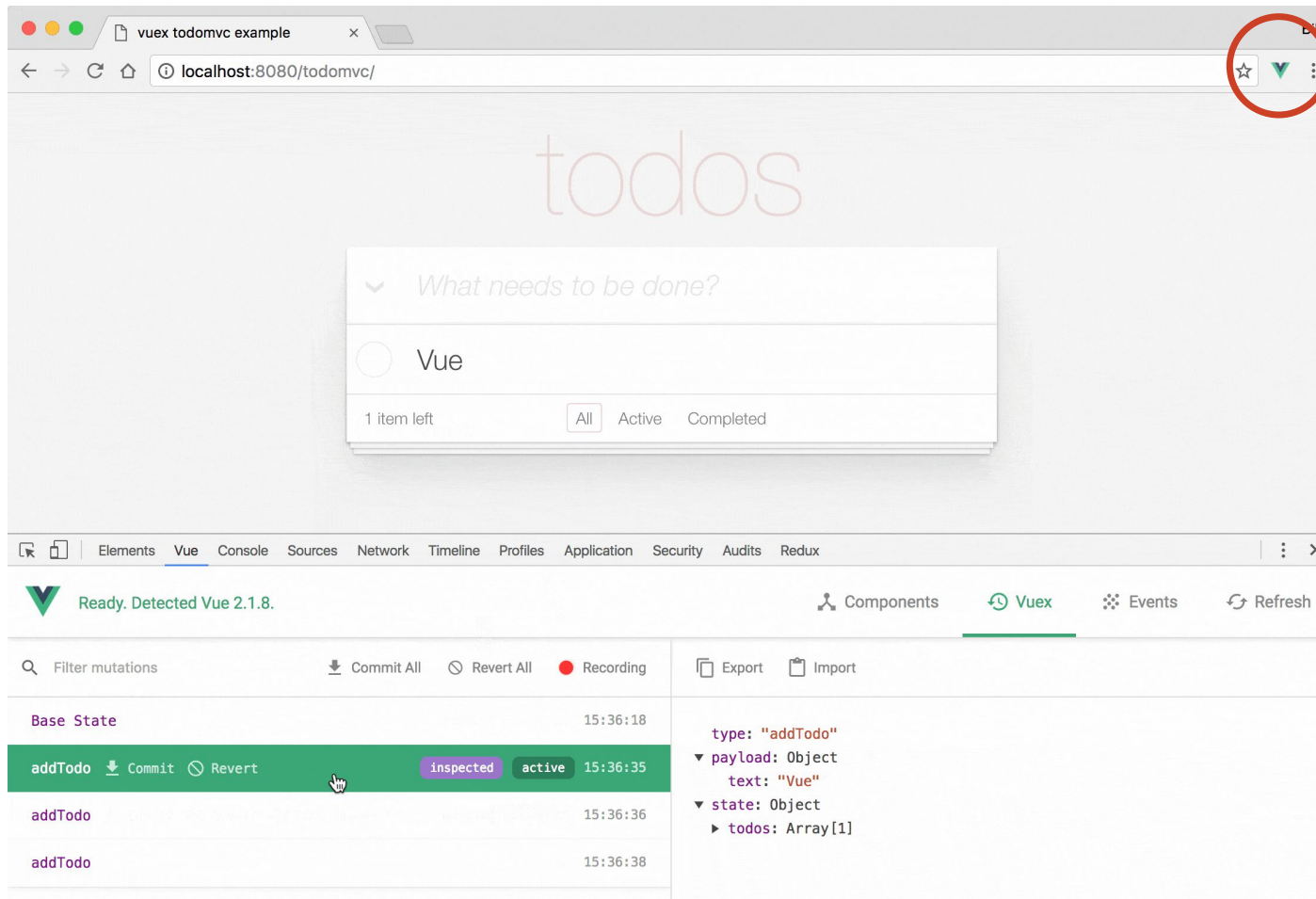
Clear guidance

- Lightweight, transparent API
- No 1.x, 2.0-rc, 2.0-rc1..., 2.1, ~~3.0~~, 4.x etc version mess
- Up to date and [detailed docs](#)

Comprehensive developer experience

- [Vue-devtools](#), [vue-cli](#)
- Tooling for all major build systems
- Syntax highlighting and snippets for [<favorite-editor-name>](#)

Dev tools



- Components tree and properties exploration
- Time-travel debugging

3.

