

# A *Vue.js* Perspective





Raju Gandhi  
@looselytyped

# WHO *is using Vue?*



WHY?



Vue (pronounced /vju:/, like view) is a **progressive framework** for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be **incrementally adoptable**. The core library is focused on the **view layer only**, and is easy to pick up and integrate with other libraries or existing projects

<https://vuejs.org/v2/guide/index.html>

(POLL - Multiple choice)

# Experience with other technologies

- Angular
- React
- Ember.js
- JQuery / Backbone / Spine / Knockout / Other

# Vue

Vue is also perfectly capable of powering sophisticated **Single-Page Applications** when used in combination with modern tooling and supporting libraries.

<https://vuejs.org/v2/guide/index.html>

THINKING IN  
*Components*

friends-hq

localhost:8080/people/

# Friends HQ

Together, we are stronger

Dashboard

Contacts

Journal

Michelle Mulroy

Venkat Subramanian

Matt Forsythe

Nate Schutta

ADD FRIEND

© Looselytyped 2019

# <App />

The screenshot shows a web application window titled "friends-hq" at "localhost:8080/people/". The interface has a dark purple header bar with a "Friends HQ" logo and a "Together, we are stronger" tagline. A sidebar on the left contains "Dashboard", "Contacts", and "Journal" links. The main content area displays a list of contacts with names and small heart icons:

Name	Action
Michelle Mulroy	Heart icon (grey)
Venkat Subramanian	Heart icon (red)
Matt Forsythe	Heart icon (grey)
Nate Schutta	Heart icon (grey)

A green "ADD FRIEND" button is located in the top right corner of the contact list area. The footer of the page includes a copyright notice: "© Looselytyped 2019".

Friends HQ

Together, we are stronger

Dashboard

Contacts

Journal

localhost:8080/people/

# <PeopleList />

Michelle Mulroy	
Venkat Subramanian	
Matt Forsythe	
Nate Schutta	

ADD FRIEND

© Looselytyped 2019

Friends HQ

Together, we are stronger

Dashboard

Contacts

Journal

localhost:8080/people/

# <PersonItem />

Michelle Mulroy	
Venkat Subramanian	
Matt Forsythe	
Nate Schutta	

ADD FRIEND

© Looselytyped 2019

Let us write a  
*Vue App!*

# *Single File* **COMPONENTS**

```
<template>
  <v-container>
    Hello {{ name }}
  </v-container>
</template>
```

```
<script>
export default {
  data() {
    return {
      name: "Vue.js"
    }
  }
};
</script>
```

