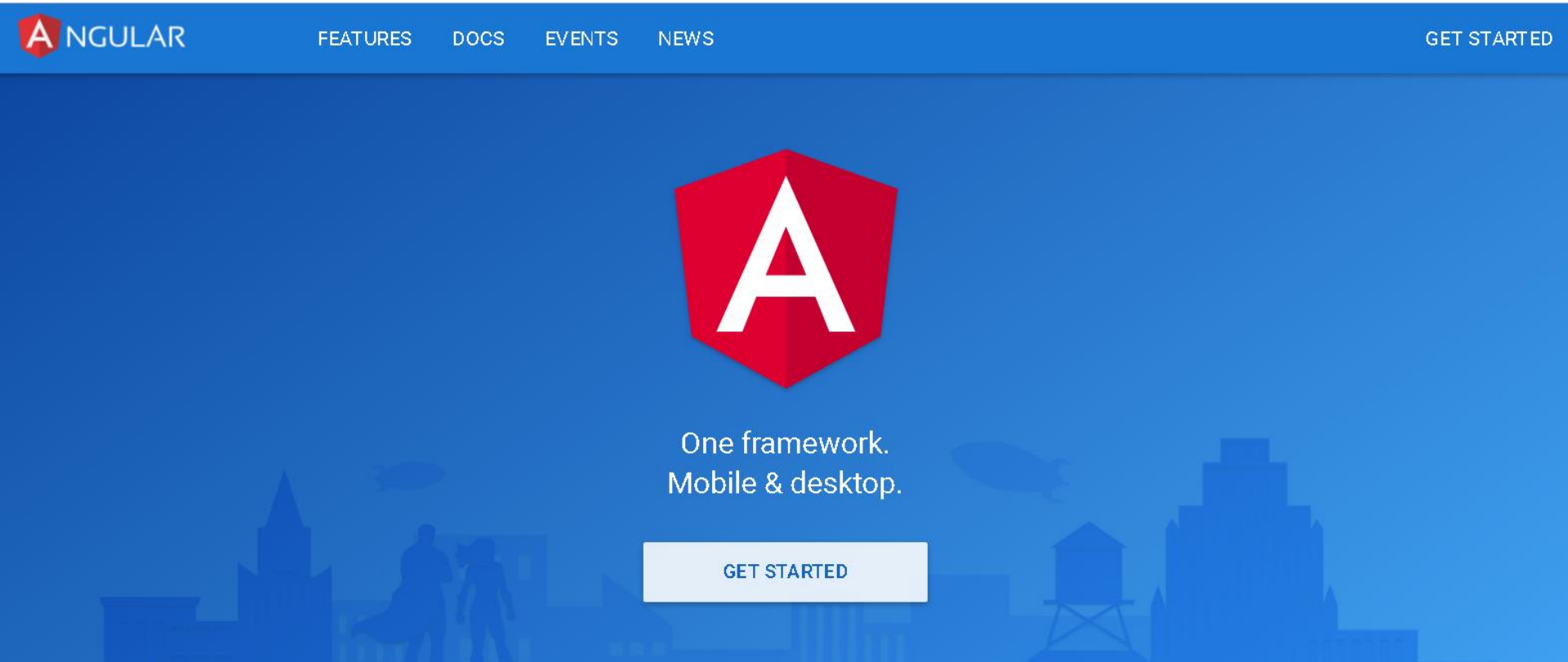
Angular 2

https://angular.io



Angular 2

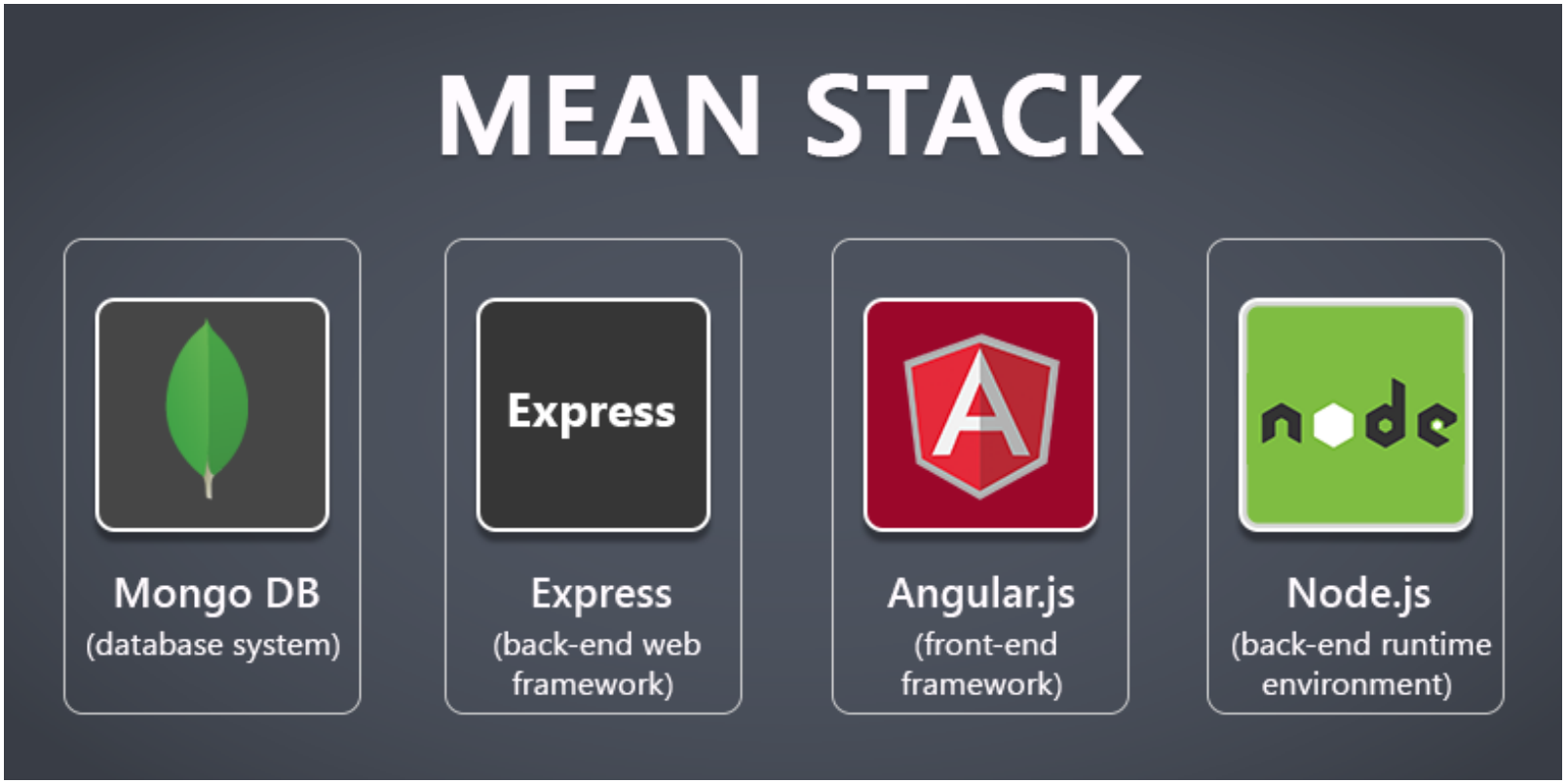
Agenda

Introduction & Big Picture
Difference between AngularJS
Architecture
Internals
Using Angular 2 (ng-cli)
Bootstrap process

