

# Intro to Vue.js

---

Sami Boudrai

# Vue.js

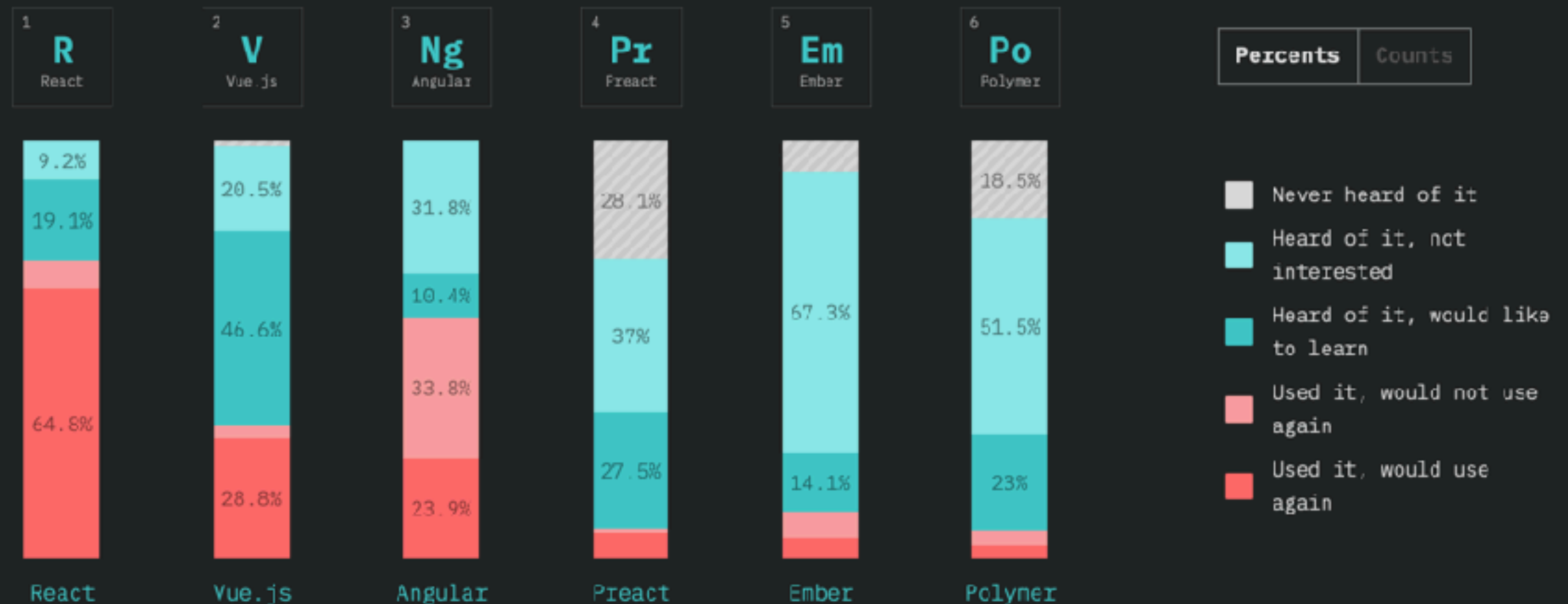
---

- Front-end framework started by Evan You while working at Google
- Simple, well documented and consistent
- No big company backing: Evan's Patreon + OpenCollective
  - React -> Facebook
  - Angular -> Google

**“Lots of developers just need a solution  
that’s tailored for them”**

*– Evan You*

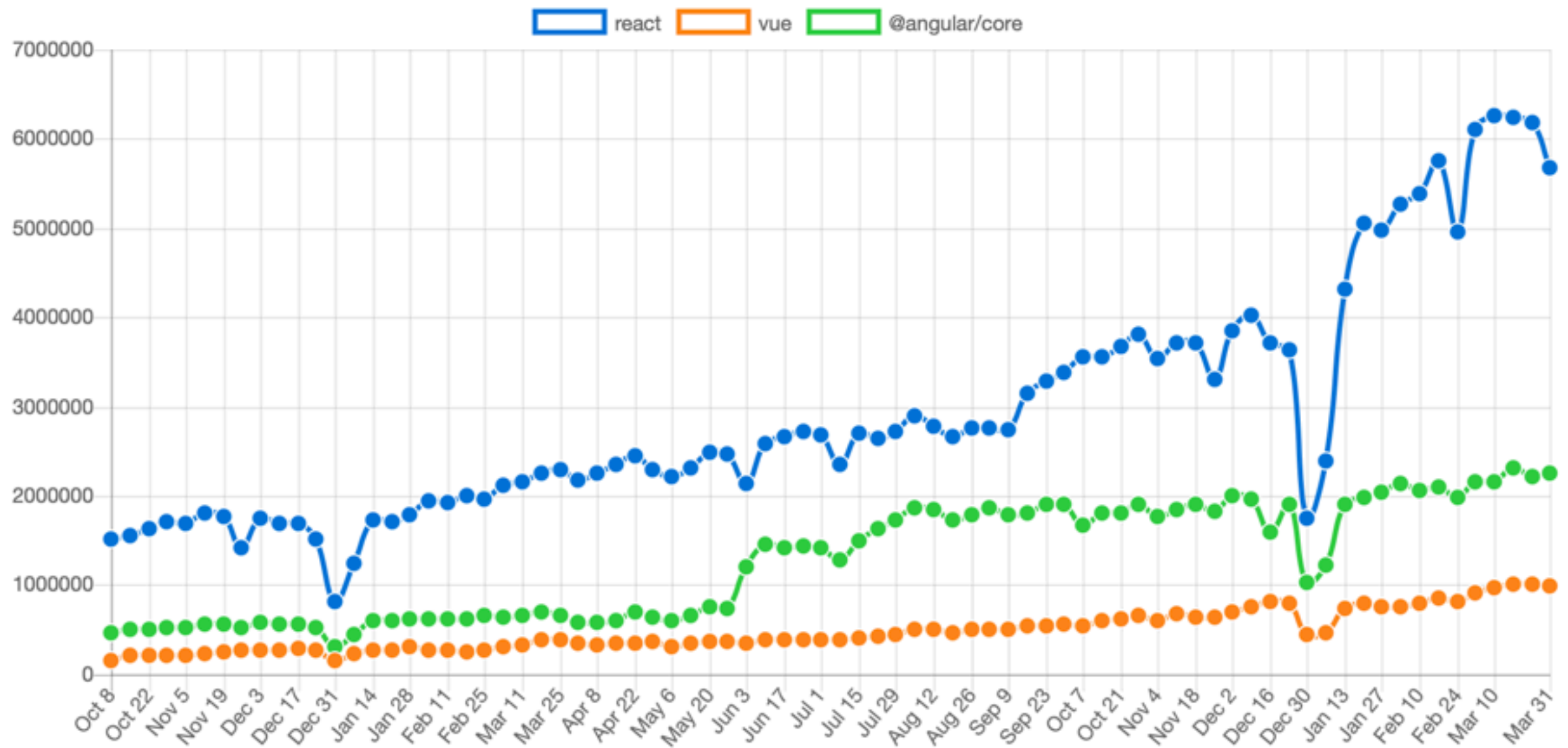
## Overall Front-end Frameworks results.