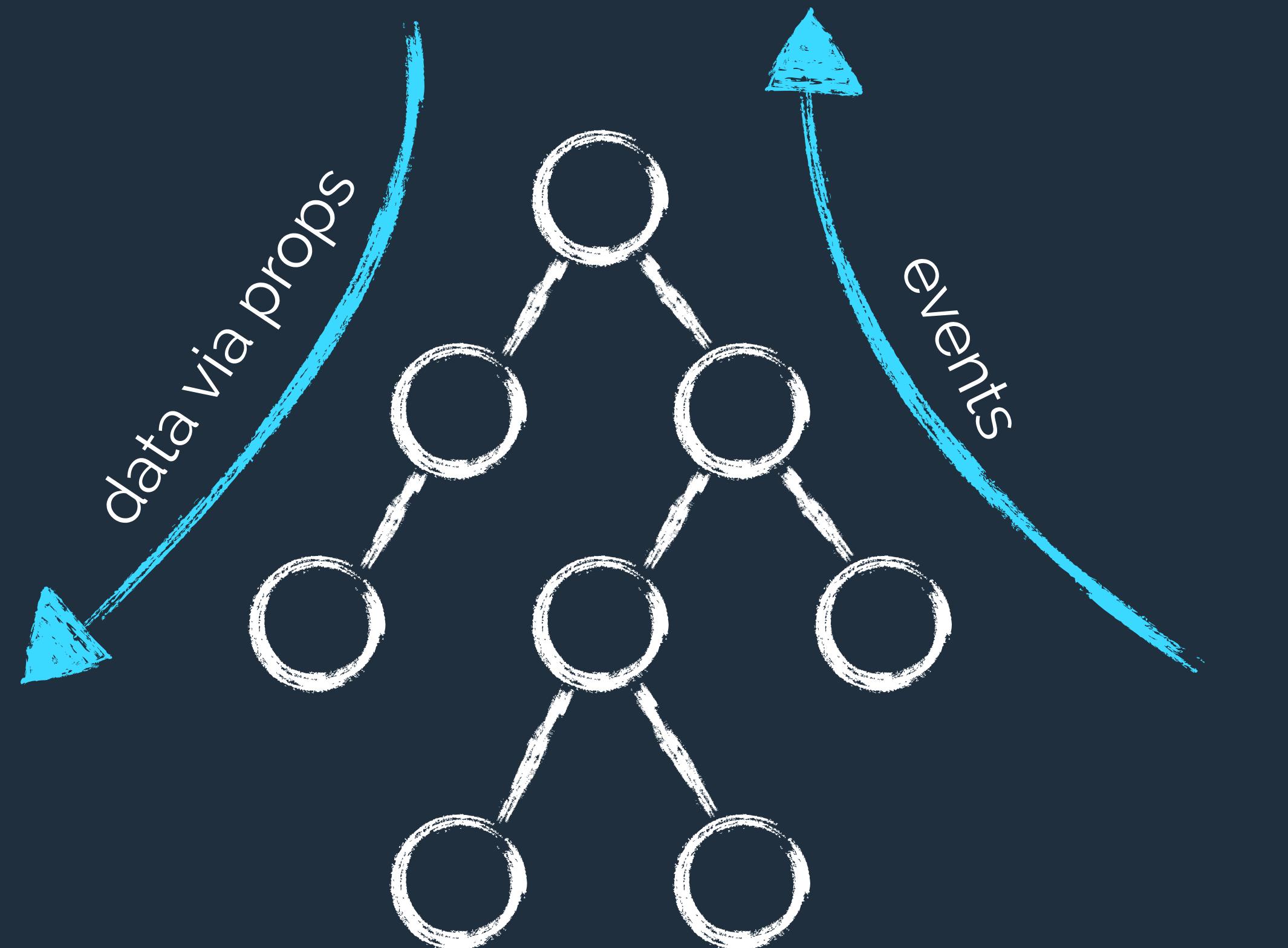
```
<style>
h1 {
  font-size: 2em;
  text-align: center;
}
</style>
```

```
<template>
  <v-container>
    Hello {{ name }} ← visible to the view
  </v-container>
</template>
```

```
<script>
export default {
  data() {
    return {
      name: "Vue.js" → component "state"
    }
  }
} ;
</script>
```

```
<style>
h1 {
  font-size: 2em;
  text-align: center;
}
</style>
```

*Props*  
-&-  
**Events**



# *Unidirectional* DATAFLOW

```
<PersonItem
  v-for="(friend, index) in friends"
  :key="friend.id"
  :friend="friend"
  :last="index + 1 < friends.length"
  @notify-parent="like"
></PersonItem>
```

```
<PersonItem
  v-for="(friend, index) in friends"
  :key="friend.id"
  :friend="friend"
  :last="index + 1 < friends.length"
  @notify-parent="like"
></PersonItem>
```

 → props

```
<PersonItem
  v-for="(friend, index) in friends"
  :key="friend.id"
  :friend="friend"
  :last="index + 1 < friends.length"
  @notify-parent="like"
></PersonItem>
```



events

# Filters



```
export default {
  // other attributes ...
  filters: {
    uppercase: value => {
      if (!value) return "";
      return value.toUpperCase();
    }
  }
};
```

```
<v-list-title class="grey--text">
  {{ item.text | uppercase }}
</v-list-title>
```

