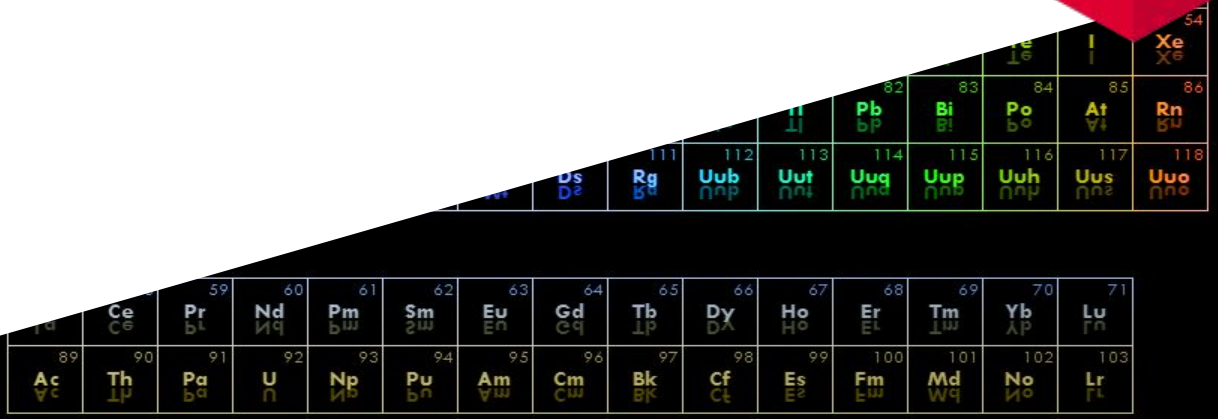
Angular Elements

M Kusold



Warning

"We're targeting v6 for an initial, experimental release of elements."

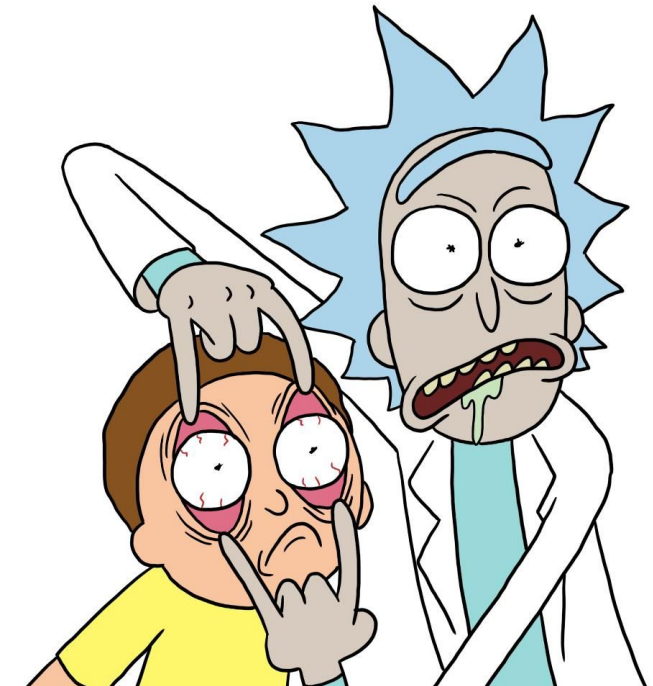
Rob Wormald (Google Dev Advocate)

The **experimental** release isn't even out yet.
This is nowhere near production ready.

Build Up

Announced as a part of “Angular Labs”
By Rob Wormald

What is it?



Angular Elements are essentially Angular Components packaged as Custom Elements

Or in other words:
Angular Components wrapped by a Web Component so that it can be **used outside of Angular**

Wait, what's a web component again?

