



WEB TECHNOLOGY COMPARISON

Intervenants :
Mathias Cousté
Rémi Pourtier

 **accenture**technology



Vue.js



Angular





Vue.js



Dan Abramov

@dan_abramov

Following



ComponentConf should be a thing. JS umbrella is too generic and I want to zoom in a little bit. But React/Vue/Angular/Ember communities are too isolated and often don't realize they talk about the same problems. Need a common space.

12:30 AM - 23 Sep 2018

Angular



COURS

_ Tous les mardis sauf le 2 Décembre: 1 h cours + 1h30 TD

_ 4 Novembre: Vue.js basics

_ 18 Novembre: Data management

_ 25 Novembre: Comparison canvas

_ 2 Décembre: Evaluations

EVALUATIONS

_ 2 Décembre - Evaluation Orale : 50 %

_ 6 Décembre à 20h - Rendu de l'article : 50%

EVALUATION ORALE – THE CONFERENCE

- _ Par groupe de 4 + 2 groupes de 3 – constitution des groupes
le **24 Novembre**
- _ **11 minutes** de presentation & **3 minutes** de questions
- _ But : Comparer **Vue.JS VS Angular**
- _ Basée sur le **canvas de comparaison**
- _ Chaque personne présente plusieurs critères de comparaison

CANVAS DE COMPARAISON

	VueJS	Angular
Les principaux concepts		
Les design patterns		
Comment les données sont-elles stockées ?		
Où se fait la logique métier ?		
Comment la vue accède aux données ?		
Comment la vue se met à jour ?		
Comment se fait la gestion du style ?		
Comment sont gérés les dépendances d'un composant ? (modules, import de composants etc..)		

	VueJS	Angular
Code guidelines du framework		
Maintenance		
Réutilisation		
Coût de développement		
Tests automatisés (unit tests, end to end tests)		

Etc...

RAPPORT – THE ARTICLE

Publish



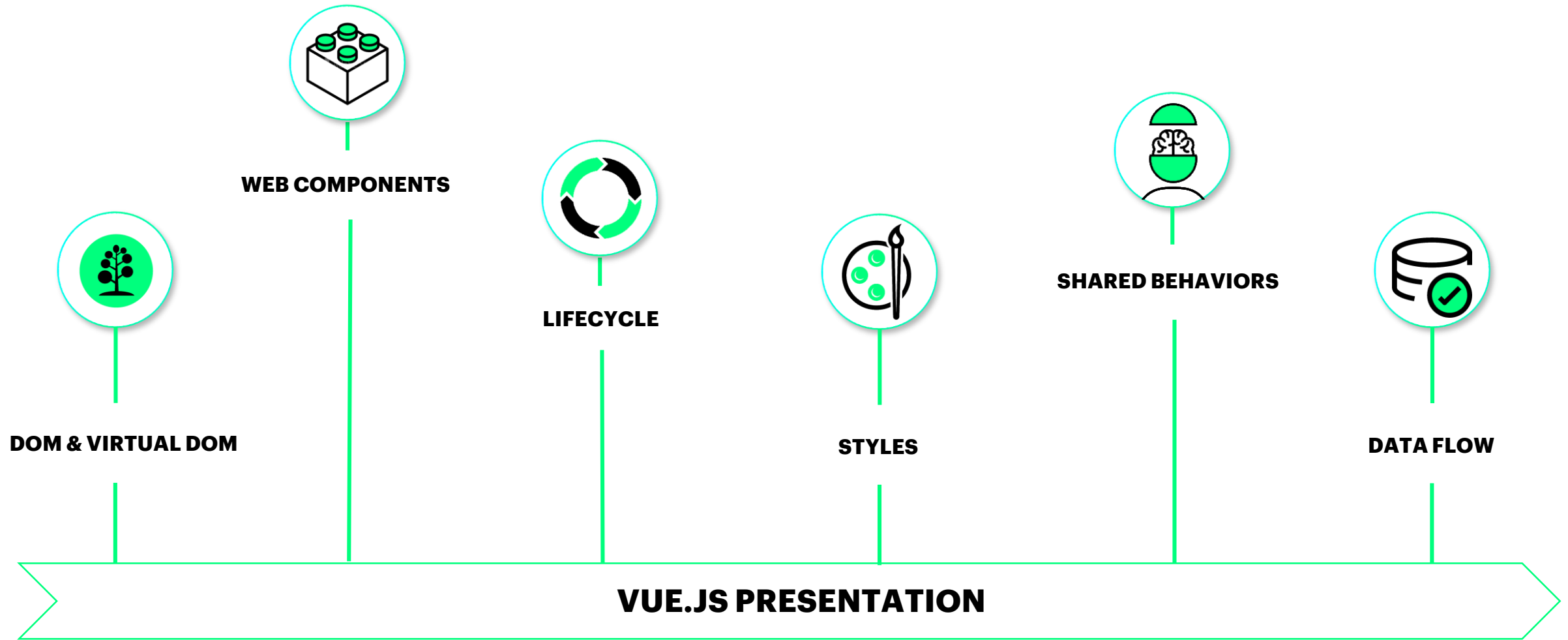
Title

LEARNING VUE.JS KNOWING
ANGULAR — My experience

Tell your story...

- _ But : **Prise de recul** sur votre experience avec Vue.JS
- _ Les **forces** et les **faiblesses** du framework
- _ En utilisant
 - Des exemples de votre code Vue.JS
 - Votre experience avec d'autres frameworks web
- _ Format: En mode "**article**", 2 pages maximum (sans les images)
- _ Rendu individuel

AGENDA



A large, stylized Vue.js logo consisting of two overlapping 'V' shapes. The outer 'V' is a medium green color, and the inner 'V' is a darker blue color.