*Mixins*



```
const logger = {  
  methods: {  
    log(msg) {  
      // do something here  
    }  
  }  
};
```

```
export default {  
  mixins: [logger],  
  // other attributes here,  
  methods: {  
    click() {  
      this.log("I was clicked");  
    }  
  }  
};
```

# Lifecycle



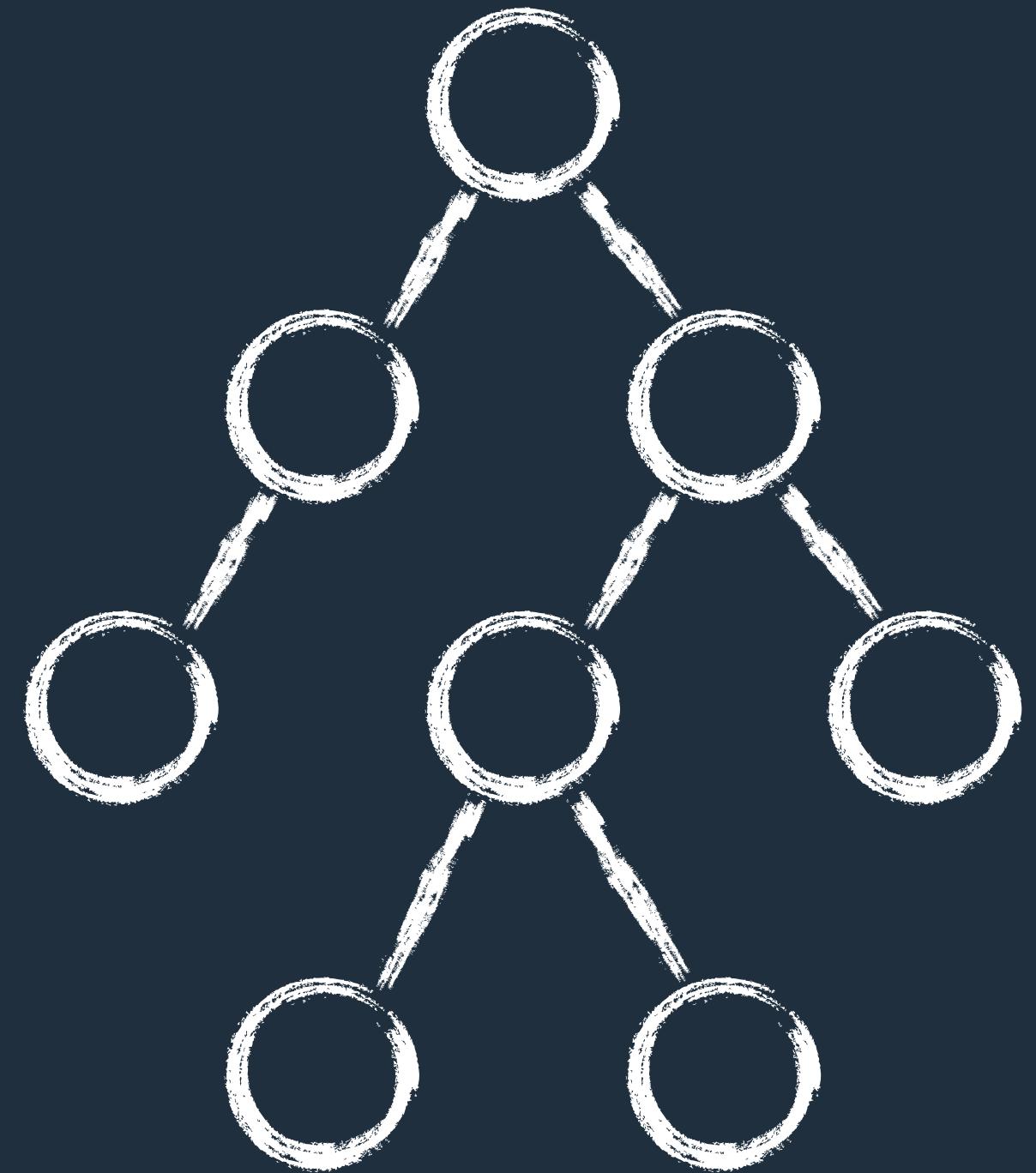


A *Vue*  
**COMPONENT**

```
export default {  
    // child components  
    components: { /** */ },  
    // parent supplied props, can be an object literal  
    props: [ /** */ ],  
    // component state  
    data() { /** */ },  
    // a "view" of the data  
    computed: { /** */ },  
    // means to manipulate data  
    methods: { /** */ },  
    // react to data changes  
    watch: { /** */ },  
    // text transformation / formatting  
    filters: { /** */ },  
    mixins: [ /** */ ],  
    // lifecycle hooks  
    created() { /** */ },  
    mounted() { /** */ },  
};
```