# State of Js 2018

<https://2018.stateofjs.com/front-end-frameworks/overview/>

## Downloads in past 2 Years ▾

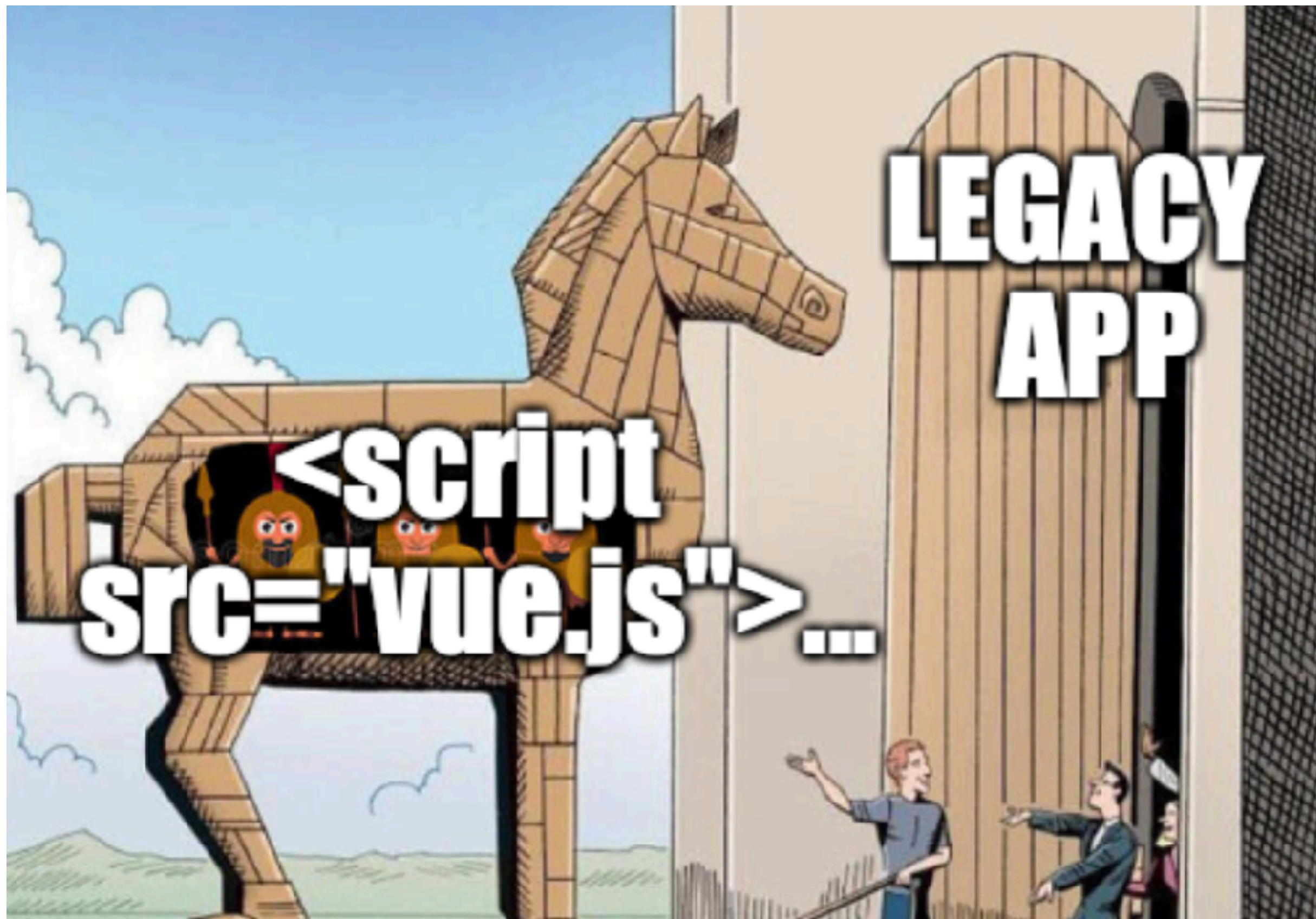


<https://www.npmtrends.com/react-vs-vue-vs-@angular/core>

## Stats

	stars 🌟	forks 🍴	issues ⚠️	updated 🔧	created 🐥	size 🏆
<a href="#">react</a>	126275	22910	638	Apr 2, 2019	May 24, 2013	minified size 6.5 KB
<a href="#">vue</a>	133986	19054	248	Apr 1, 2019	Jul 29, 2013	minified size 63.5 KB
<a href="#">@angular/core</a>	46697	12344	2699	Apr 2, 2019	Sep 18, 2014	minified size 252.2 KB

<https://www.npmtrends.com/react-vs-vue-vs-@angular/core>



(from [@vueconfus](#))



# jQuery way

---

Type a message

- first message
- second message

2 messages so far.