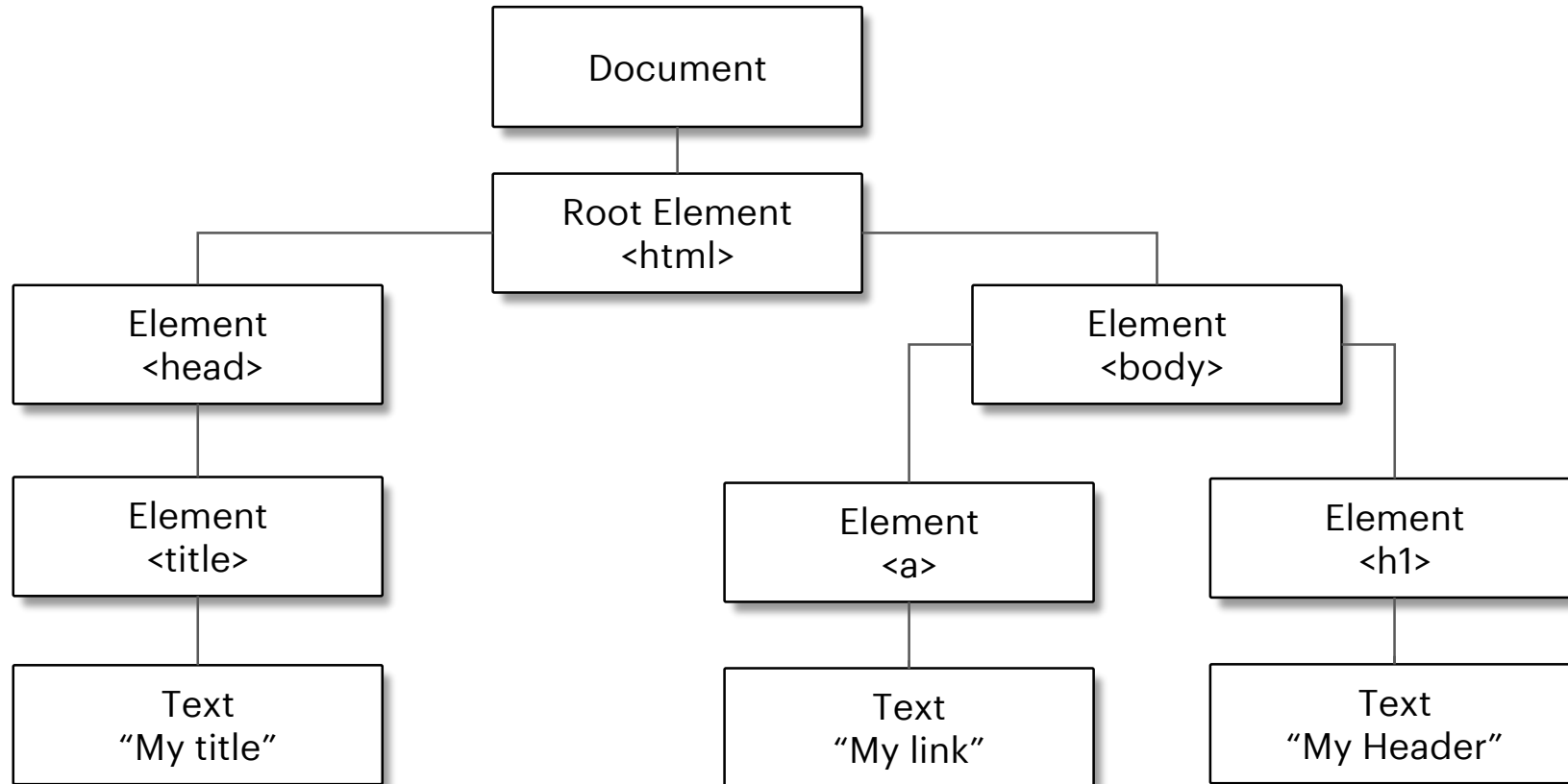
VUE.JS

Vue.js

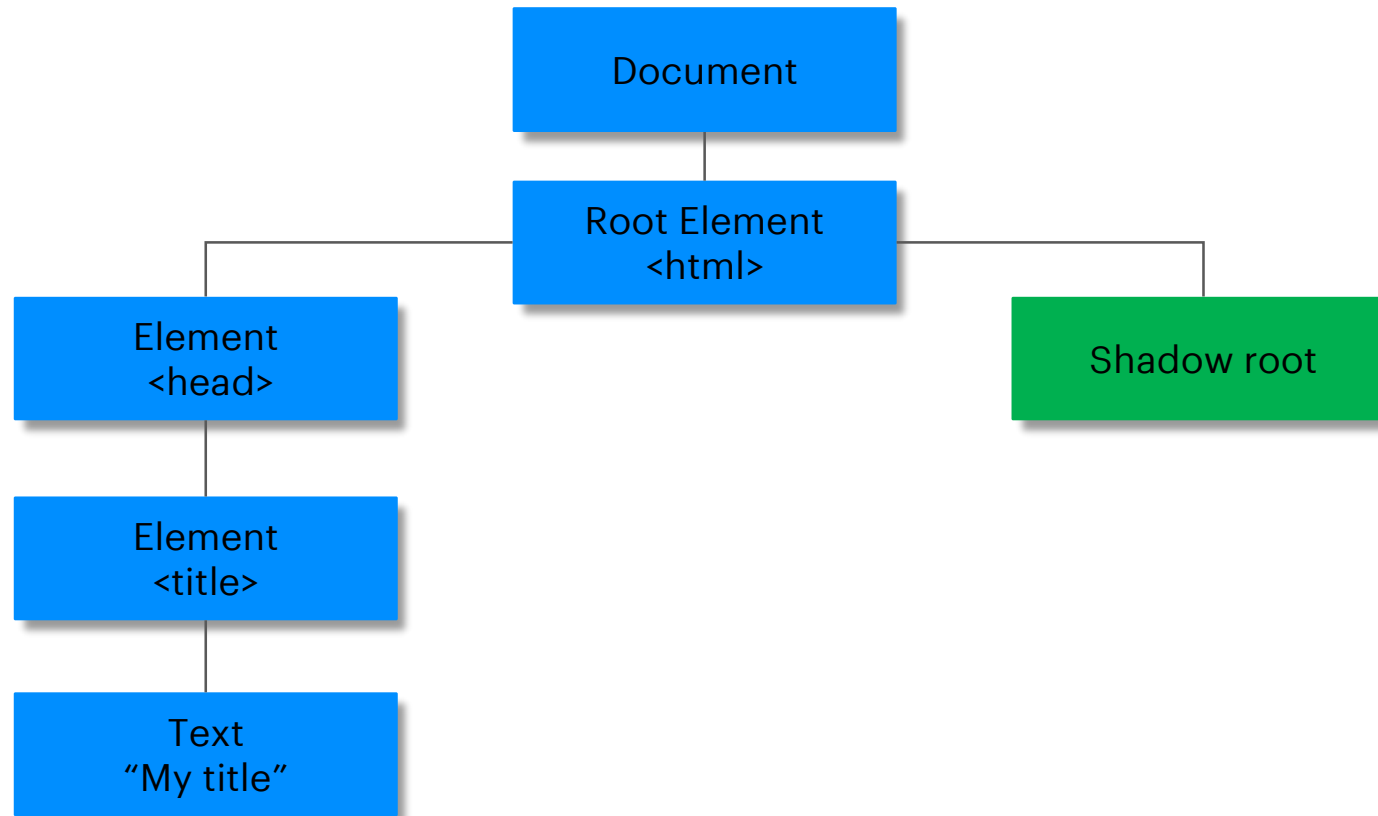
A low-angle, upward-looking photograph of a tree. The dark, textured trunk of the tree is at the bottom center, with many thin, dark branches radiating outwards and upwards. The branches are covered in dense, bright green leaves. The background is a clear, light blue sky. The text 'DOM, SHADOW DOM & VIRTUAL DOM' is overlaid in the center, with 'SHADOW' and 'VIRTUAL' in green and 'DOM' in white. A thin white horizontal line is positioned to the left of the text.