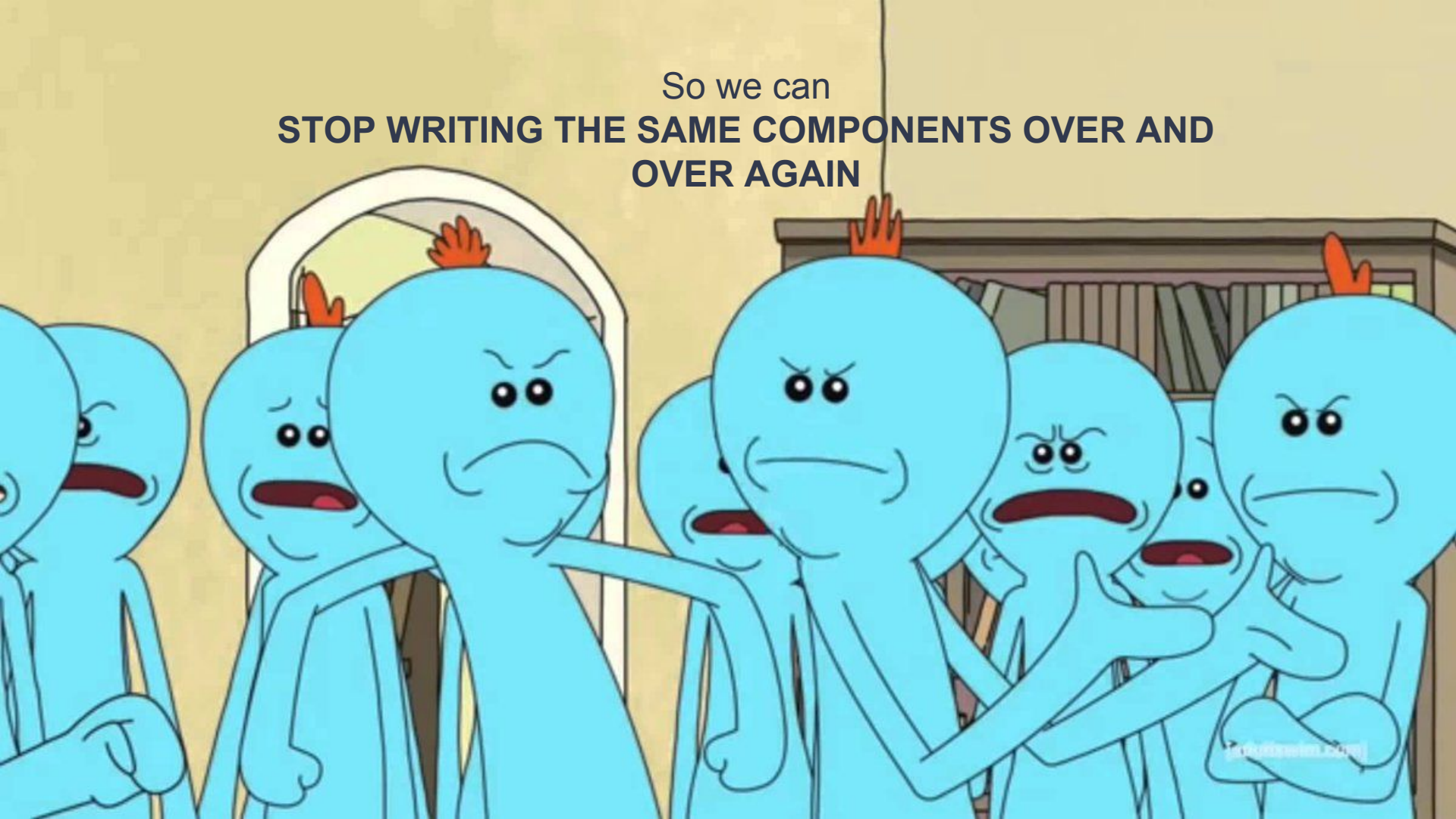
WEB COMPONENTS

TEMPLATES	SHADOW DOM	HTML IMPORTS	CUSTOM ELEMENTS

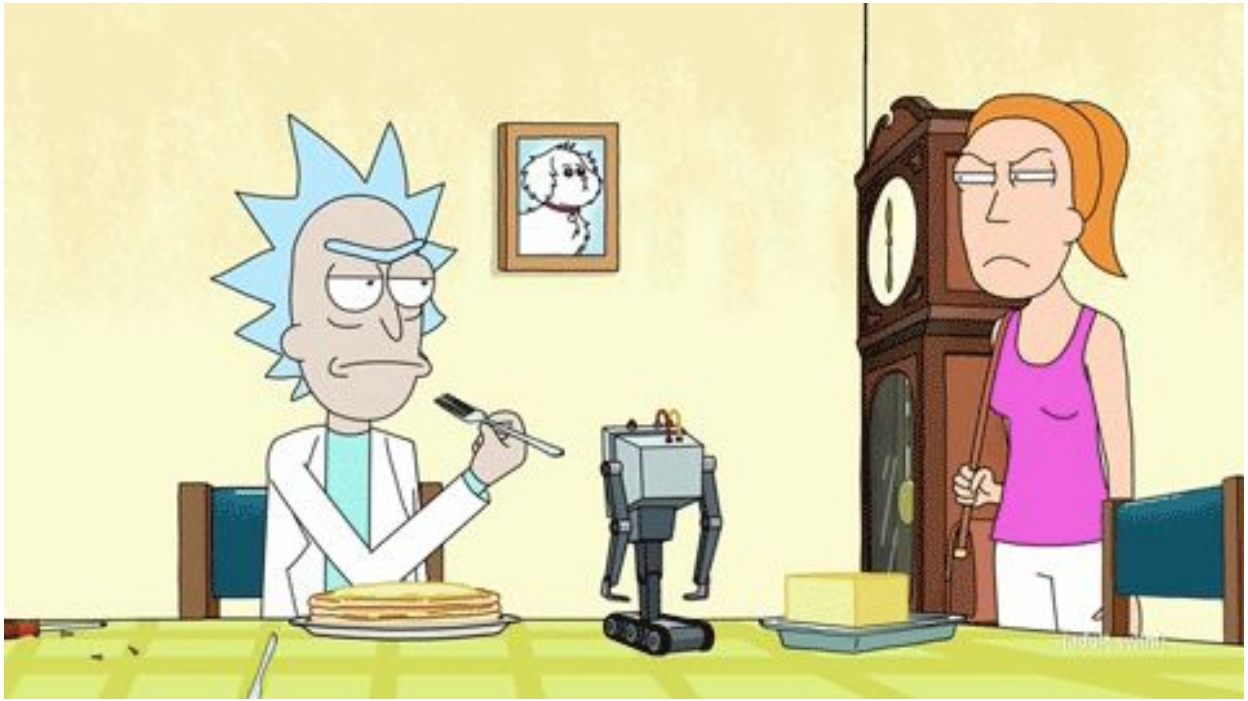
`<template id="">`
`</template>`
	`div`
	`  #document-fragment`
	`  span`
		`<link rel="import"`