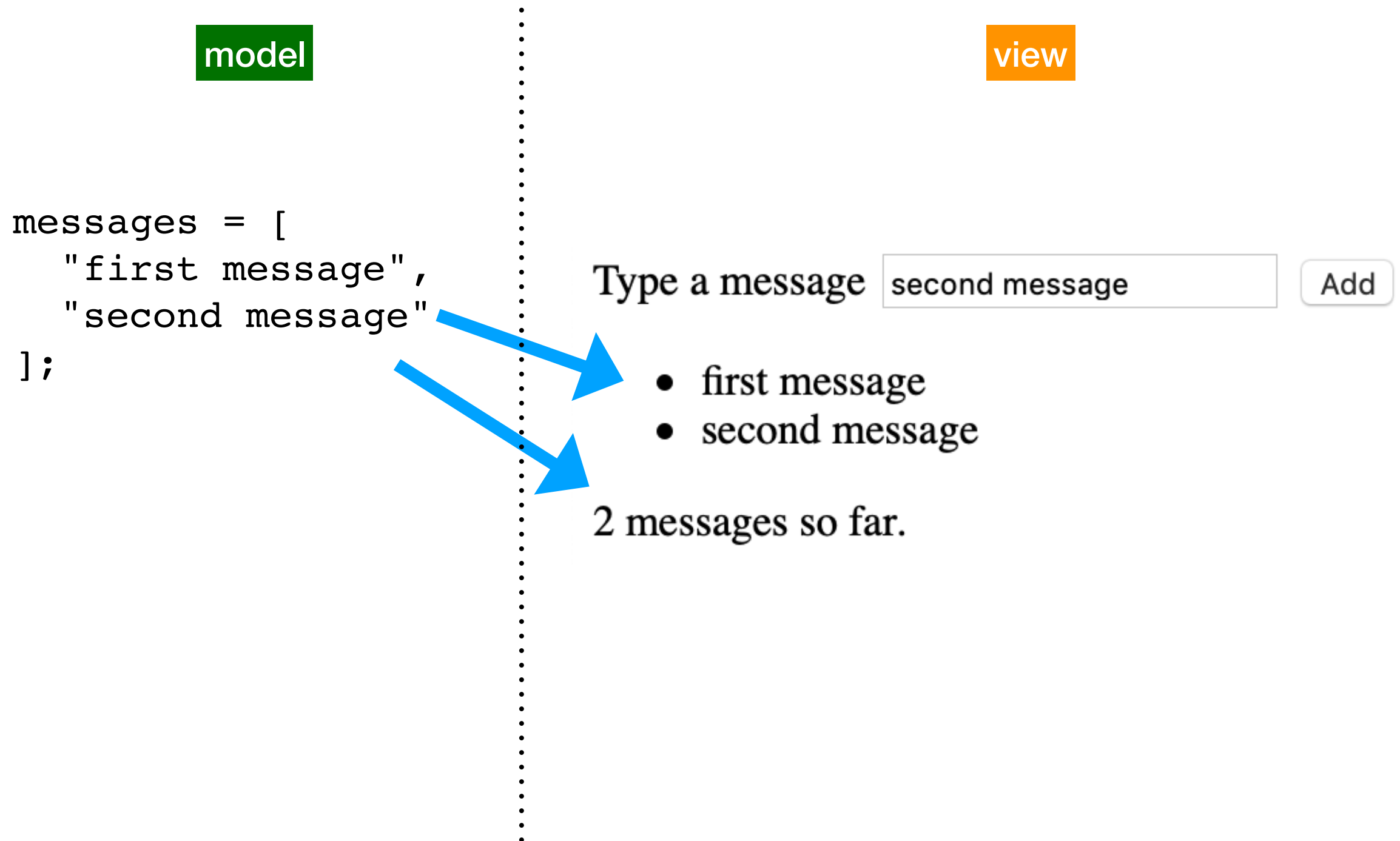
direct manipulation  
of the DOM





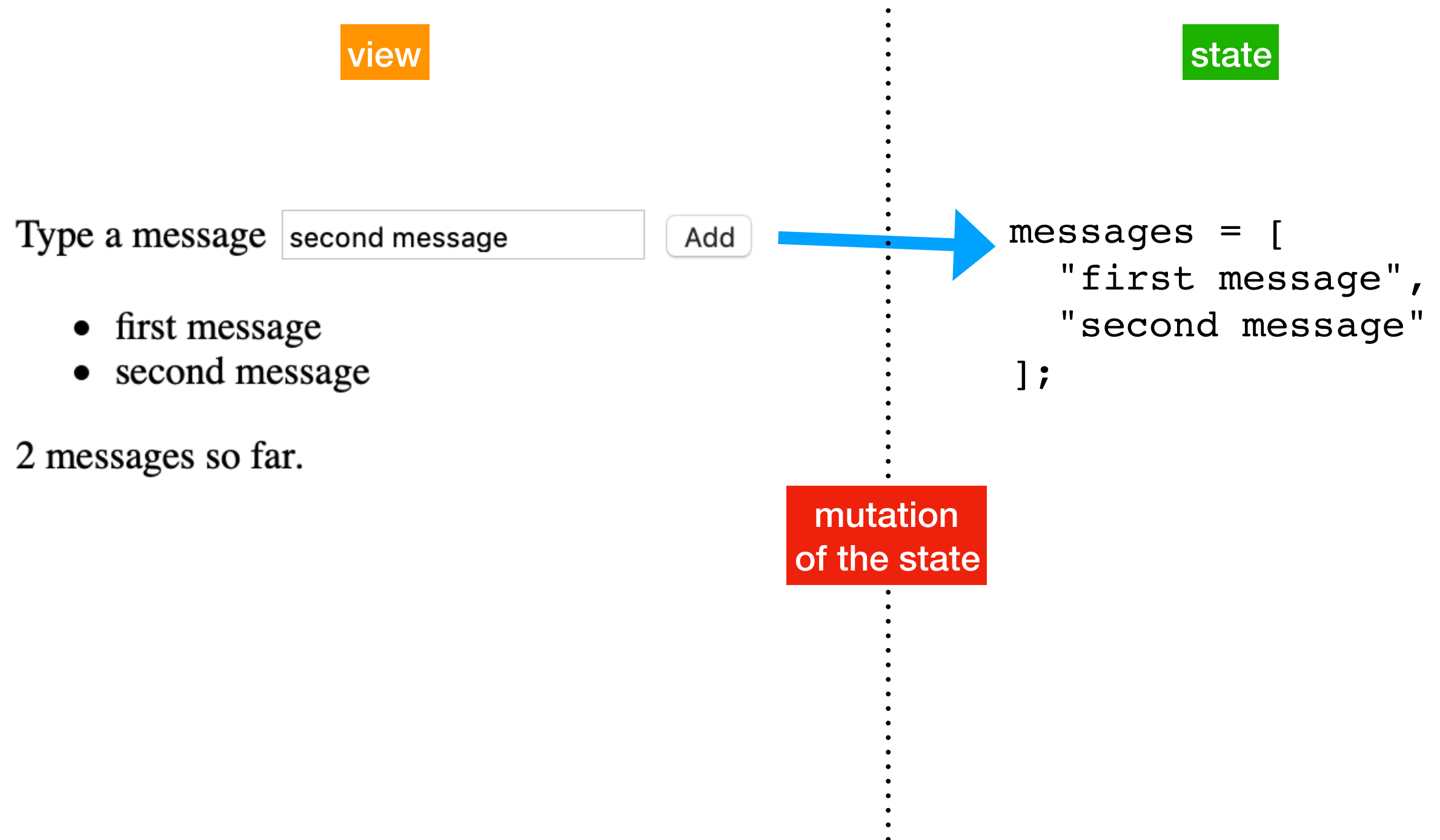
# reactive way

---



# reactive way

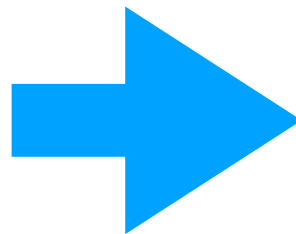
---



# defineReactive

cf. <https://github.com/vuejs/vue/blob/dev/src/core/observer/index.js>

```
data = {  
  messageToAdd: '',  
  messages: []  
}
```



```
data = {  
  messageToAdd: { ← dep1  
    get () {  
      ...  
    },  
    set (newVal) {  
      ...  
    }  
  },  
  messages: { ← dep2  
    get () {  
      ...  
    },  
    set (newVal) {  
      ...  
    }  
  }  
}
```

# reactive data gets read

```
messagesLength : function() {  