DOM, SHADOW DOM & VIRTUAL DOM

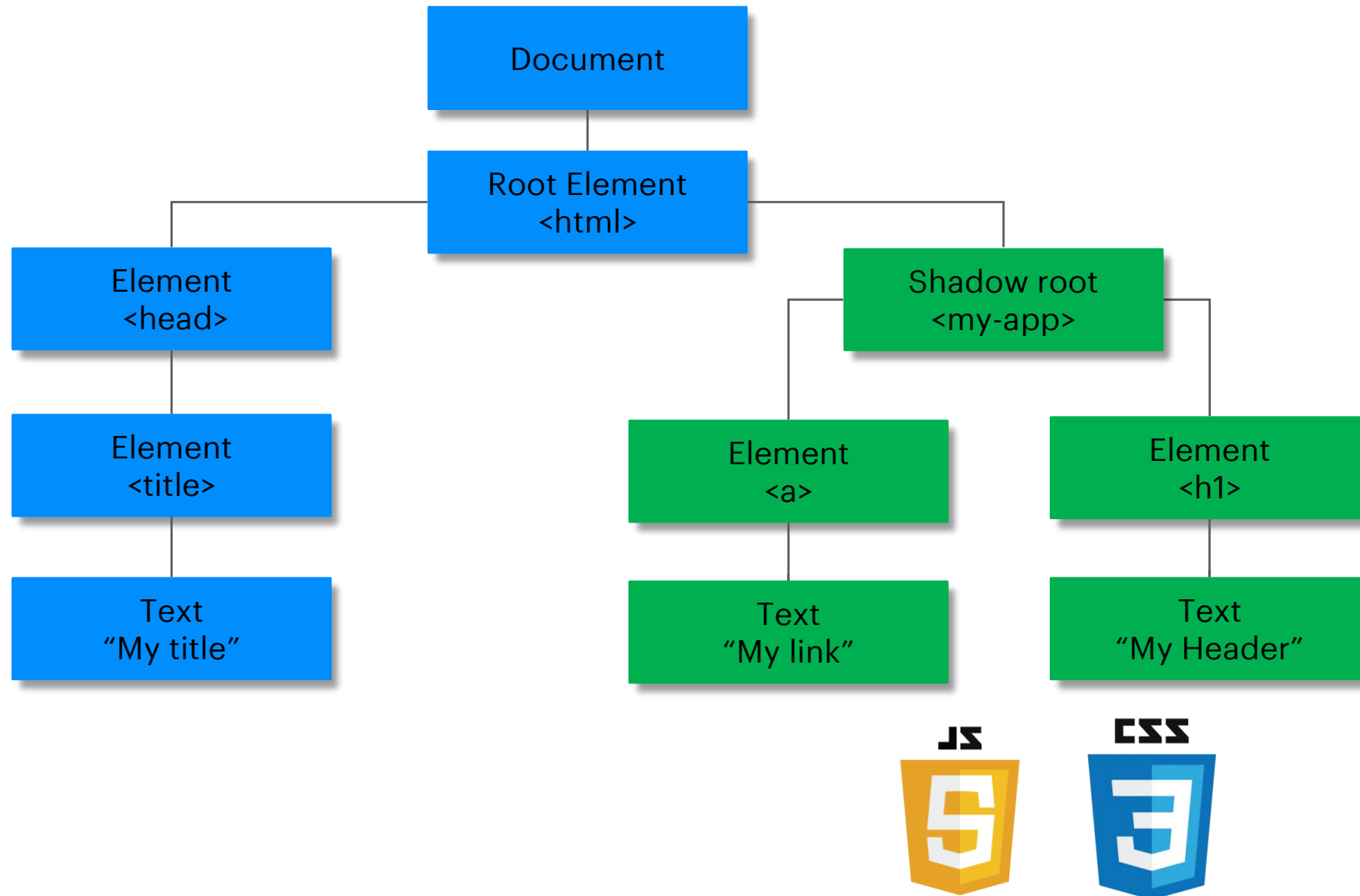
DOM – DOCUMENT OBJECT MODEL



SHADOW DOM

