Building reusable Web Components using Angular Elements

Rabi Kiran (Ravi)
Associate Architect
InnoMinds

- Blog: http://sravi-kiran.blogspot.com
- Author: DotNetCurry, Sitepoint, Auth0
- Organizer of ngHyderabad
- Awards:
  Microsoft MVP (Developer Tech),
  DZone MVB
- Twitter: @sravi_kiran

Agenda

- Need of Reusable components
- Web Components
- Angular Elements and Demos

Thoughts from Angular Team

Angular is ideal for building complete applications
Rob Wormald, Angular Team

But it becomes challenging for scenarios that don't fit in Single Page Application Model

The need?

Can you use an Angular component in...

- In a React or Vue Application
- In a server rendered page like an ASP.NET page
- Integrate as a component in a CMS site

Do Web Standards answer this?

Web Component

Web Component Spec

- Template

```
<template>
  <h1>Hello World!</h1>
</template>
```

- Shadow DOM

```
<my-app>
  <#shadow-root>
    <style>.title { font-size: 14px; }
    </style>
    <h1 class=”title”></h1>
  </shadow-root>
</my-app>
```

Web Component Spec

- Custom Elements

```
customElements.define('my-navbar',
HelloWorld);
```

```
<body>
  <my-navbar></my-navbar>
  <div>
    <my-sidebar></my-sidebar>
    <my-main-content></my-main-content>
  </div>
  <my-footer></my-footer>
</body>
```

What is a Custom Element?

- A class inherited from HTMLElement

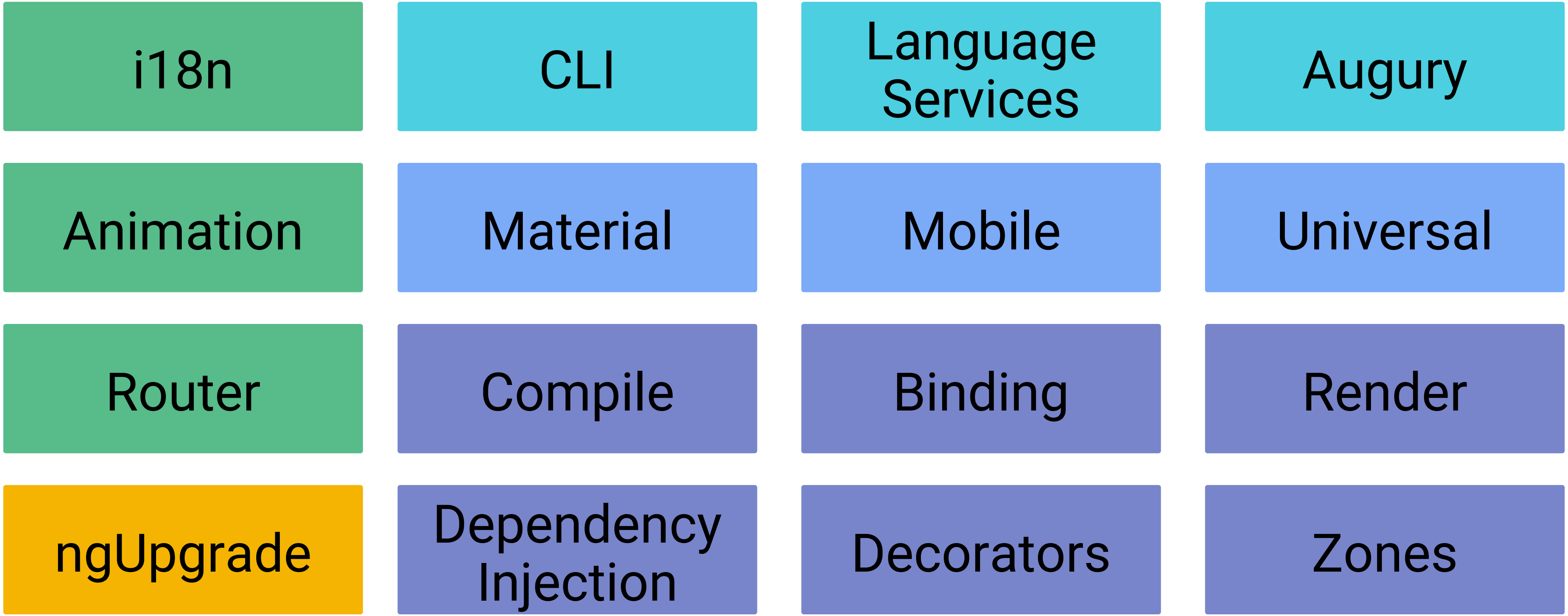
```
class MyComponent extends HTMLElement {
  static observedAttributes = ['my-comp'];

  attributeChangedCallback(oldV, newV, key){
  //update the DOM somehow
  }
}
```

- Like other HTML elements, it has attributes and events

Then why not embrace it and why are we talking about Angular now?