**Syntax you already
know (and love)**

Component lifecycle



beforeCreate/created

beforeMount/mounted

beforeUpdate/updated

beforeDestroy/destroyed



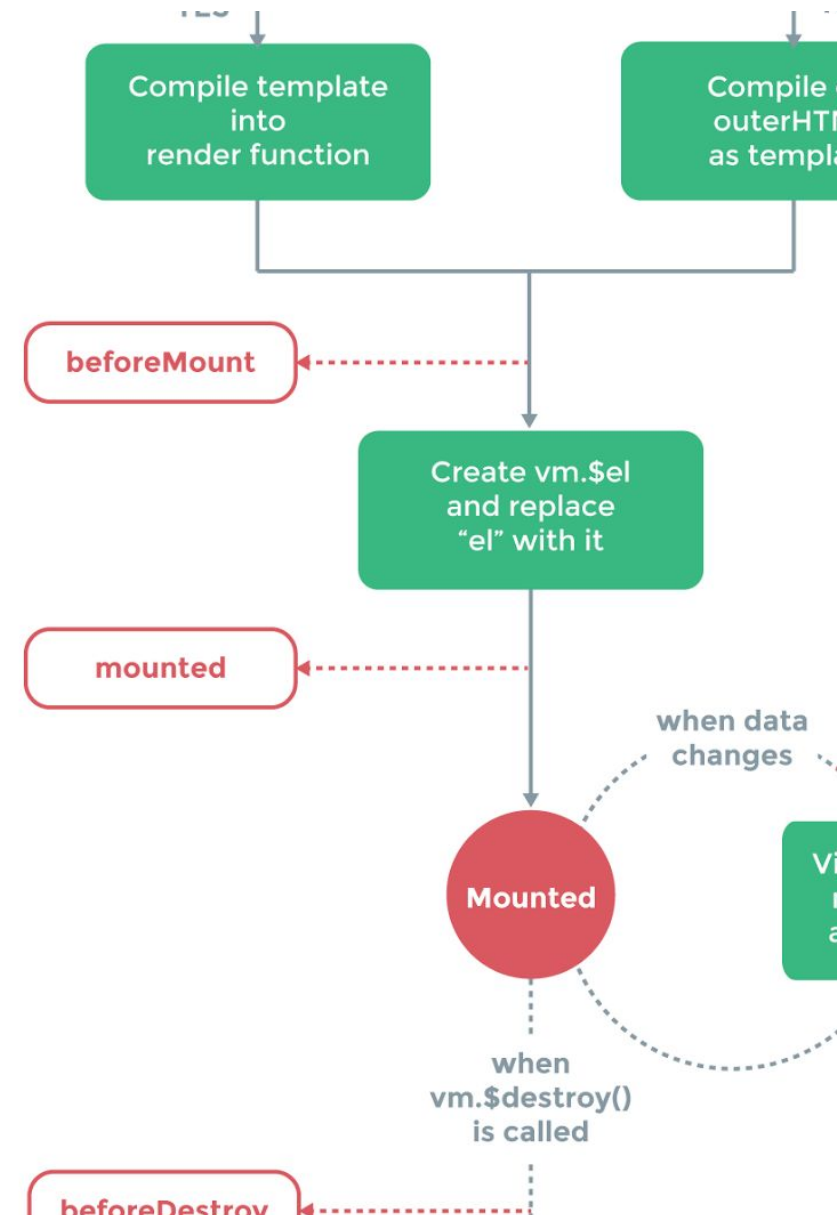
\$onInit

\$postLink

\$onChanges

\$onDestroy

<https://vuejs.org/v2/guide/instance.html#Lifecycle-Diagram>
<https://toddmotto.com/angular-1-5-lifecycle-hooks#using-oninit>



Template syntax

```
<div v-html="rawHtml"></div>
```

```
<p v-if="seen">Now you see me</p>
```

```
<a v-on:click="doSomething"></a>
```

```
<a @click="doSomething"></a>
```

```
<div v-bind:class="{ active: isActive }"></div>
```

And so on...