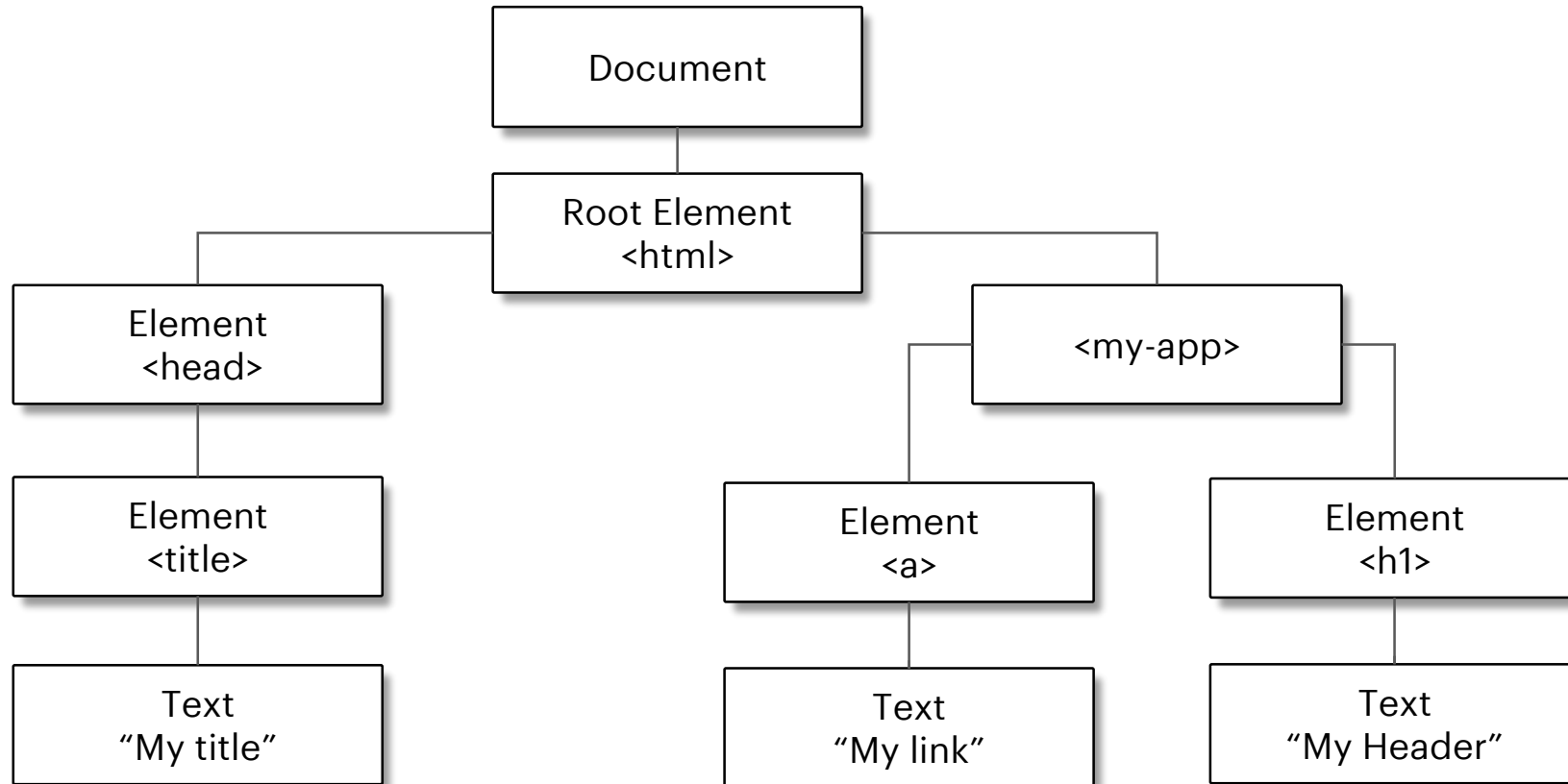


SHADOW DOM



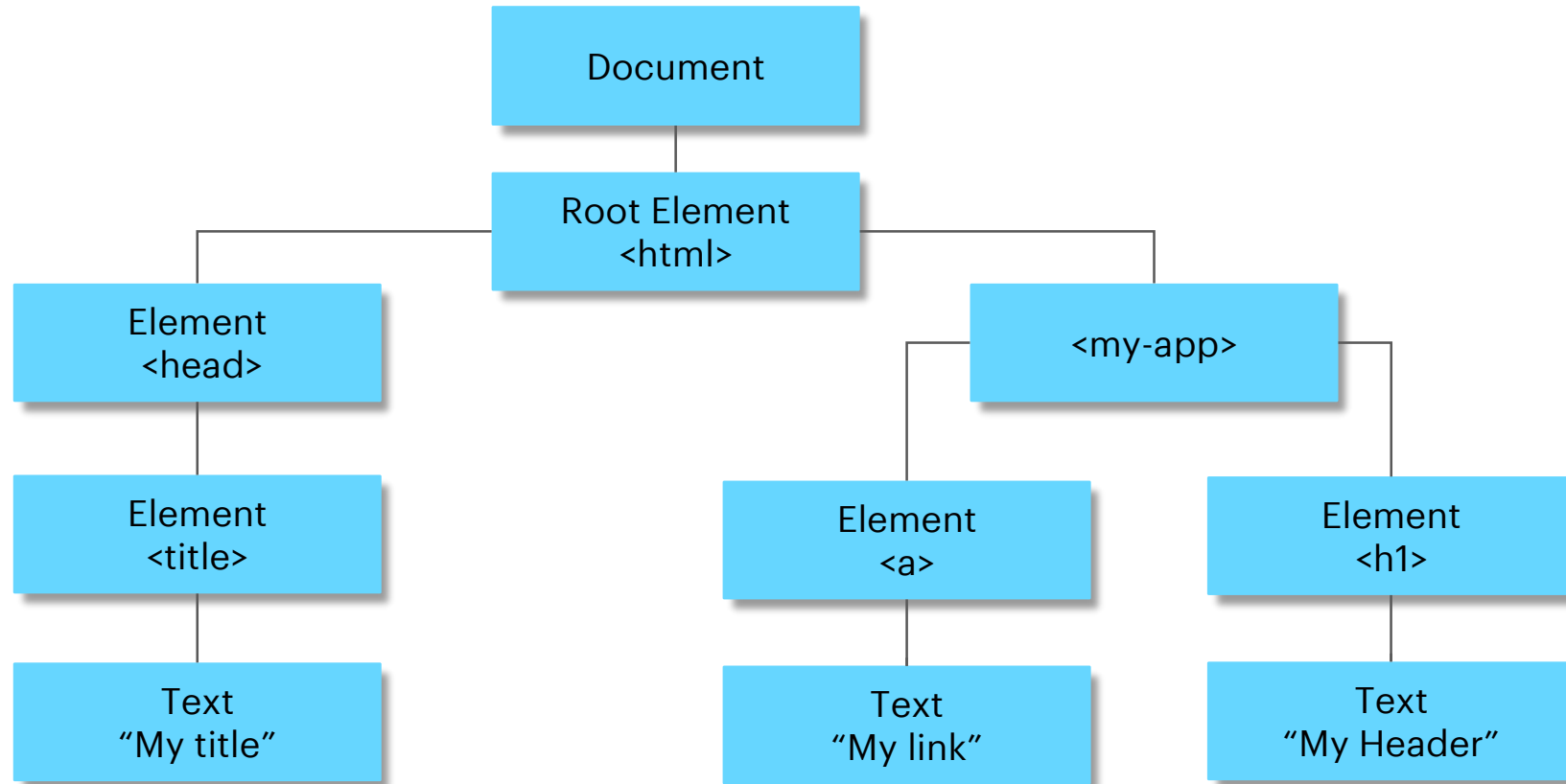
VIRTUAL DOM

Page DOM



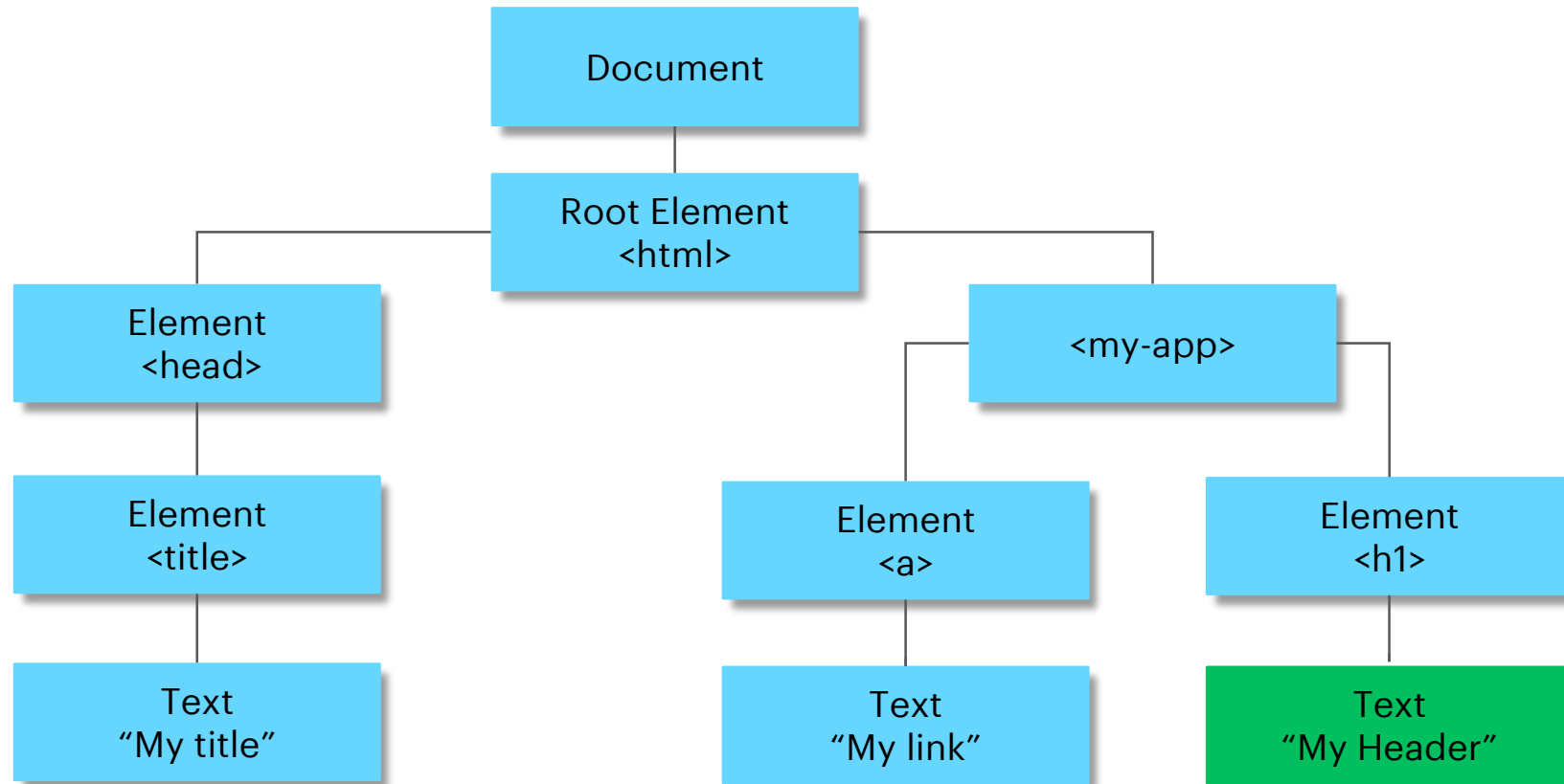
