It’s Not About Web Components vs. React

Solve Different Problems

Thomas Fuchs 🌵
@thomasfuchs

Legendary Apollo project programmer Margaret Hamilton, next to a printout of the node_modules directory listing for her first Hello World react app

♡ 15.7K 9:54 AM - Mar 7, 2019

5,105 people are talking about this

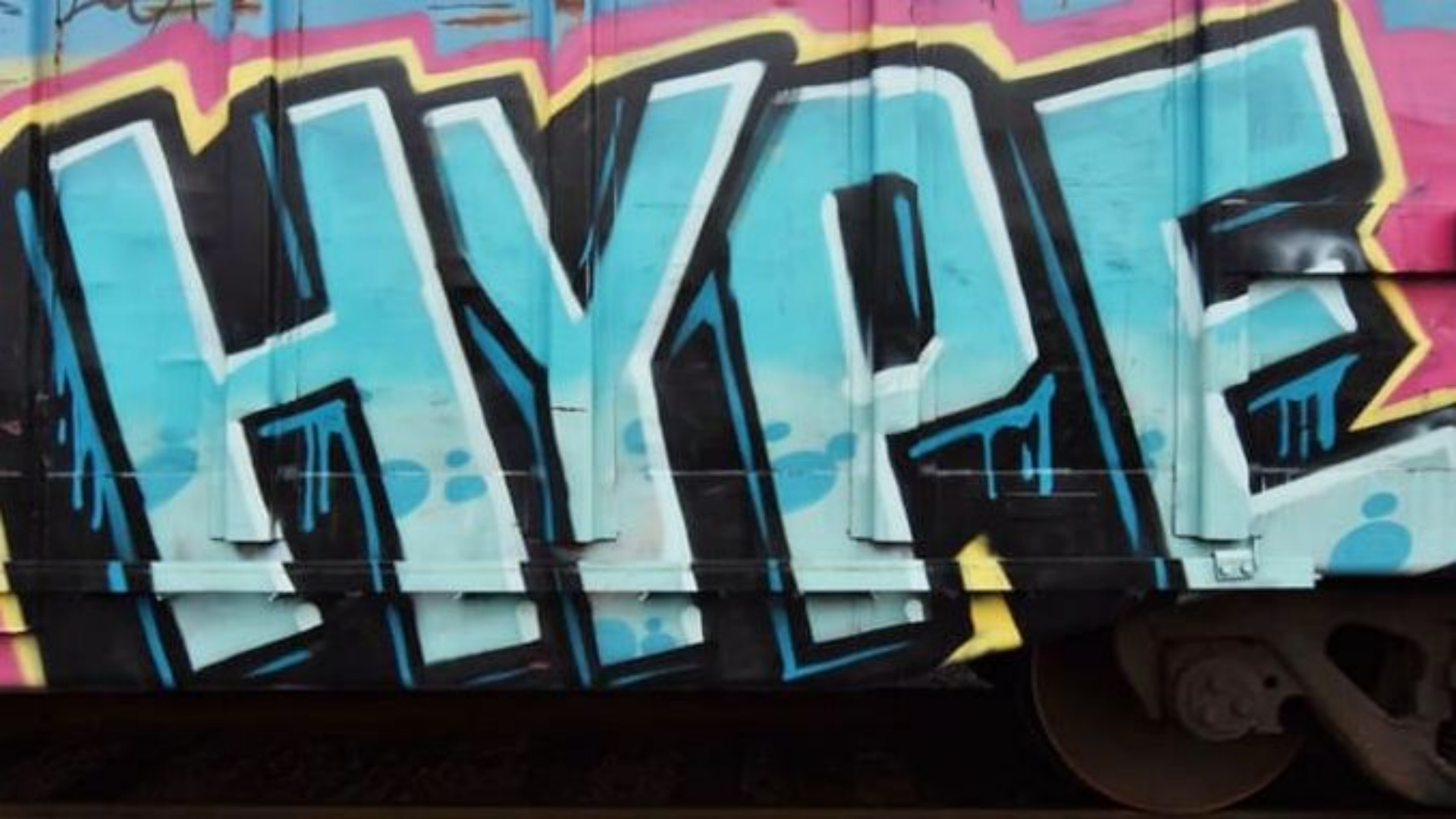
Web App

Web Components

Spec

It’s not ready

Living Document


HYPE

No community

The community is all of us

DOM is Bad

Browser API

```
class MyCustomElement extends HTMLElement {
 constructor() {
   super();
   const template = document.createElement('template');
   const shadowRoot = this.attachShadow({mode: 'open'});
   template.innerHTML = `<p>I'm a paragraph in a custom element!</p>`;
   shadowRoot.appendChild(template.content.cloneNode(true));
 }
}
customElements.define('my-custom-element', MyCustomElement)
```

```
document.body.appendChild(
    document.createElement('my-custom-element')
)
```

```
get observedAttributes() {
  return ['theme'];
}
get attributeChangedCallback(name, oldValue, newValue) {
  if (name === 'theme') {
    this.classList.add(newValue);
  }
}
```