Angular’s Platform



i18n	CLI	Language Services	Augury
Animation	Material	Mobile	Universal
Router	Compile	Binding	Render
ngUpgrade	Dependency Injection	Decorators	Zones

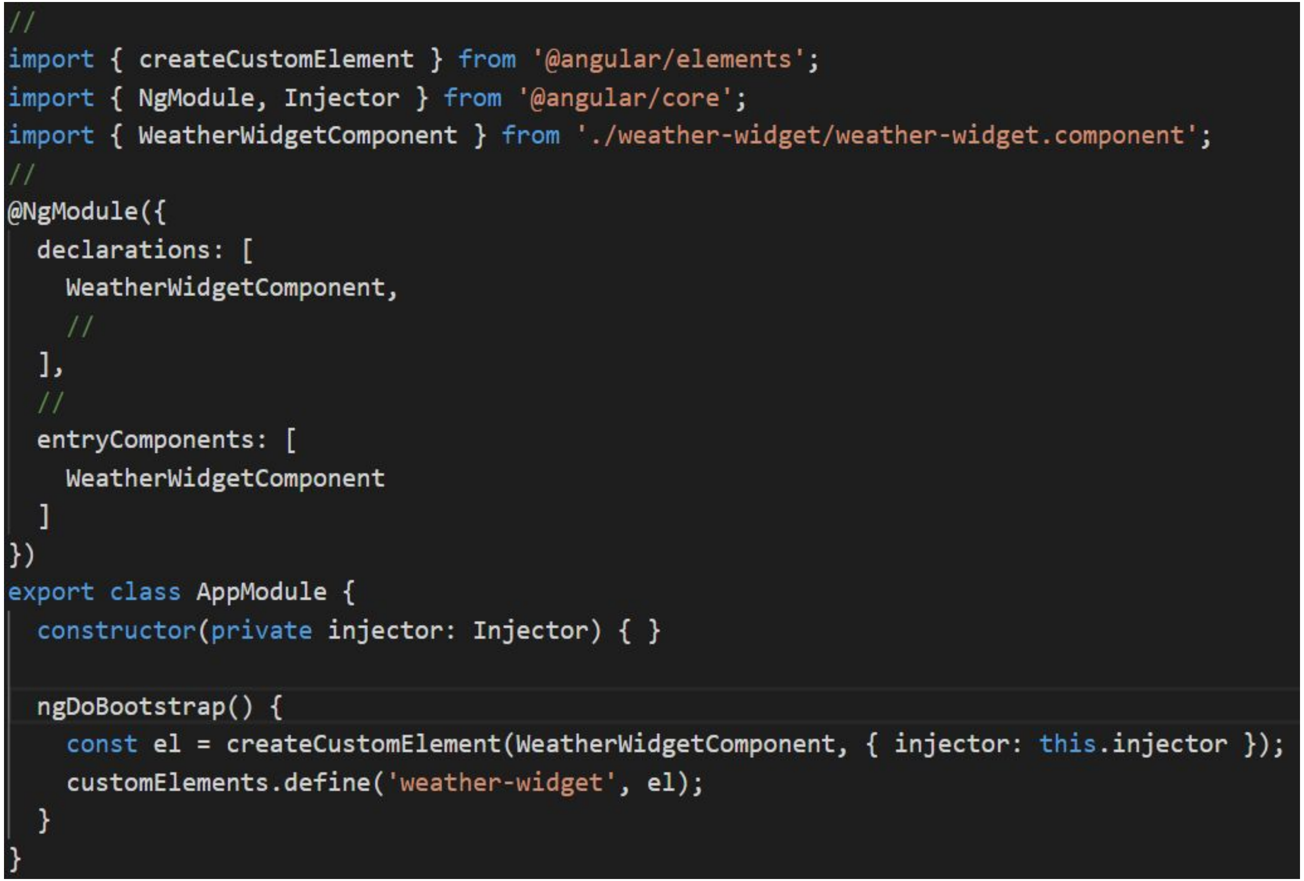
Building Angular Elements using Angular CLI

Setup

- Generate a project using Angular CLI
- *ng new mycomponentsapp*

- Add elements package
- *npm install @angular/elements*

Registering component



```
//
import { createCustomElement } from '@angular/elements';
import { NgModule, Injector } from '@angular/core';
import { WeatherWidgetComponent } from './weather-widget/weather-widget.component';
//
@NgModule({
  declarations: [
    WeatherWidgetComponent,
    //
  ],
  //
  entryComponents: [
    WeatherWidgetComponent
  ]
})
export class AppModule {
  constructor(private injector: Injector) { }

  ngDoBootstrap() {
    const el = createCustomElement(WeatherWidgetComponent, { injector: this.injector });
    customElements.define('weather-widget', el);
  }
}
```

Demos

https://github.com/sravikiran/angular-elements-demos

How will it benefit Angular?

- Easier to create dynamic components
- To build static sites like Angular documentation
- Upgradation from AngularJS

Further Learning

- https://angular.io/guide/elements
- https://blog.angular.io/the-angular-team-at-angularmix-2d56fd7fde65
- https://github.com/angular/angular/issues/21706
- http://slides.com/andreiantal/ng-europe_angular-elements
- https://medium.com/@ezekizibzibadze/angular-and-web-components-a-k-a-angular-elements-e009b057cd75
- https://blog.angularindepth.com/angular-elements-how-does-this-magic-work-under-the-hood-3684a0b2be95

“ *Questions?* ”

Thank-you