Similar concepts

Template expressions

Directives (built-in and custom)

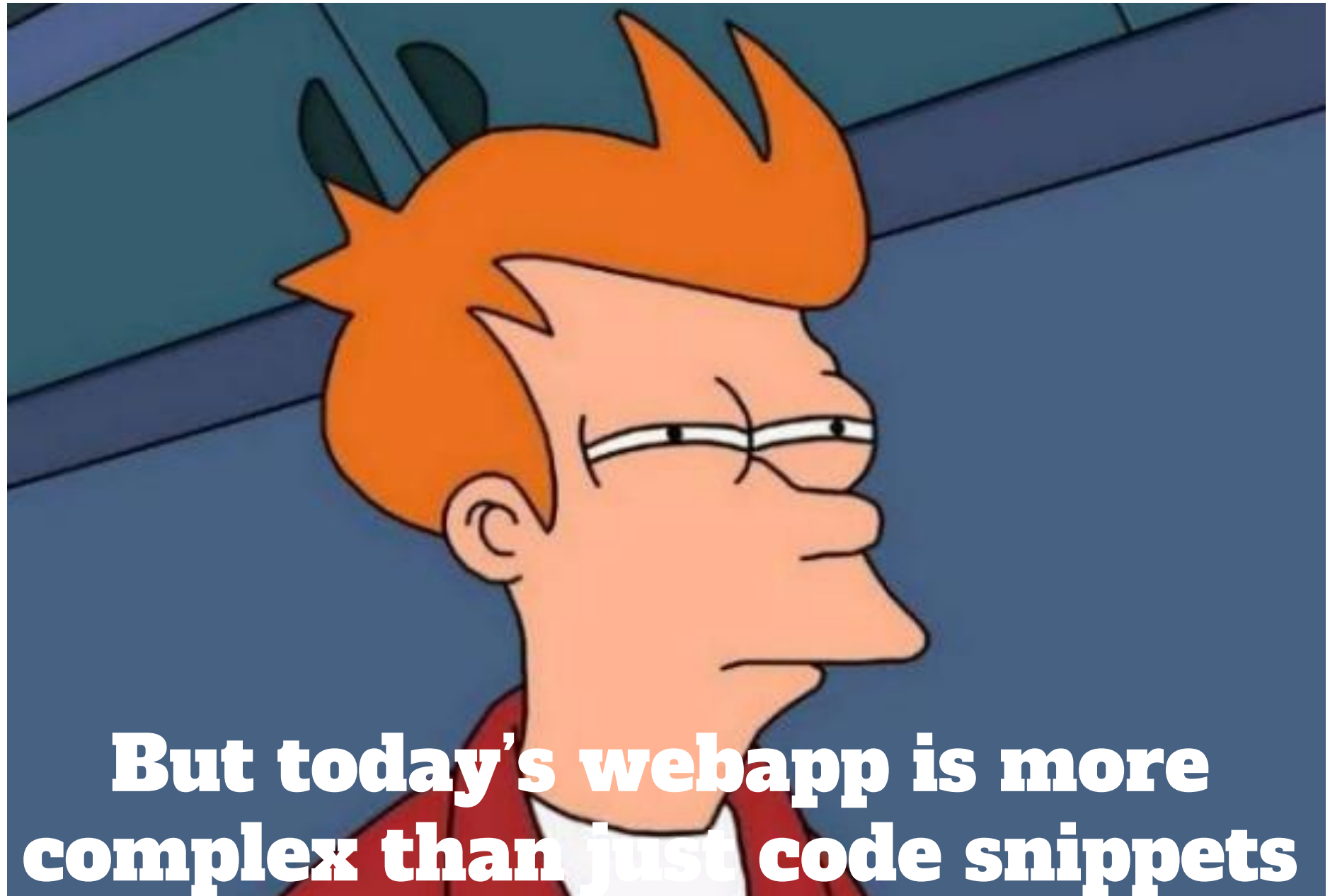
Filters

Events (\$on, \$off, \$emit, \$once)