		`  href="part.html">`
			`<my-elem>`
			`</my-elem>`

So we can

STOP WRITING THE SAME COMPONENTS OVER AND OVER AGAIN



What things would be good for this?



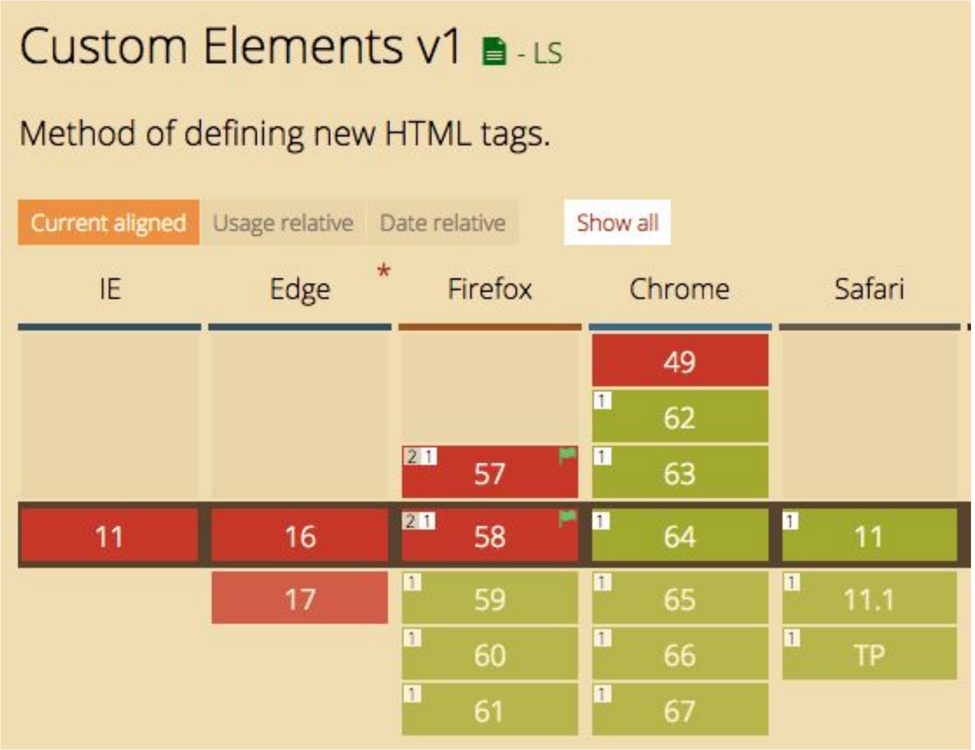
Can’t we already do this?