  return this.messages.length  
}
```

———— runs —————→ calls this.messages.get()

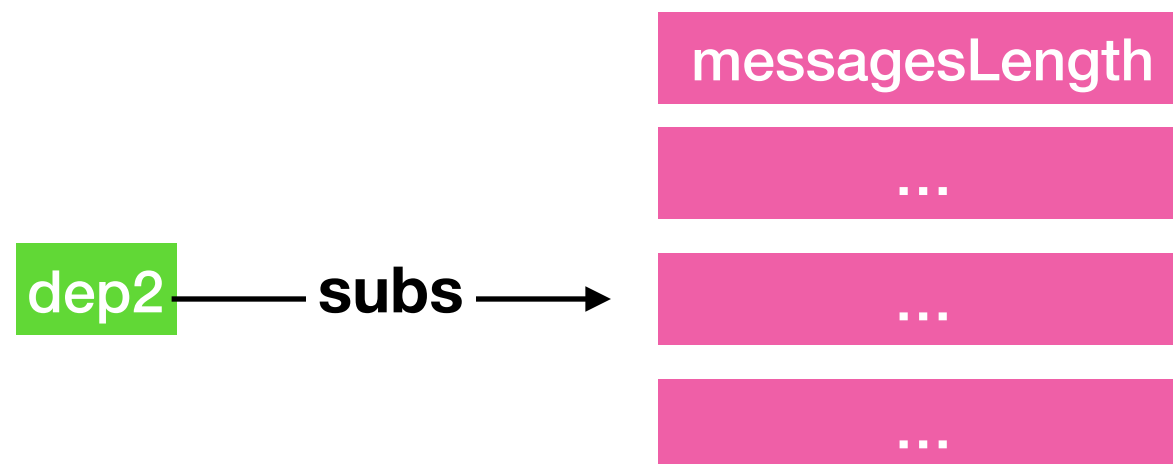
```
messages: {  
  get () {  
    dep2.addSub( messagesLength )  
    return internalValue;  
  },  
  set (newVal) {  
    ...  
  }  
};
```

comes from  
a global variable\*  
**Dep.target**

\*<https://github.com/vuejs/vue/blob/dev/src/core/observer/dep.js>

# reactive data gets read

---



# reactive data gets updated

```
this.messages.push( 'Task 2' )
```

—— runs ———→ calls `this.messages.set(...)`

```
messages: {  
  get () {  
    ...  
  },  
  set (newVal) {  
    internalValue = newVal;  
    dep2.notify()  
  }  
};
```