Watchables

\$render function

<slot> (like ng-transclude)



**But today's webapp is more
complex than just code snippets**

App code samples

Vue component

```
// register
Vue.component('my-component', {
  el: '#counter-event-example',
  template: '<div>A custom component!</div>',
  props: ['message'],
  methods: {
    incrementTotal: function () {
      this.total += 1
    }
  }
})
```

Solid template - github.com/posva/vue-plugin-template

Online generator - changjoo-park.github.io/vue-component-generator/

```
1 <template lang="jade">
2   div
3     p {{ greeting }} World!
4     other-component
5 </template>
6
7 <script>
8   import OtherComponent from './OtherComponent.vue'
9
10  export default {
11    data () {
12      return {
13        greeting: 'Hello'
14      }
15    },
16    components: {
17      OtherComponent
18    }
19  }
20 </script>
21
22 <style lang="stylus" scoped>
23   p
24     font-size 2em
25     text-align center
26 </style>
```


Vue router

```
const routes = [  
  { path: '/foo', component: Foo },  
  { path: '/bar', component: Bar }  
]
```

```
// 3. Create the router instance and pass the `routes` option  
// You can pass in additional options here, but let's  
// keep it simple for now.
```

```
const router = new VueRouter({  
  routes // short for routes: routes  
})
```

```
// 4. Create and mount the root instance.  
// Make sure to inject the router with the router option to make  
the
```

```
// whole app router-aware.
```

```
const app = new Vue({  
  router  
}).$mount('#app')
```

```
<!-- use router-link component for navigation. -->
```

```
<!-- specify the link by passing the `to` prop. -->
```

```
<!-- <router-link> will be rendered as an `<a>` tag by  
default -->
```

```
<router-link to="/foo">Go to Foo</router-link>
```

```
<router-link to="/bar">Go to Bar</router-link>
```