WEB COMPONENTS

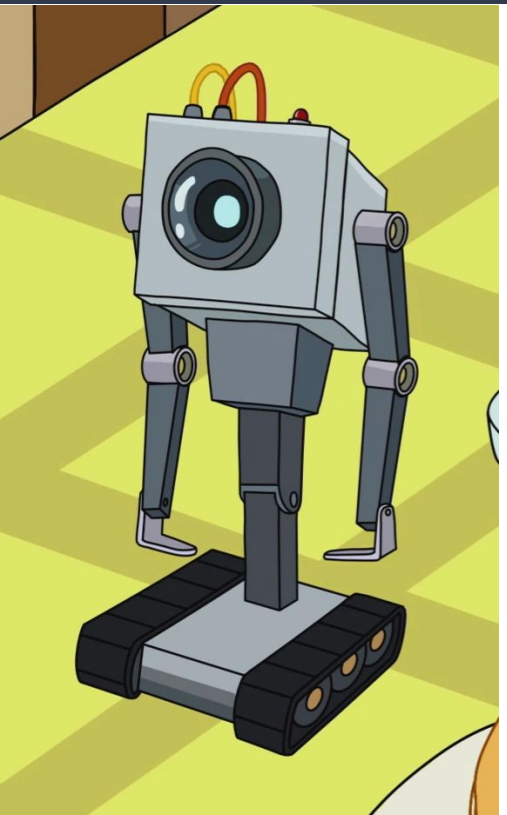
TEMPLATES	SHADOW DOM	HTML IMPORTS	CUSTOM ELEMENTS

`<template id=“”>`
`</template>`
	`div`
	`  #document-fragment`
	`  span`
		`<link rel=“import”`
		`  href=“part.html”>`
			`<my-elem>`
			`</my-elem>`

Don’t We Already Have That?

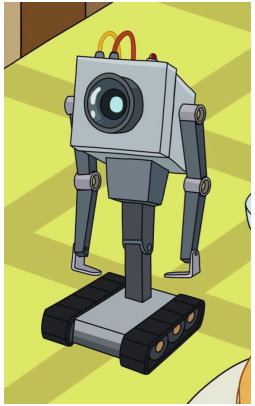


What is my purpose?



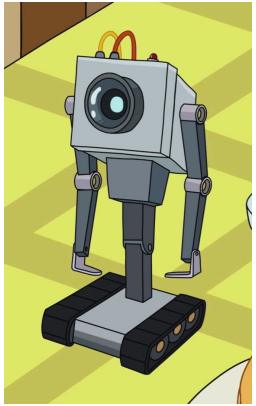
You Pass the Butter

Polymer



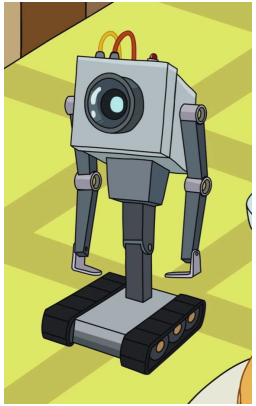
```
// boilerplate,                                re, go to my github)
class PolymerP                                  t {
     static ge                                  butter'; }
     static ge
       return
         amount: { type
         goingRight: { ty                  n, value:true }
       };
     }
     passButter() {...}
   }
```

Writing an Angular Element (as of 2.27.19)