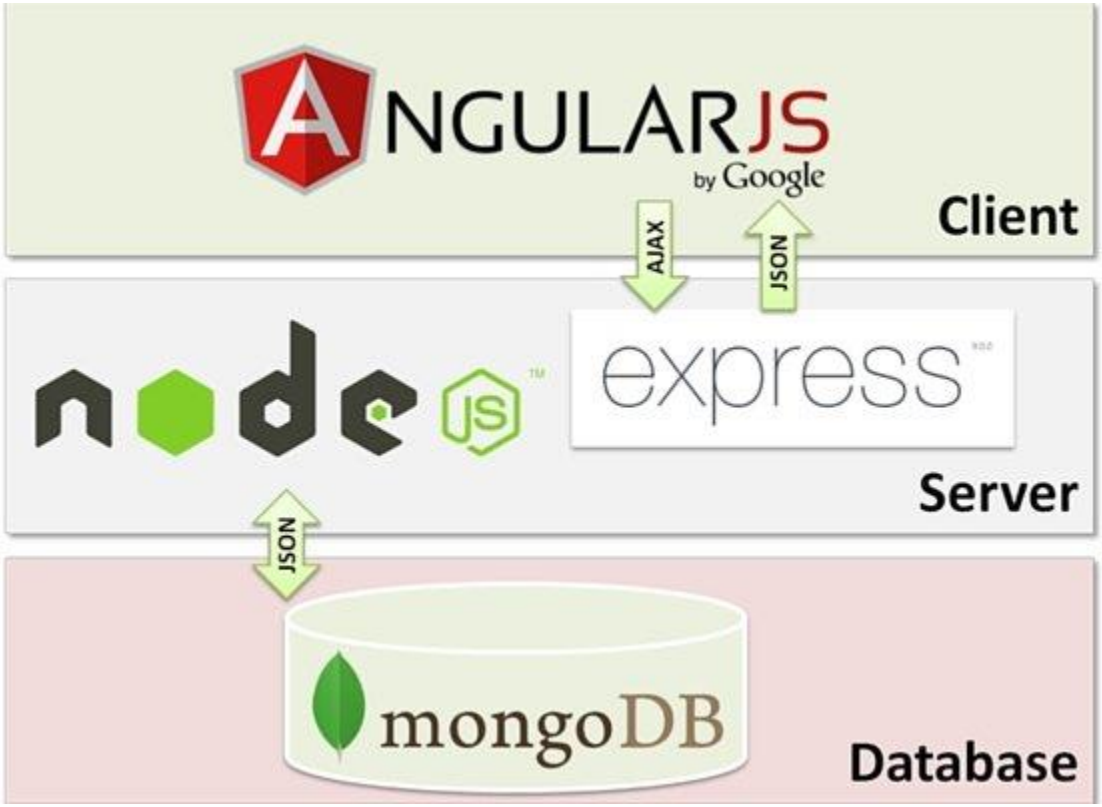
Introduction

Front-end technology
Modern web practices
JS SPA framework

MEAN STACK


MEAN STACK
Express
Mongo DB
(database system)
Express
(back-end web framework)
Angular.js
(front-end framework)
Node.js
(back-end runtime environment)

MEAN STACK


ANGULARJS
by Google
Client
AJAX
JSON
express
Server
JSON
mongoDB
Database

AngularJS

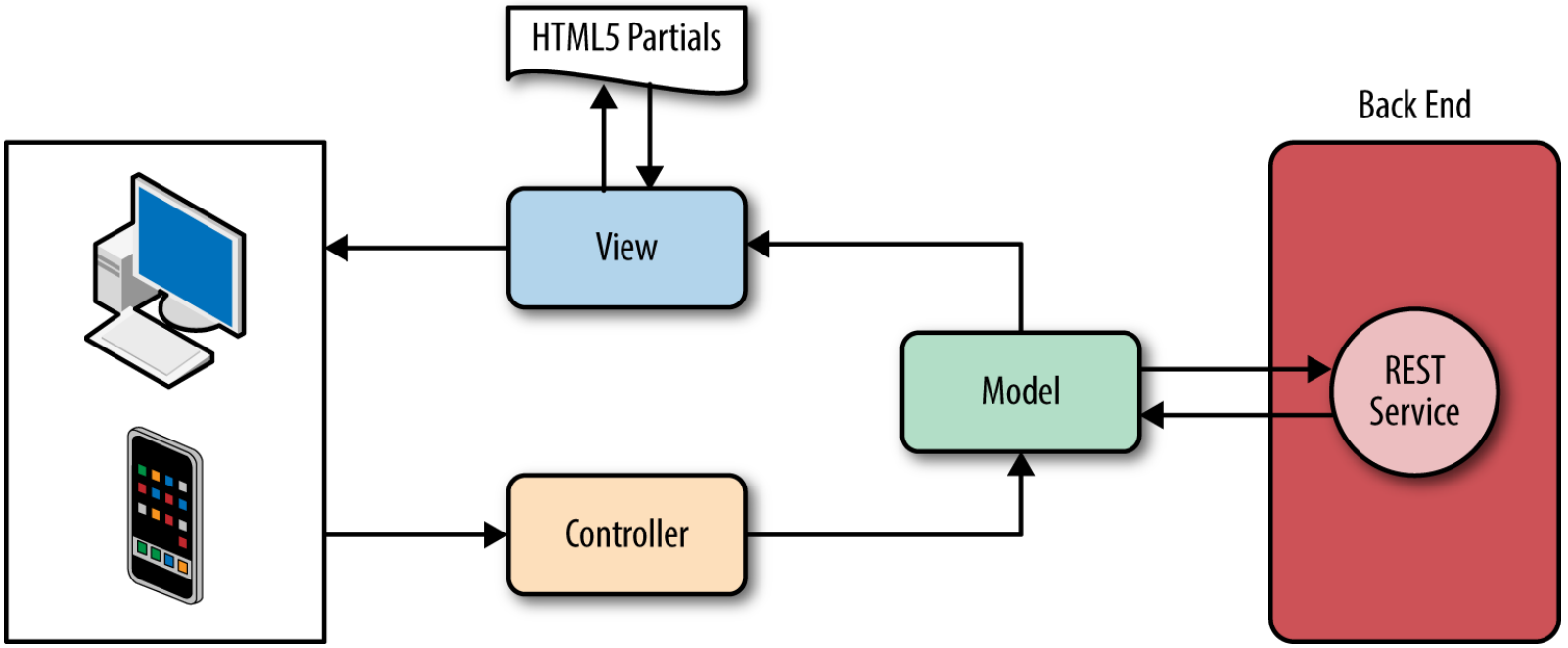
JS framework
SPA
By Google
Open Source
Many applications
Directives, Controllers,
Services, Routing,
$scope,
ng-if, ng-for