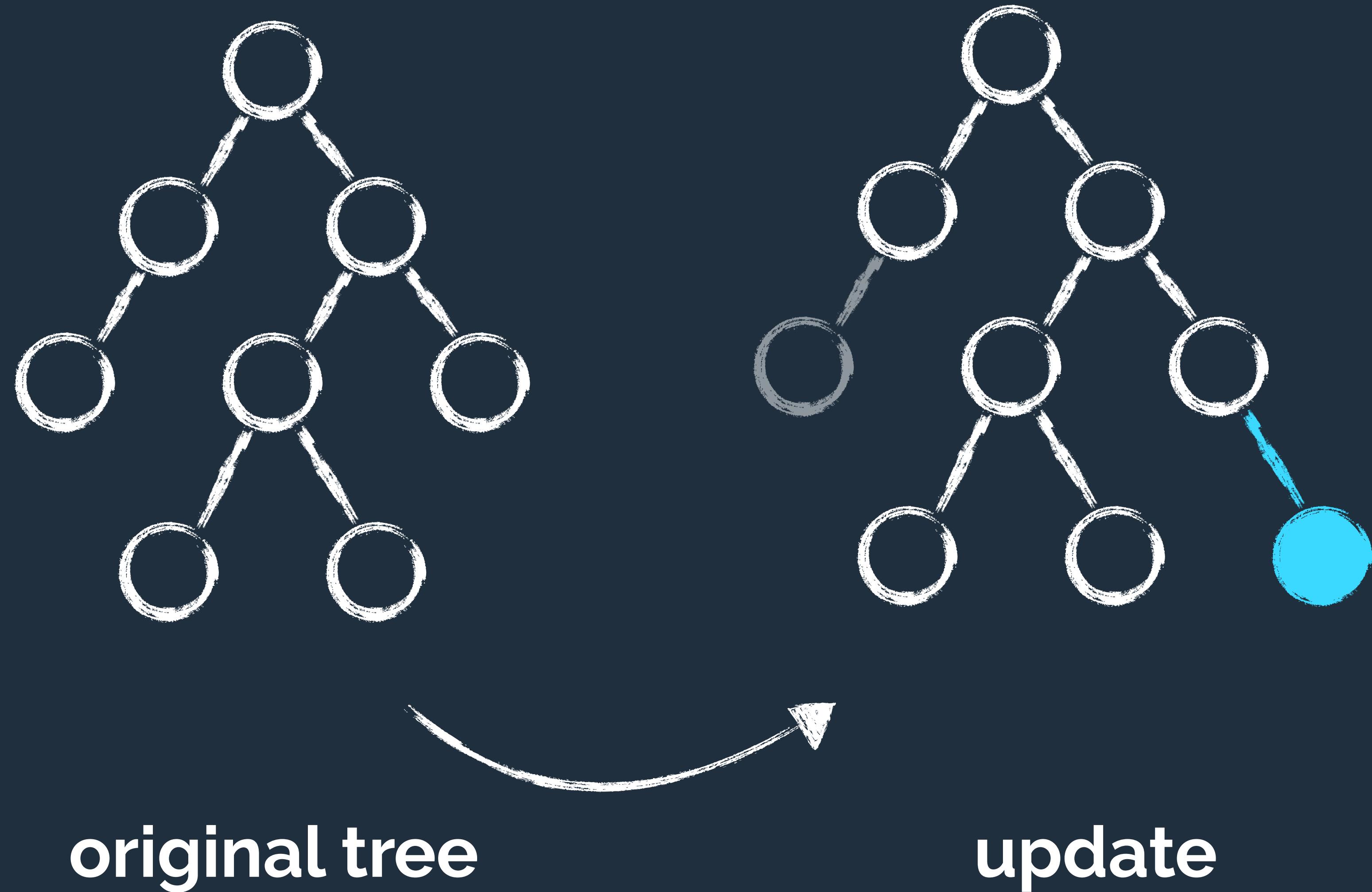
*Virtual*  
**DOM**

# *Virtual* DOM

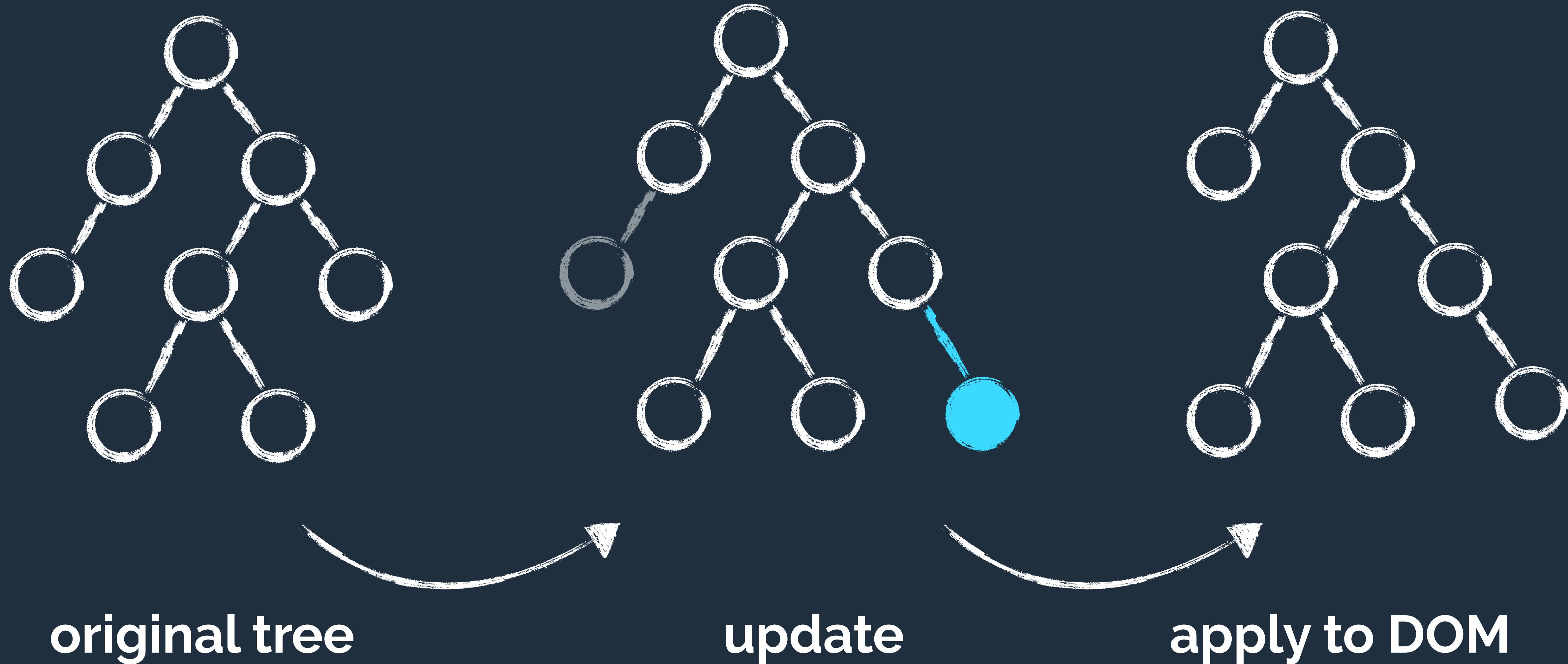


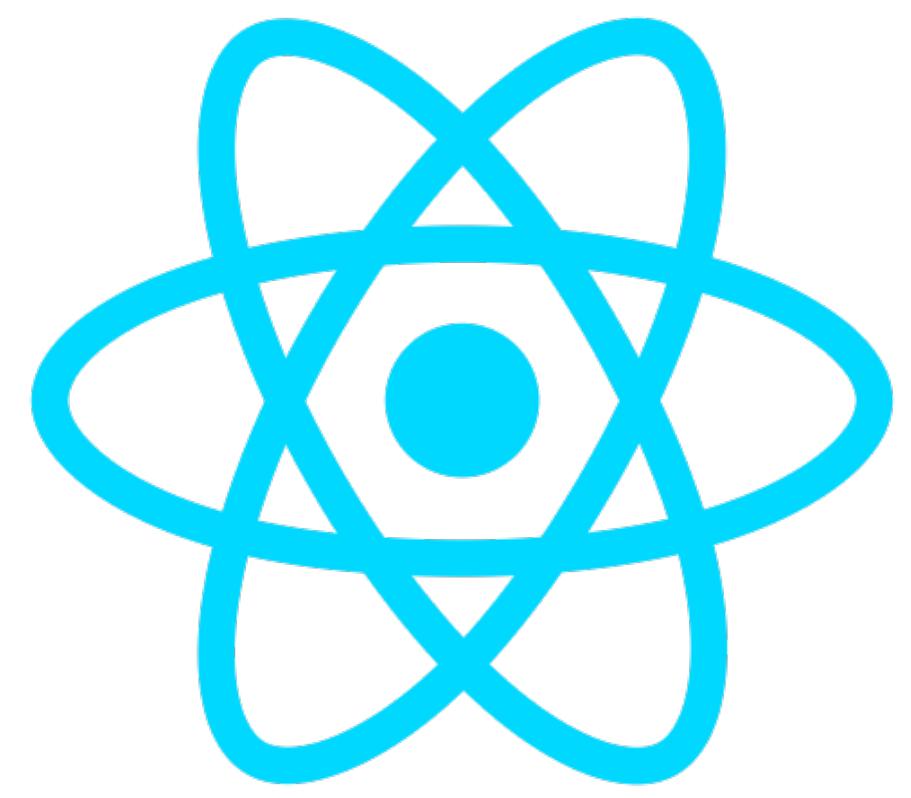
original tree