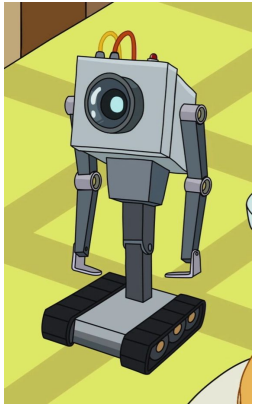
```
import {
...
 ViewEncapsulation
} from '@angular/core';
@Component({
 selector: 'butter-passer',
 template: ` div class="butter-passer">
   <button class="command btn" (click)="passButter()">{{text}}</button>
   <img class="butter" #butter [style.margin-left]="ml + '%'" src="./images/butter-bot.png"/>
 </div>`,
 styles: [`...`],
 encapsulation: ViewEncapsulation.Native
})
```

1. Write component



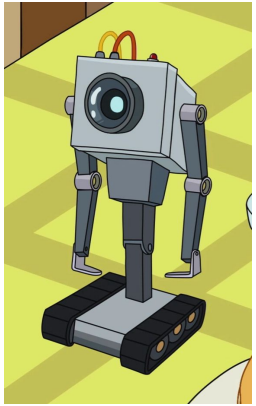
```
export class ButterComponent {
 @Input() text;
 @Output() butterPassed = new EventEmitter<string>();
  goingRight = true;
  ml = 0;
  @ViewChild('butter') butterRef;

 passButter() {
  // ... lots of buttery goodness
   this.butterPassed.emit();
 }
}
```

2. Write module



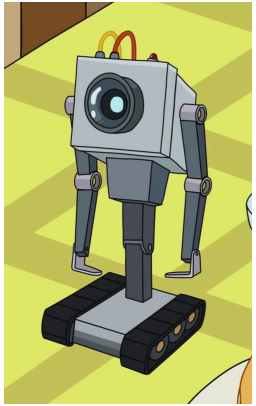
```
import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';
import { ButterComponent } from './butter.component';

@NgModule({
 imports: [BrowserModule],
 declarations: [ButterComponent],
 entryComponents: [ButterComponent]
})
export class ButterModule {
 ngDoBootstrap() {}
}
```

3. Register custom element



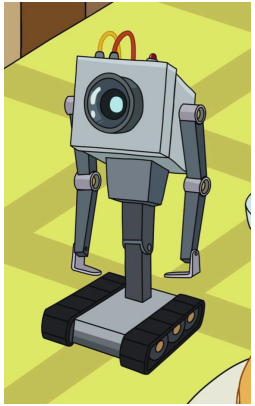
```
import 'zone.js/dist/zone';

import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';

import { registerAsCustomElements } from '@angular/elements';

import { ButterModule } from './butter.module';
import { ButterComponent } from './butter.component';

registerAsCustomElements([ButterComponent], () =>
 platformBrowserDynamic().bootstrapModule(ButterModule)
);
```

4. Use it



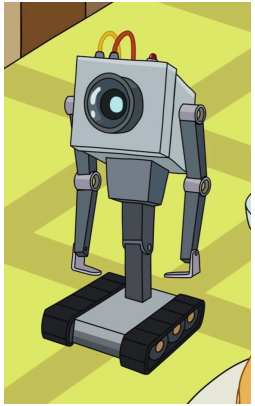
```
<!-- Angular Element -->
<script src="./angular-element.bundle.js"></script>

...

<butter-passer text="Pass the Butter" imgSrc="./images/butter-bot.png"></butter-passer>
```