VIRTUAL DOM

Virtual DOM = DOM copy



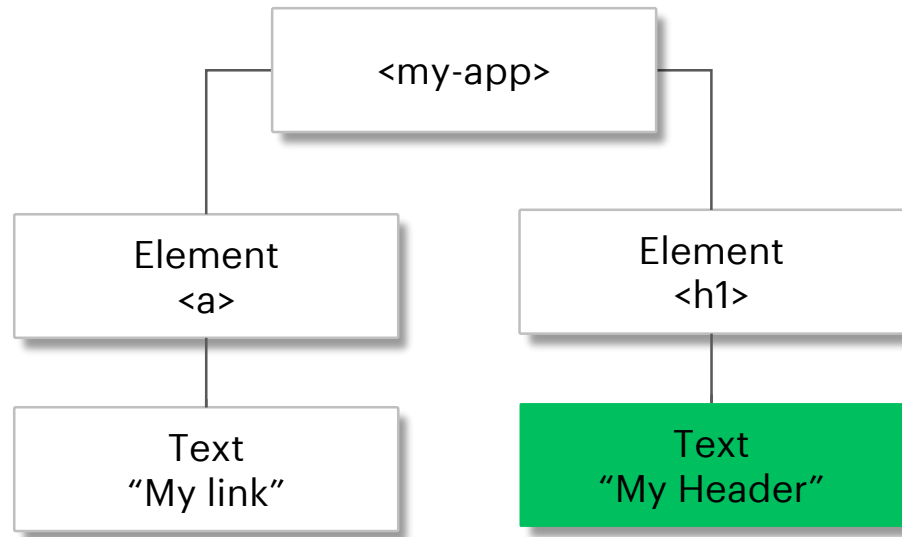
VIRTUAL DOM

How does Vue apply the change?