<https://router.vuejs.org/en/essentials/getting-started.html>

Vuex

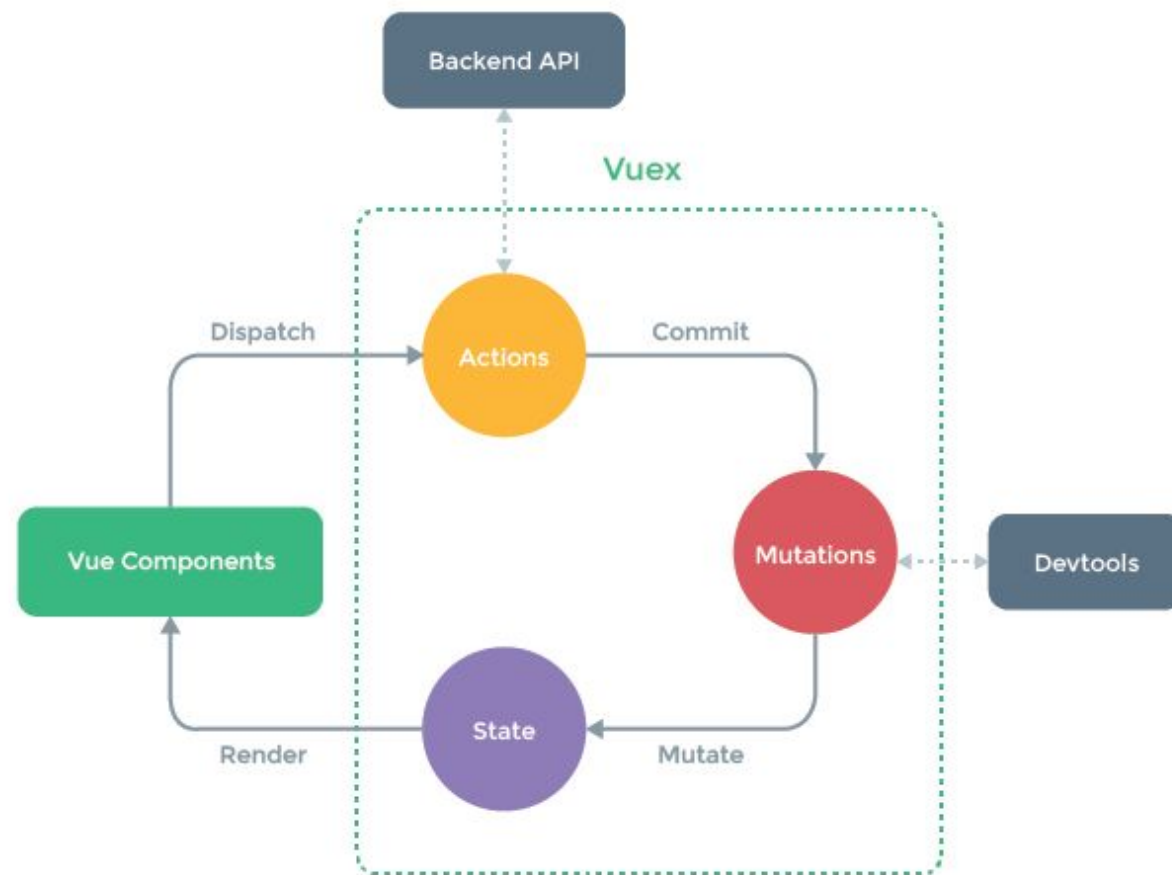
```
import store from './store'
import App from './components/App.vue'

new Vue({
  store, // inject store to all children
  el: '#app',
  render: h => h(App)
})
```

```
//in the component methods
this.$store.commit('addTodo', { text })
```

```
//mutation example
export const mutations = {
  addTodo (state, { text }) {
    state.todos.push({
      text,
      done: false
    })
  },
}
```

<https://vuex.vuejs.org/en/structure.html>



Other highlights

- Handful of UI libraries (material, bootstrap, mobile)
- SSR (e.g. [nuxt.js](#) or [prerender plugin](#))
- State management (Redux clone [vux](#), Event Bus)
- Lazy component loading (eg. [vue-lazy-component](#))
- Form handling

**Live coding
time!**

Setup starter project

```
$ npm install vue-cli -g
```

```
$ vue init webpack my-nice-project
```

```
$ cd my-nice-project
```

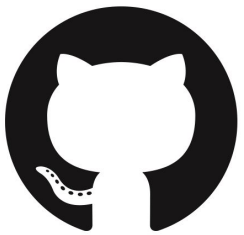
```
$ npm install
```

```
$ npm i mint-ui axios vue-online --save
```

```
$ npm run dev
```

Live coding plan

- Install stuff
- Connect to <http://pokeapi.co/> using **axios**
- Create components to output + some UI with **mint-ui**



<https://github.com/shershen08/vue-pokemon-app-demo>

Do or Don't?

Points to be aware of

- Vue.js SSR is bounded to NodeJS only env
- No monster corporation behind
- Ecosystem is still 10x smaller than React / Angular

Why you should start now

- Mental bridge from 'Angular' to 'React' world
- Can start immediately if you comfortable with Angular
- **No opinionation** (how you handle network requests or structure your modules)
- So small so you can start with a fragment of your UI

Check out the links

- Introduction mini-book (PDF + github repo)
- Awesome list: <https://github.com/vuejs/awesome-vue>
- Newsletter: <https://www.getrevue.co/profile/vuenewsletter>
- Twitter: <https://twitter.com/vuejs> and #vuejs tag
- Stackoverflow: <http://stackoverflow.com/tags/vue.js/info>
- Chats: <https://gitter.im/vuejs/vue>, http://telegram.me/vuejs_ru