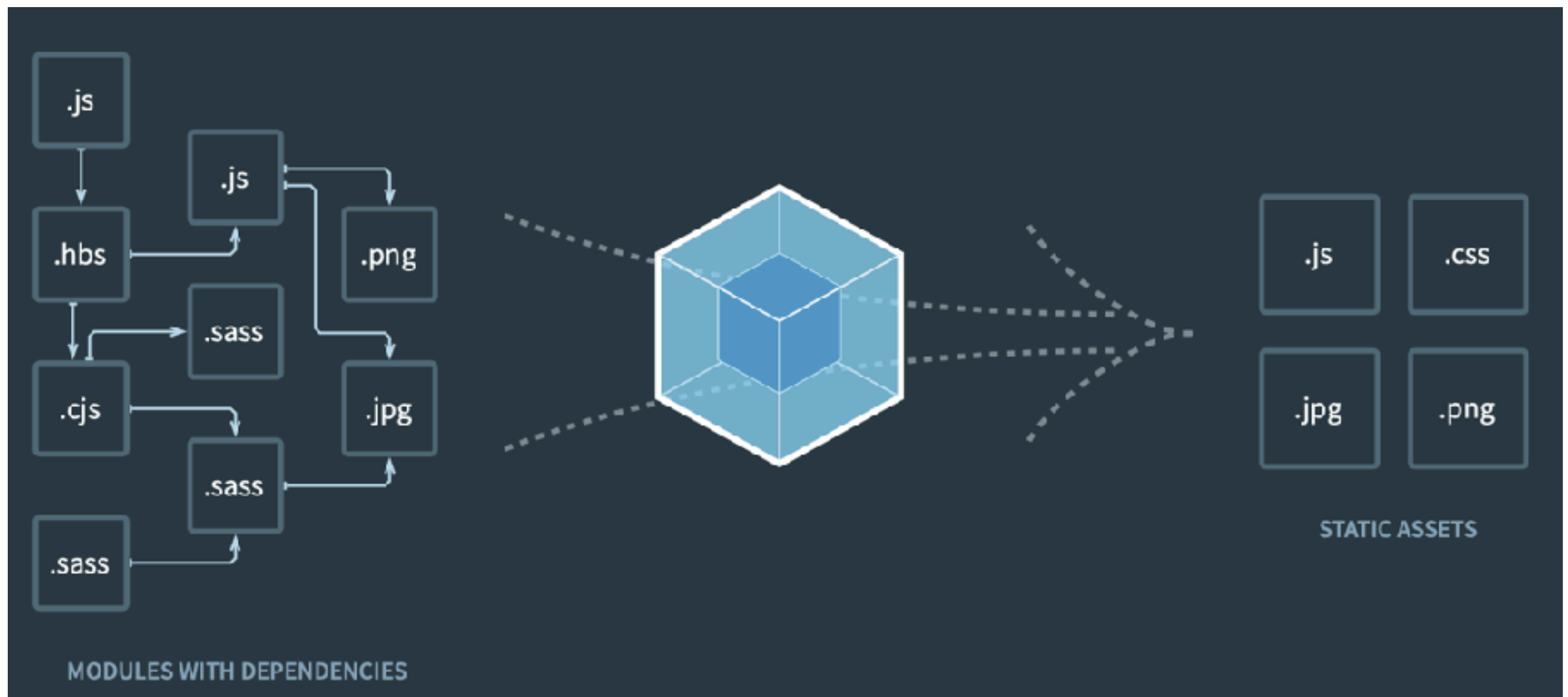
dep2 .subs

messagesLength

...

...

...



# Webpack + vue-cli



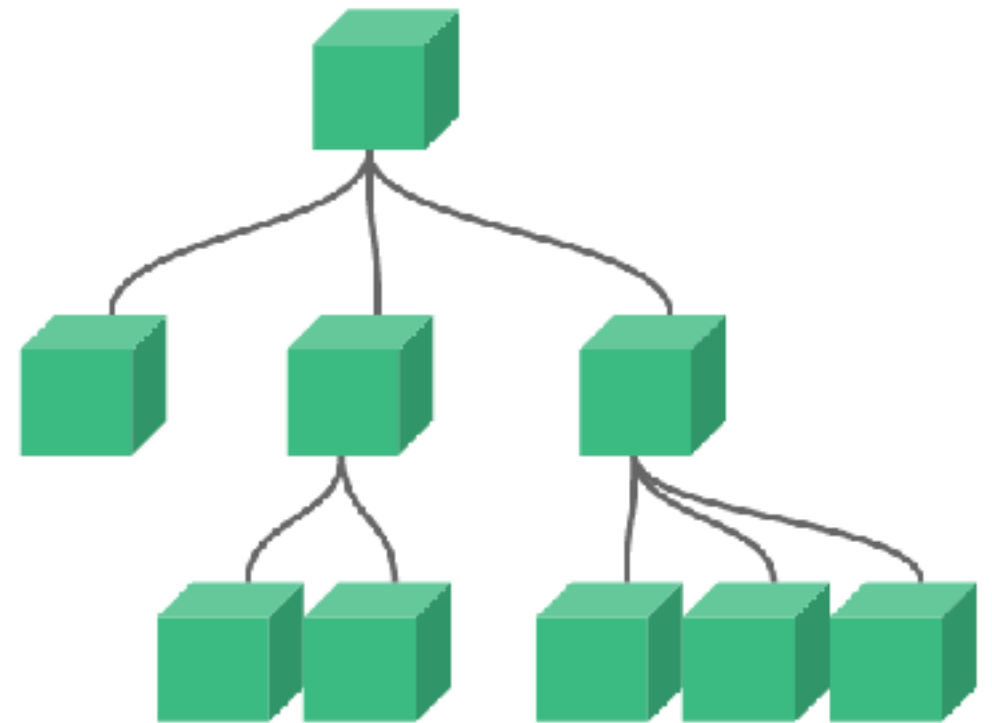
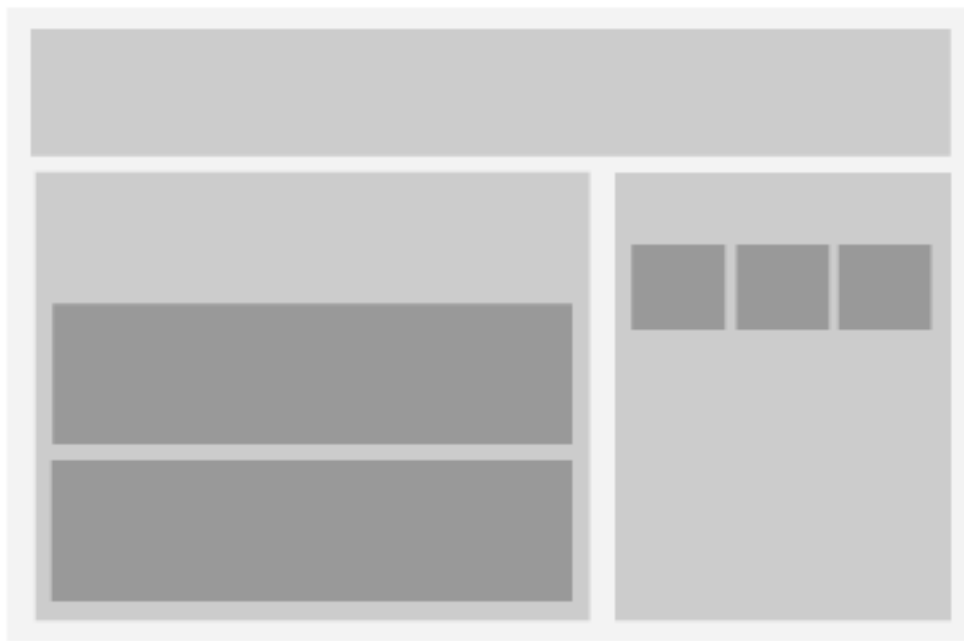
# Webpack + vue-cli