Using it in Vue



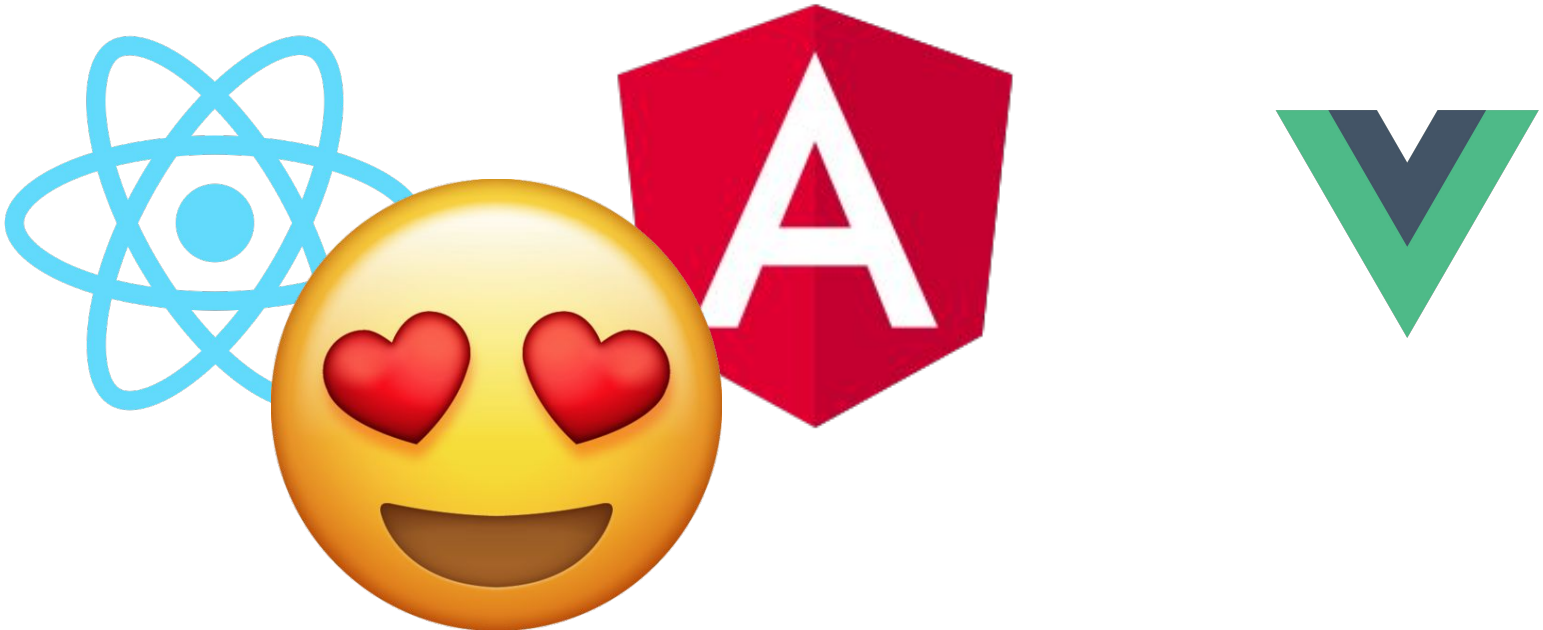
```
<!-- index.html -->
<body>
 <div id="vue-app"></div>
 <!-- built files will be auto injected -->
 <script src="angular-element.bundle.js"></script>
</body>
```

Using it in Vue



```
// App.vue
<template>
 <div>
   <h1>VUE!</h1>
   <butter-passer v-bind:text="text"></butter-passer>
 </div>
</template>
<script>
export default {
 name: 'vue-app',
 data() { return { text: "Pass the Butter"}; },
 methods: { }
};
</script>
```

React, Angular, & Anything else



Looking Into the *Future*

“We're targeting v6 for an initial, experimental release of elements. We'll be validating it with Angular.io, and aim to solve that use case first.”

“The current codebase is on the labs/elements branch - https://github.com/angular/angular/tree/labs/elements/packages/elements/ but we expect to merge it to master shortly, pending initial docs.”

“We want to be open about what we're doing, but set the expectation this is not production ready yet. We'll have some basic docs for the initial release, and full documentation and integration when it graduates from Labs status.”

Questions?

Github Page:
https://michellekusold.github.io/angular-elements-presentation/

Code:
https://github.com/michellekusold/angular-elements-presentation

Personal Site:
http://www.michellekusold.com/

References

Rom Wormald's Latest Update:
https://github.com/angular/angular/issues/20891
Rom Wormald's Initial Presentation:
https://www.youtube.com/watch?v=ljsOPm4MMEo&feature=youtu.be
Wormald's Proof of Concept:
https://github.com/robwormald/angular-elements

Great Demo:
https://github.com/vogloblinsky/angular-elements-demo
Andrei Antal:
https://www.youtube.com/watch?v=-pS8M_RBf84
Andrei Antal's Demo:
https://github.com/andrei-antal/ng-europe-demo-angular-elements