Custom Elements API

Composable
Encapsulation
Interoperable
Accessible
Portable
Reusable
Lifetime

Use the platform

Let’s make the browser our platform

Let’s speak the same language

Custom Elements
Shadow DOM
ES Modules
HTML Templates

Web Components
createTreeWalker
Proxy
CustomEvent
BroadcastChannel
Web Animations

Decorators
HTML Modules

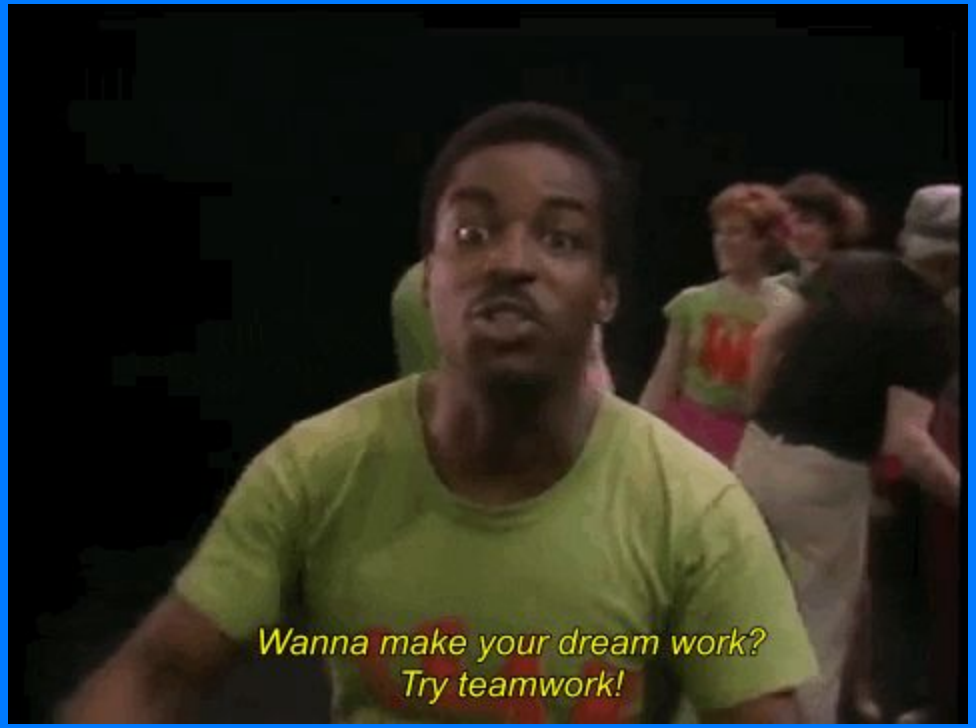
You Still Need A Framework

```
class MyCustomElement extends HTMLElement {
 constructor() {
   super();
   const template = document.createElement('template');
   const shadowRoot = this.attachShadow({mode: 'open'});
   template.innerHTML = `<p>I'm a paragraph in a custom element!</p>`;
   shadowRoot.appendChild(template.content.cloneNode(true));
 }
}
customElements.define('my-custom-element', MyCustomElement)
```

```
@Component({
  template:`<p>I'm a paragraph in a custom element!</p>`,
  style:`:host { background: red }`,
})
class MyComponent extends CustomElement {
  constructor() {}
}
customElements.define('my-custom-element', MyComponent)
```

~1Kb ‘Hello World’

I’m not selling anything


Wanna make your dream work?
Try teamwork!

This has been a friendly message

@iplayitofflegit

```
import { Component, CustomElement } from '@readymade/core';

@Component({
  template:`<p>I'm a paragraph in a custom element!</p>`,
  style:`:host { background: red }`,
})
class MyComponent extends CustomElement {
  constructor() {}
}
```

```
class ButtonState {
 public model: string = 'Click';
}
class MyButtonComponent extends CustomElement {
  constructor() {}
  @State()
  public getState() {
    return new ButtonState();
  }
}
```

Method Decorator

```
@Component({
 template: html`
  <span>{{model}}</span>
  `
})
class MyButtonComponent extends CustomElement {
...
```

One Way Data Binding

```
@Emitter('bang', { bubbles: true, composed: true })
@Listen('click')
public onClick(event: MouseEvent) {
 this.emitter.broadcast('bang');
}
@Listen('keyup')
public onPress(event: KeyboardEvent) {
  if (event.key === 'Enter') {
    this.emitter.broadcast('bang');
  }
}
```

Event Binding

```
@Emitter('bang', { bubbles: true, composed: true })
@Listen('click')
public onClick(event: MouseEvent) {
 this.emitter.broadcast('bang');
}
@Listen('keyup')
public onPress(event: KeyboardEvent) {
  if (event.key === 'Enter') {
    this.emitter.broadcast('bang');
  }
}
```

Event Binding

```
get observedAttributes() {
  return ['theme'];
}
get attributeChangedCallback(name, oldValue, newValue) {
  if (name === 'theme') {
    this.classList.add(newValue);
  }
}
```

Custom Elements API