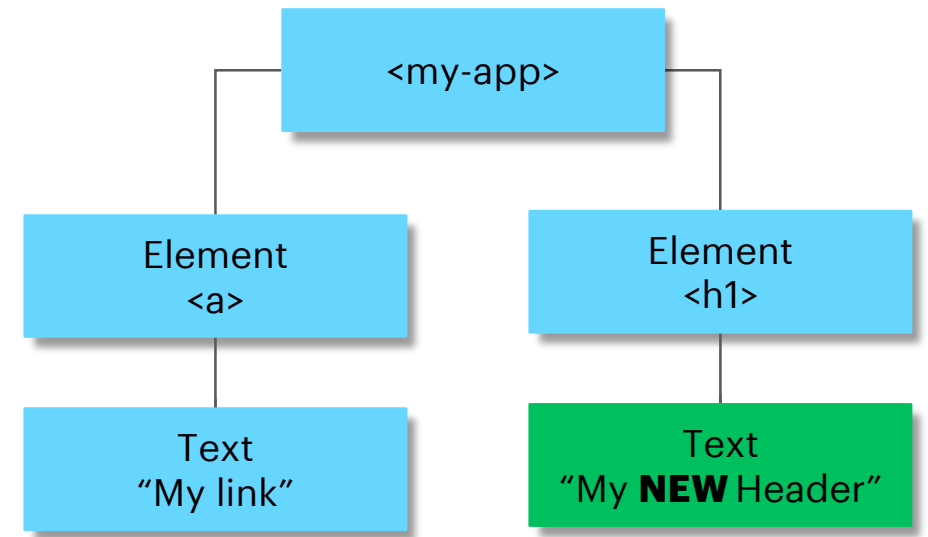


VIRTUAL DOM

Reconciliation - Virtual DOM compares the differences



DOM state

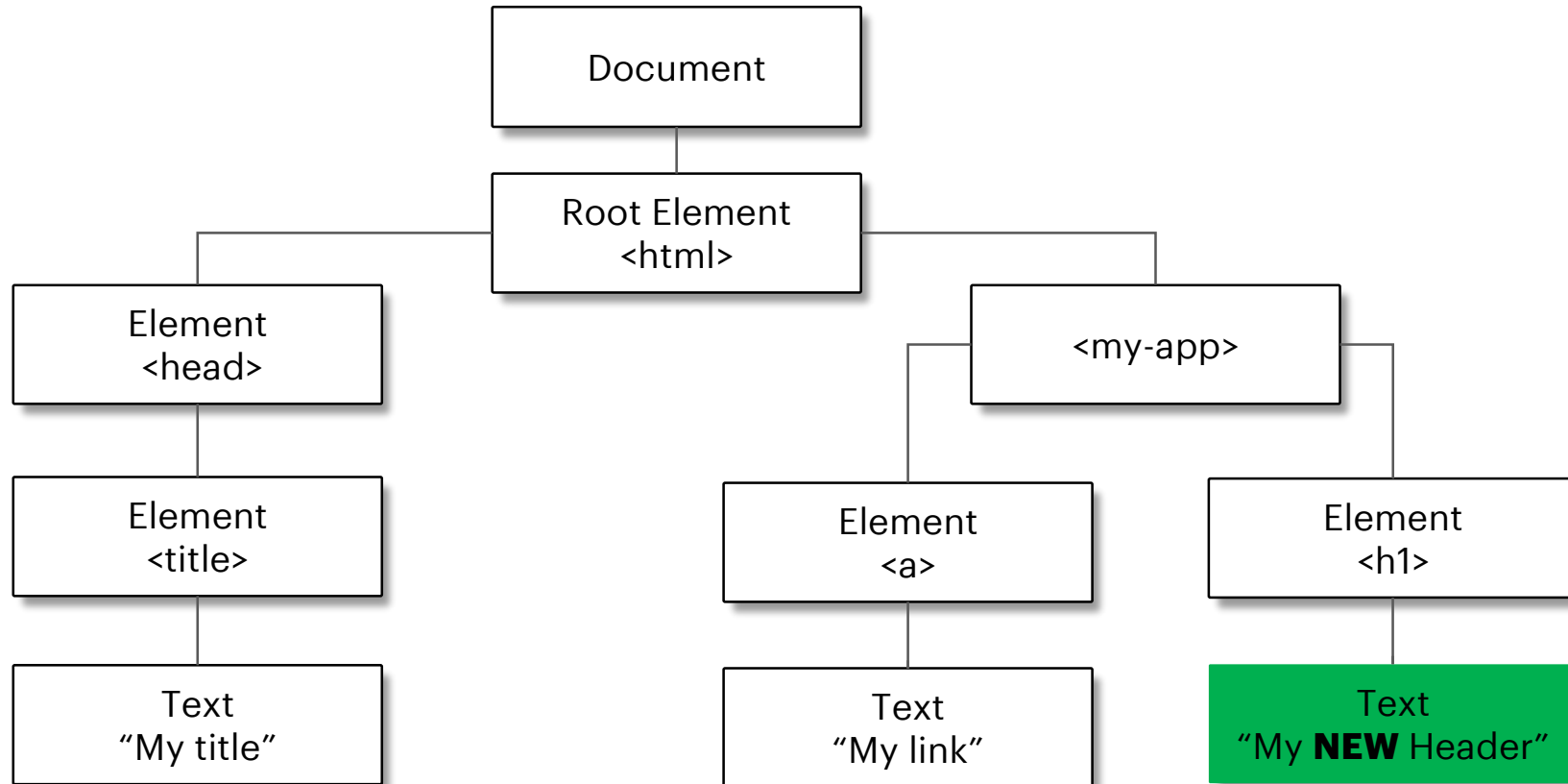


Virtual DOM state
New state

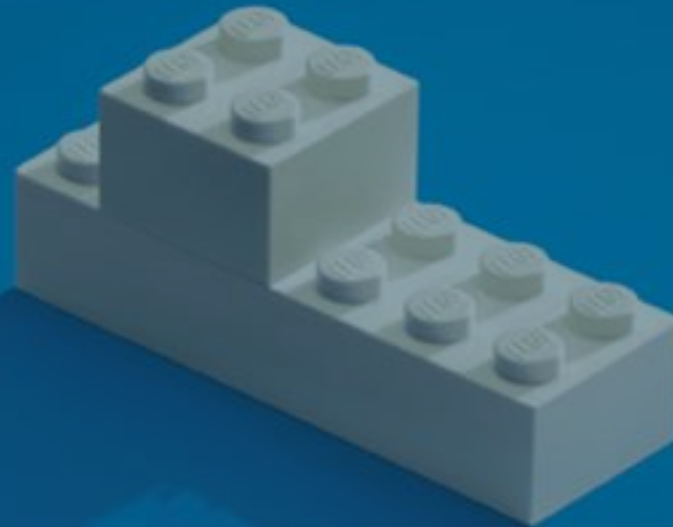
Does the DOM need to be updated? **YES**

VIRTUAL DOM

New DOM



WEB COMPONENTS



WEB COMPONENTS

1 file – Component.vue

Template

Style



“Class”

WEB COMPONENTS

1 file – Component.vue

“Class”

```
<script>
export default {
  name: "HelloWorld",
  props: {
    msg: String,
  },
  methods: {
    say(message) {
      console.log(message);
    },
  },
};
</script>
```

Template

```
<template>
  <div class="hello">
    <h1>{{msg}}</h1>
    <button @click="say('Hello')">Say hello</button>
  </div>
</template>
```

Style

```
<style scoped>
h1 {
  margin: 40px 0 0;
}
</style>
```

WEB COMPONENTS

1 file – Component.vue

_ Need attributes? `data()`

_ Need to call a function from the template? `computed : { }`

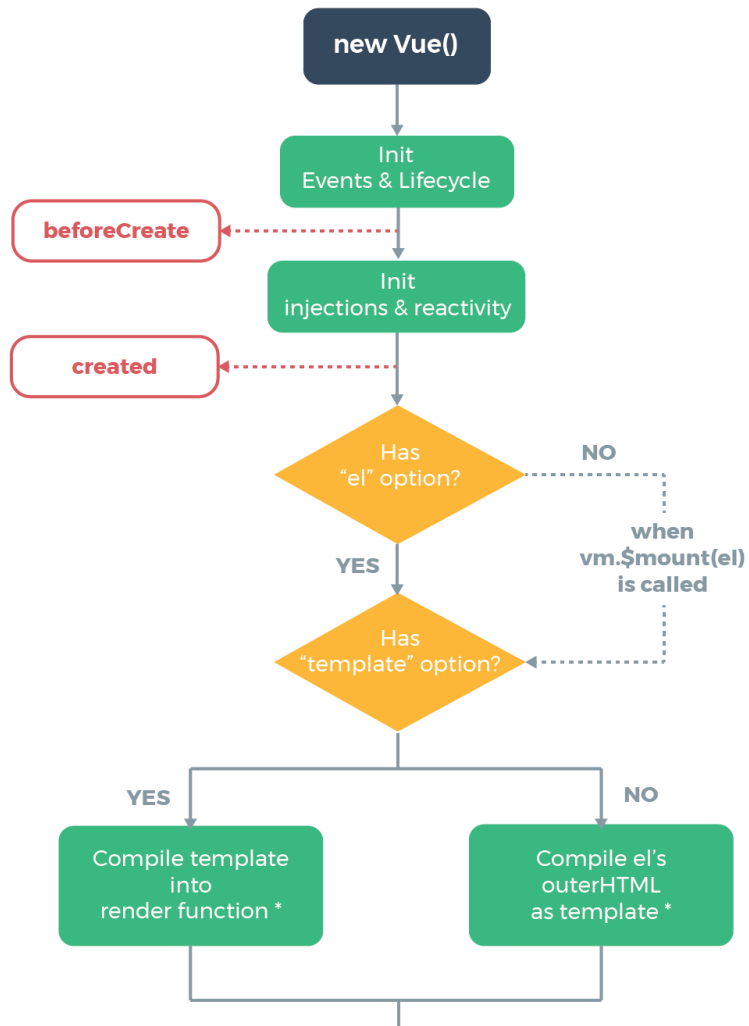
```
import Home from "../Home.vue";
export default {
  data() {
    return {
      notifications: [],
    };
  },
  components: {
    Home,
  },
  computed: {
    formatNotification() {
      // some code
    },
  },
  methods: {
    fetchData(progress) {
      // some code
    },
    loadMore() {
      this.fetchData(this);
    }
  }
}
```