# *Virtual* DOM



# *Virtual* DOM





**library**



**framework**

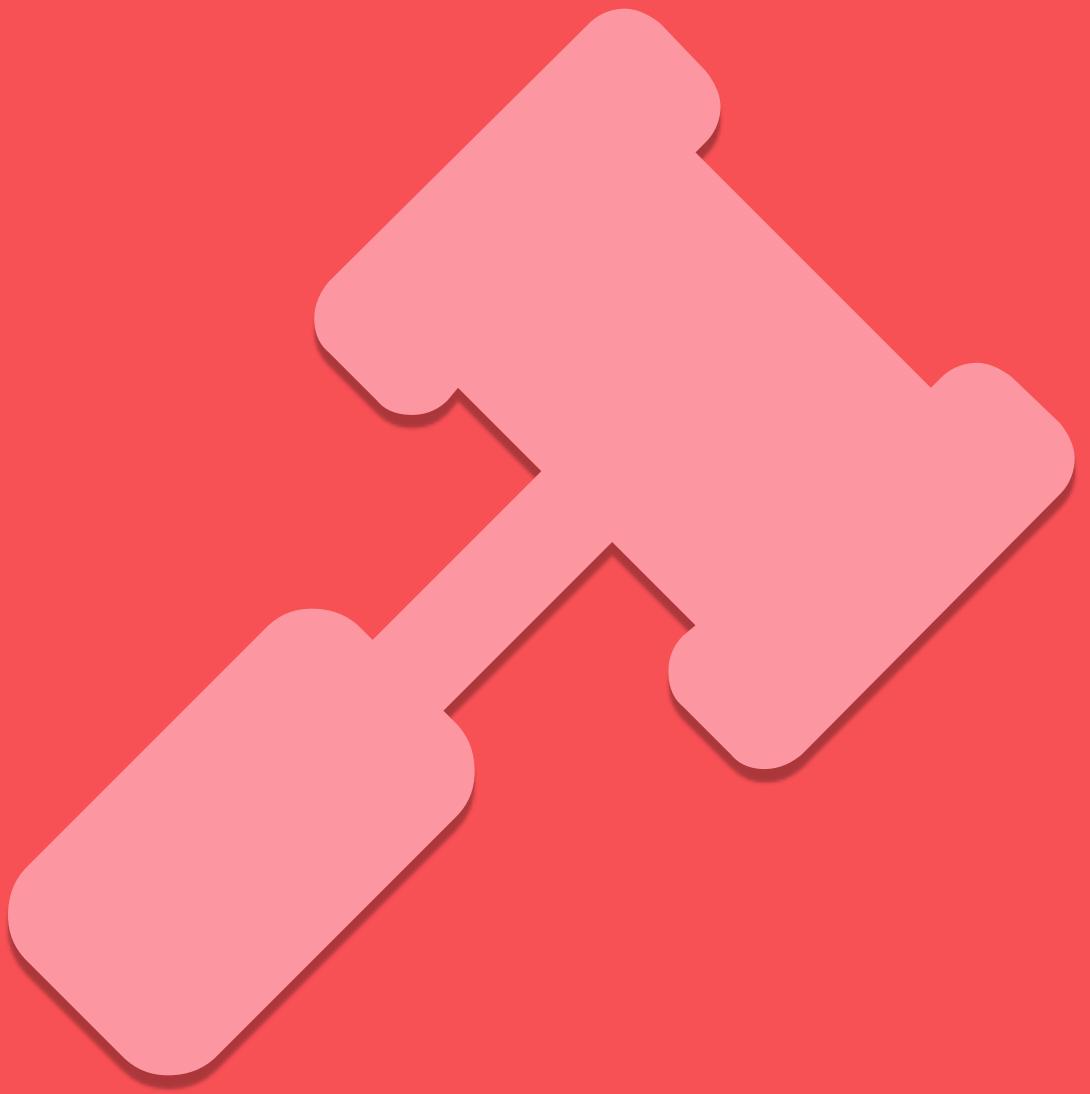
# THE *Ecosystem*



**IT'S A**  
*Vue-tivus!*

vuex  
vue-router  
vue-devtools  
vuepress  
vue-cli  
ui  
nuxt  
vuetify

THE  
*Tools*



*Vetur*

**syntax-highlighting  
snippet  
emmet  
linting / error checking  
formatting  
auto completion  
debugging**

*vue-cli*

project scaffolding  
plugins  
rapid prototyping

*THANK*  
You!

# Resources

[Vue.js Docs](#)

[Vue.js in Action](#)

[Testing Vue.js Applications](#)

# *Credits*

Theme

Cover Slide **Background**