---

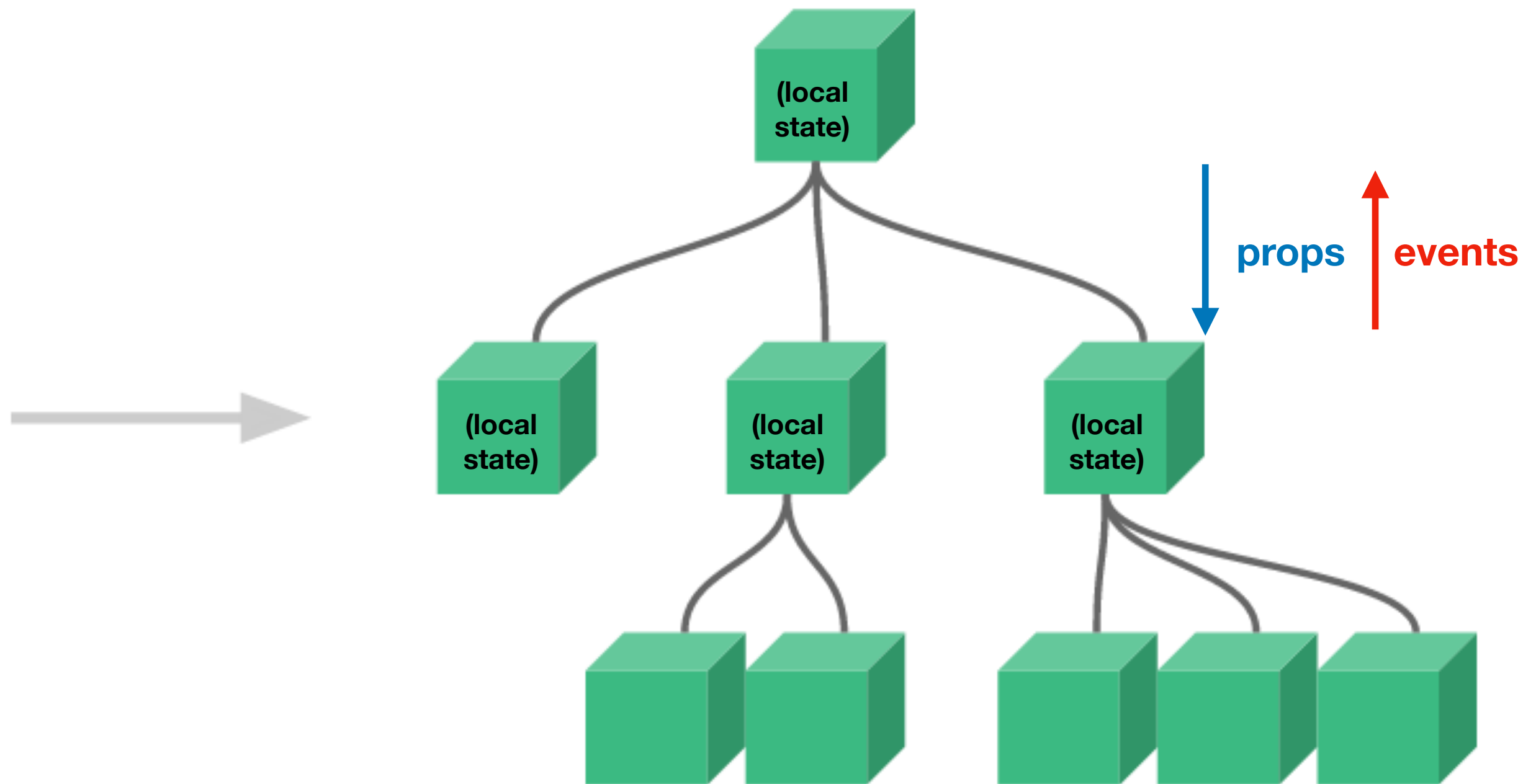
- Bundle all our dependencies in vendor.js
- Split your app in several js files and load them asynchronously when needed
- Run sass on your stylesheets, merge them all and minify it
- Compress your images, inline small images in your HTML
- Add a hash in the name of all the files for easier cache management
- Hot-reloading