WEB COMPONENTS LIFECYCLE

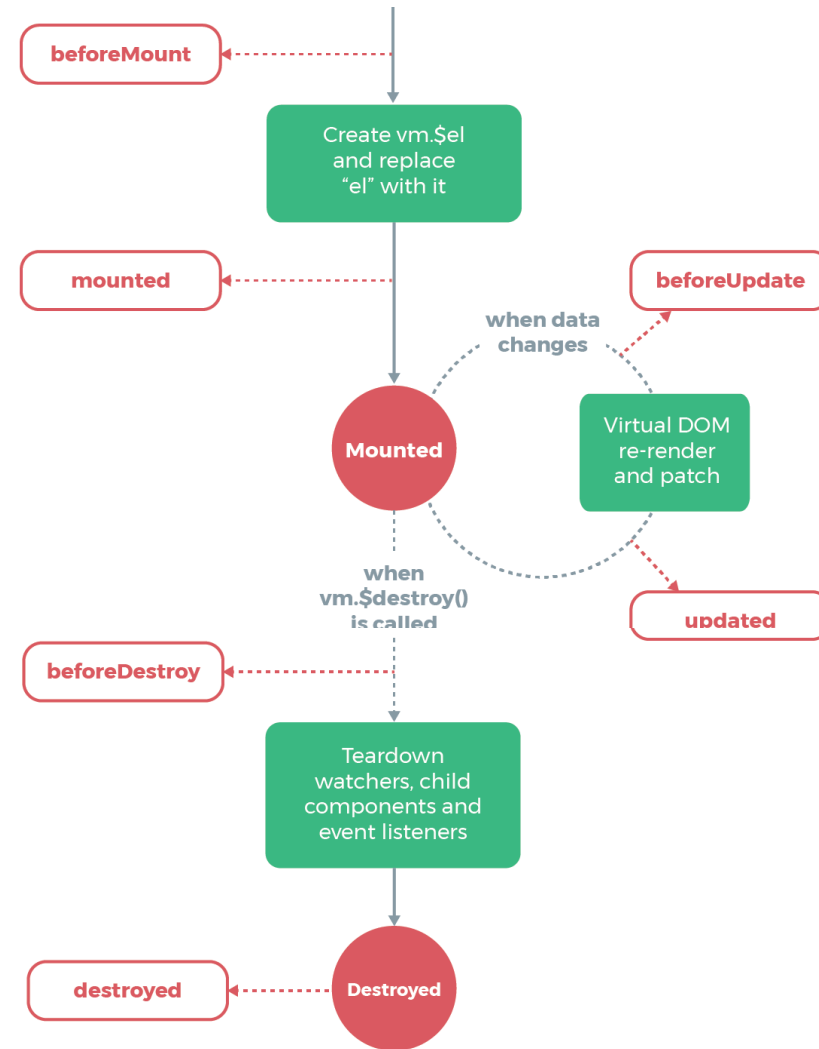


COMPONENT LIFECYCLE

Initialization



Mount & Run



Destroy

COMPONENT LIFECYCLE

_Example

```
import Home from './Home.vue';

export default {
  data() {
    return {
      notifications: [],
    };
  },
  components: {
    Home,
  },
  created() {
    console.log('created!');
  },
  beforeMount () {
    console.log('beforeMount!');
  },
  mounted () {
    console.log('mounted!');
  },
  methods: {
    fetchData(progress) {
      // some code
    },
    loadMore() {
      this.fetchData(this);
    }
  }
}
```

WEB COMPONENTS STYLE

```
346 .widget-area-sidebar {
347     font-size: 13px;
348 }
349
350
351 /* =Menu
352 -----
353
354 #access {
355     display: inline-block;
356     height: 69px;
357     float: right;
358     margin: 11px 28px 0px 0px;
359     max-width: 800px;
360 }
361
362 #access ul {
363     font-size: 13px;
364     padding-left: 0;
365     margin: 0 0 0 -0.8125em;
366     padding-left: 0;
367     z-index: 99999;
368     text-align: right;
369 }
370
371 #access li {
372     display: inline-block;
373     text-align: left;
```

WEB COMPONENTS STYLE

GLOBAL STYLE

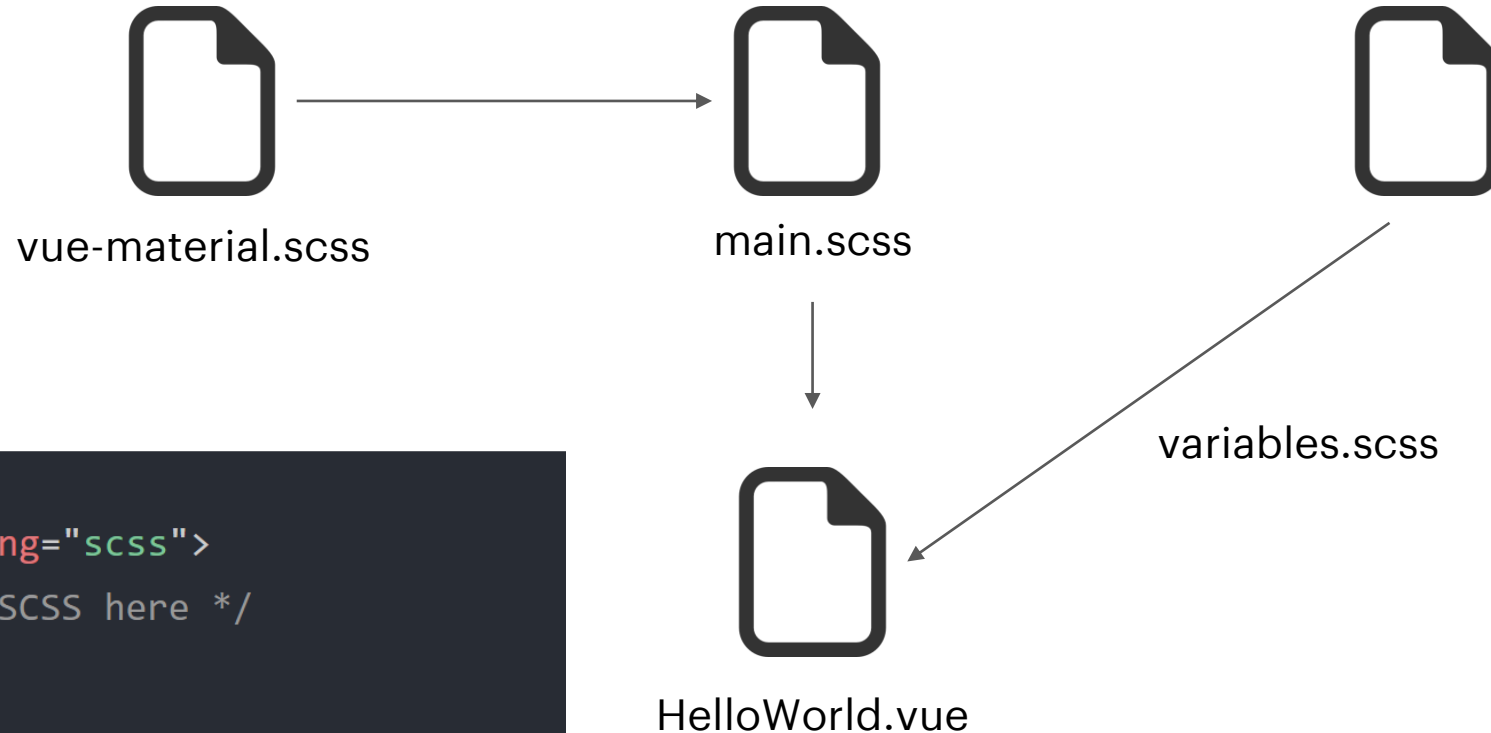
```
<style>
html, body {
  height: 100%;
}
#app {
  font-family: Avenir, Helvetica, Arial, sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  color: #2c3e50;
}
```

SCOPED STYLE

```
<!-- Add "scoped" attribute to limit CSS to this component only -->
<style scoped>
.message {
  display: flex;
}
.separator {
  border-top: 2px solid #f7f7f7;
  margin-bottom: 1em;
  margin-top: 1em;
}
```


WEB COMPONENTS STYLE

FRAMEWORK FOR STYLE



```
<style lang="scss">
/* write SCSS here */
</style>
```



SHARED BEHAVIORS

SHARED BEHAVIORS

Directives

_ **V:BIND**: class, style etc..

```
<!-- bind an attribute -->


<!-- dynamic attribute name -->
<button v-bind:[key]="value"></button>

<!-- shorthand -->


<!-- shorthand dynamic attribute name -->
<button :[key]="value"></button>
```

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

```
computed: {
  isActive() {
    return {
      active: this.isActive && !this.error
    };
  }
}
```

SHARED BEHAVIORS

Directives

_ **V:IF, V:FOR, V:ON** etc...

```
<div v-if="type === 'A'">
|   A
</div>
<div v-else-if="type === 'B'">
|   B
</div>
<div v-else-if="type === 'C'">
|   C
</div>
<div v-else>
|   Not A/B/C
</div>
```

```
<div v-for="item in items" :key="item.id">
|   {{ item.text }}
</div>
```

```
<!-- method handler -->
<button v-on:click="doThis"></button>

<!-- dynamic event -->
<button v-on:[event]="doThis"></button>

<!-- inline statement -->
<button v-on:click="doThat('hello', $event)"></button>

<!-- shorthand -->
<button @click="doThis"></button>
```




--- DATA FLOW

DATA FLOW

Communication **Parent** → **Child** component

_:prop

Parent

```
<div class="app">
  <HelloWorld :msg="'Welcome to Your Vue.js App'"/>
</div>
```

Child

```
export default {
  name: "HelloWorld",
  props: {
    msg: String,
  },
}
```

```
<div class="hello">
  <h1>{{msg}}</h1>
  <button @click="say('Hello')">Say hello</button>
</div>
```

DATA FLOW

Communication **Child -> Parent** component _this.\$emit

Child

```
emitCount() {  
  | this.$emit("count", 3);  
},
```

Parent

```
<HelloWorld v-on:count="updateCount"/>
```

```
methods: {  
  | updateCount(count) {  
    |   this.count = count;  
  }  
},
```


5X

K!

PRACTICE TIME!

TD – VUE.JS BASICS

Communication via Slack

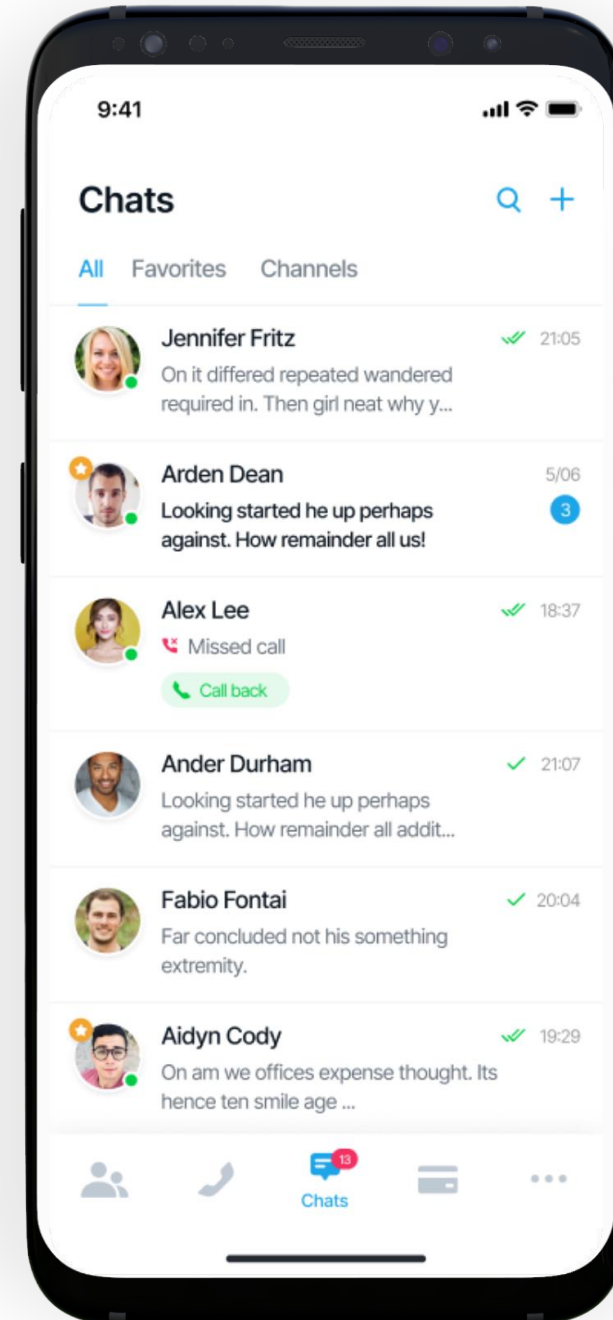


Canal **#si4-lang_doc**

Intervenants TDs :

_ **G1 & G2 : Remi Pourtier**

_ **G3 : Mathias Couste**





WEB TECHNOLOGY COMPARISON

Intervenants :
Mathias Cousté
Rémi Pourtier

 **accenture**technology