# Components

---



# Components



# Components

Type a message:   component: App

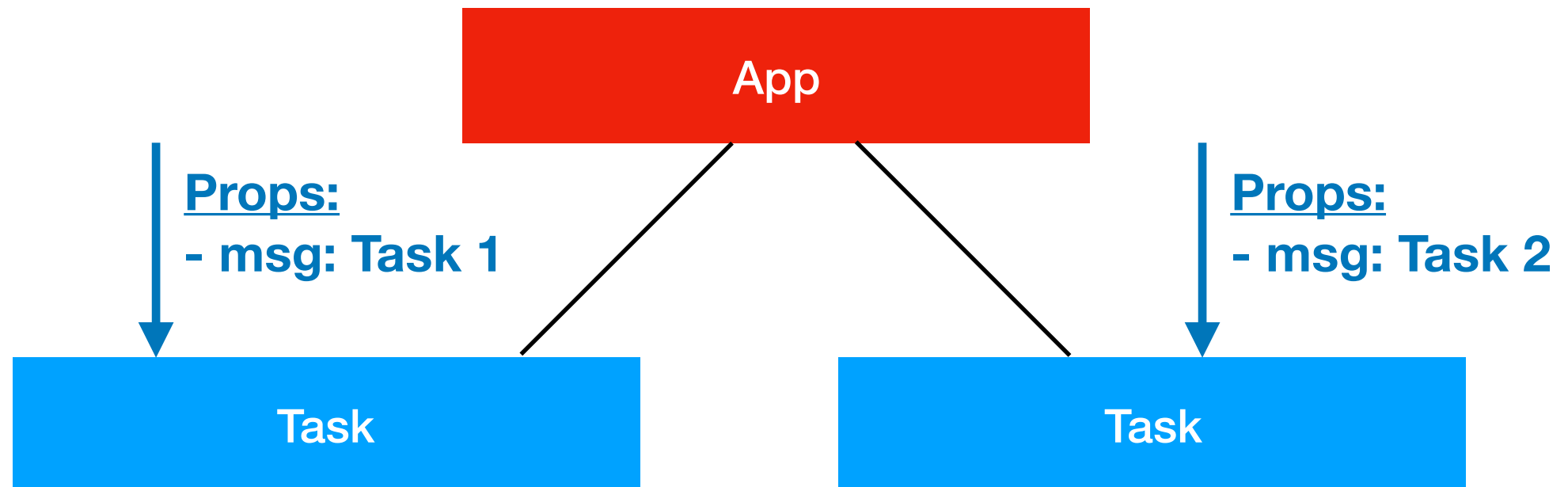
• Task 1	<input type="button" value="x"/>	<u>component: Task</u>	<u>msg: Task 1</u>
• Task 2	<input type="button" value="x"/>	<u>component: Task</u>	<u>msg: Task 2</u>

I received 2 messages so far.

# Components

---

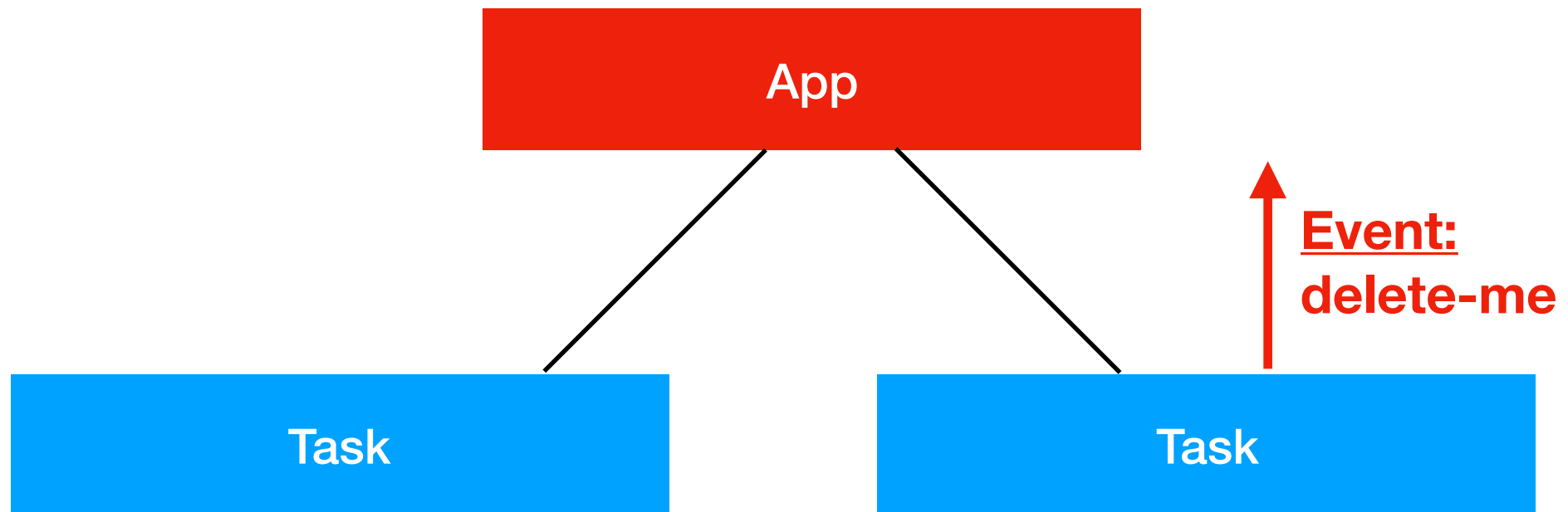
when rendering:



# Components

---

when clicking on the delete button:



# Async component loading

```
components: {  
  'Multiselect': () => import(/* webpackChunkName: "vue-multiselect" */  
    'vue-multiselect')  
}
```

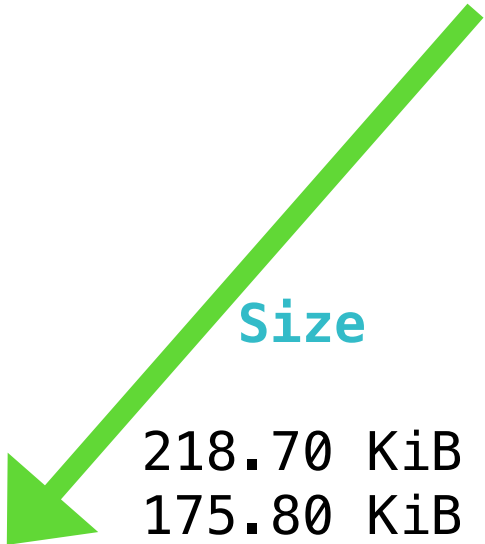
→

File

Size

Gzipped

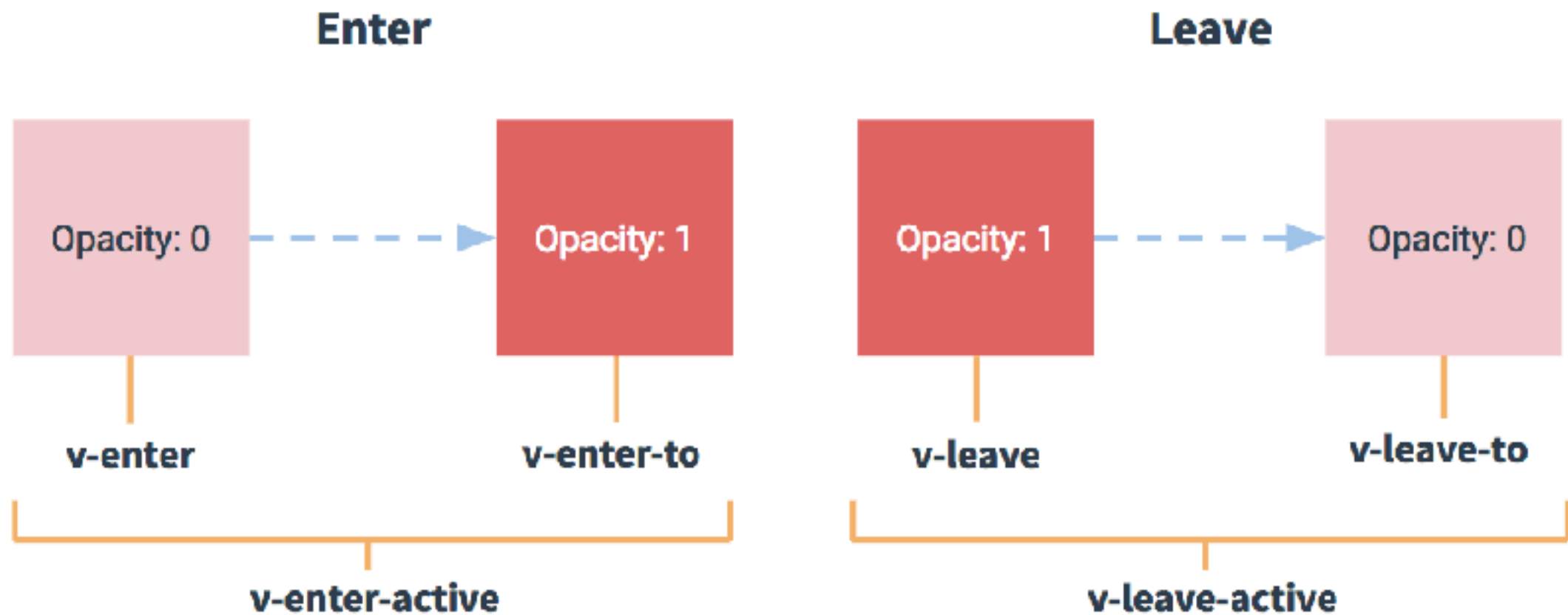
dist/js/app.6aa4f982.js	218.70 KiB	32.57 KiB
dist/js/chunk-vendors.cd89b1f3.js	175.80 KiB	61.74 KiB
dist/js/vue-multiselect.62f2b9e2.js	43.87 KiB	13.84 KiB
dist/css/app.c51b1984.css	193.92 KiB	24.36 KiB
dist/css/chunk-vendors.ee9570a6.css	7.97 KiB	1.77 KiB





# Transitions

---



# Conclusion

---

- Elegant and powerful framework if you adhere to its philosophy
- Smaller learning curve, scales to big applications
- Easy way to modernise legacy application (jQuery, AngularJS 1, ...)
- Used by big companies: Gitlab, Alibaba, IBM, Adobe, Nintendo, etc..

# Conclusion

---

- More jobs/developers for React
- Bigger ecosystem on React
  - React Native -> vue-native, weex, NativeScript-vue
  - Next.js -> Nuxt.js, Quasar Framework
- Competition is good
  - Vue gets inspired from React (for example Hooks) and vice-versa
  - Helps keep Facebook in check :-)

# Interesting links

---

- <https://github.com/vuejs/awesome-vue>
- <https://www.monterail.com/state-of-vuejs-report>
- <https://twitter.com/youyuxi>
- [https://twitter.com/sarah\\_edo](https://twitter.com/sarah_edo)
- <https://twitter.com/VueNewsletter>



# Thank you!

[sami@boudrai.com](mailto:sami@boudrai.com)