Angular 2

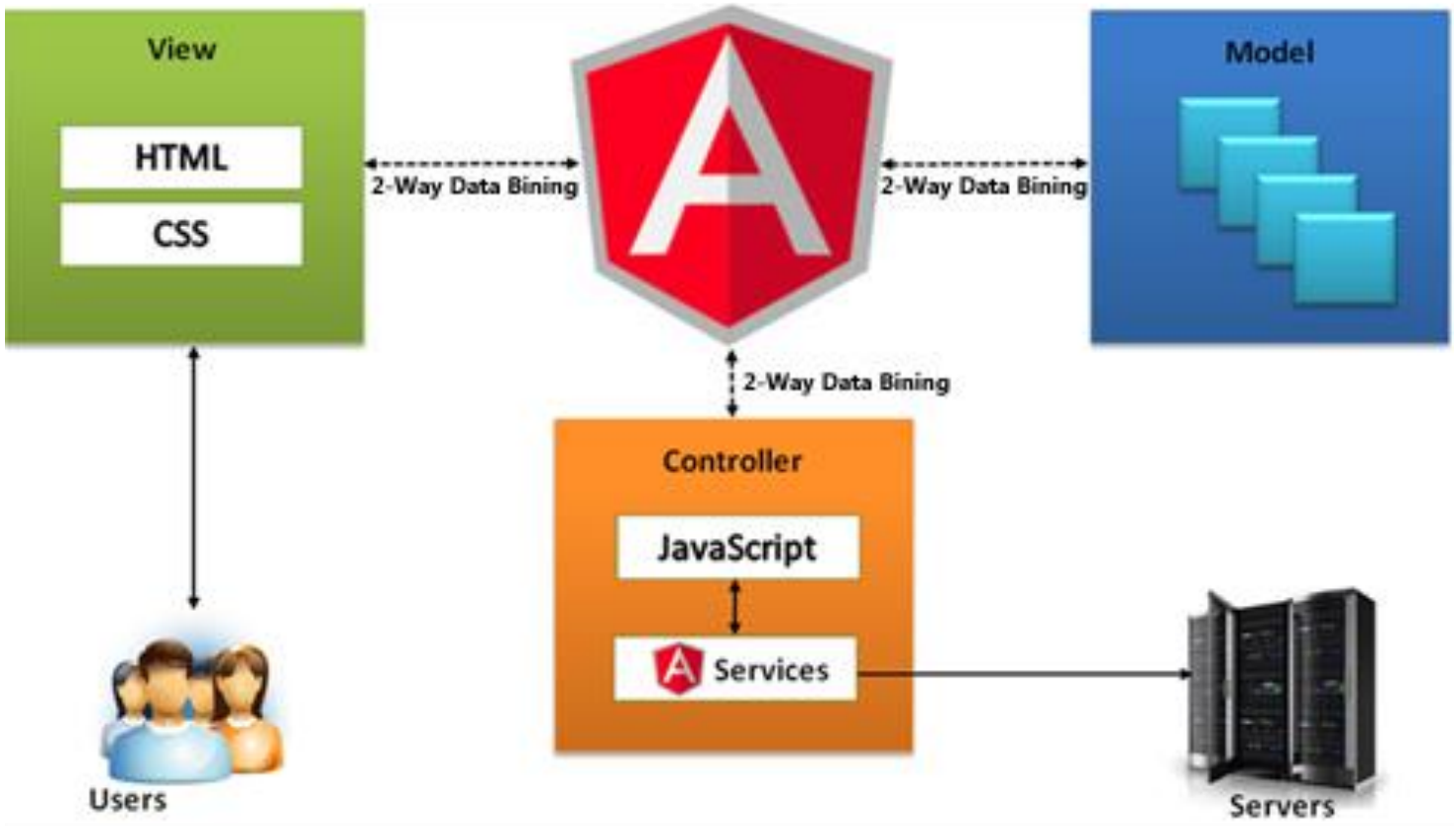
JS framework
SPA
Fast / Fluid Experience
Best Practices
From Google
Great Community
Many resources
Open Source

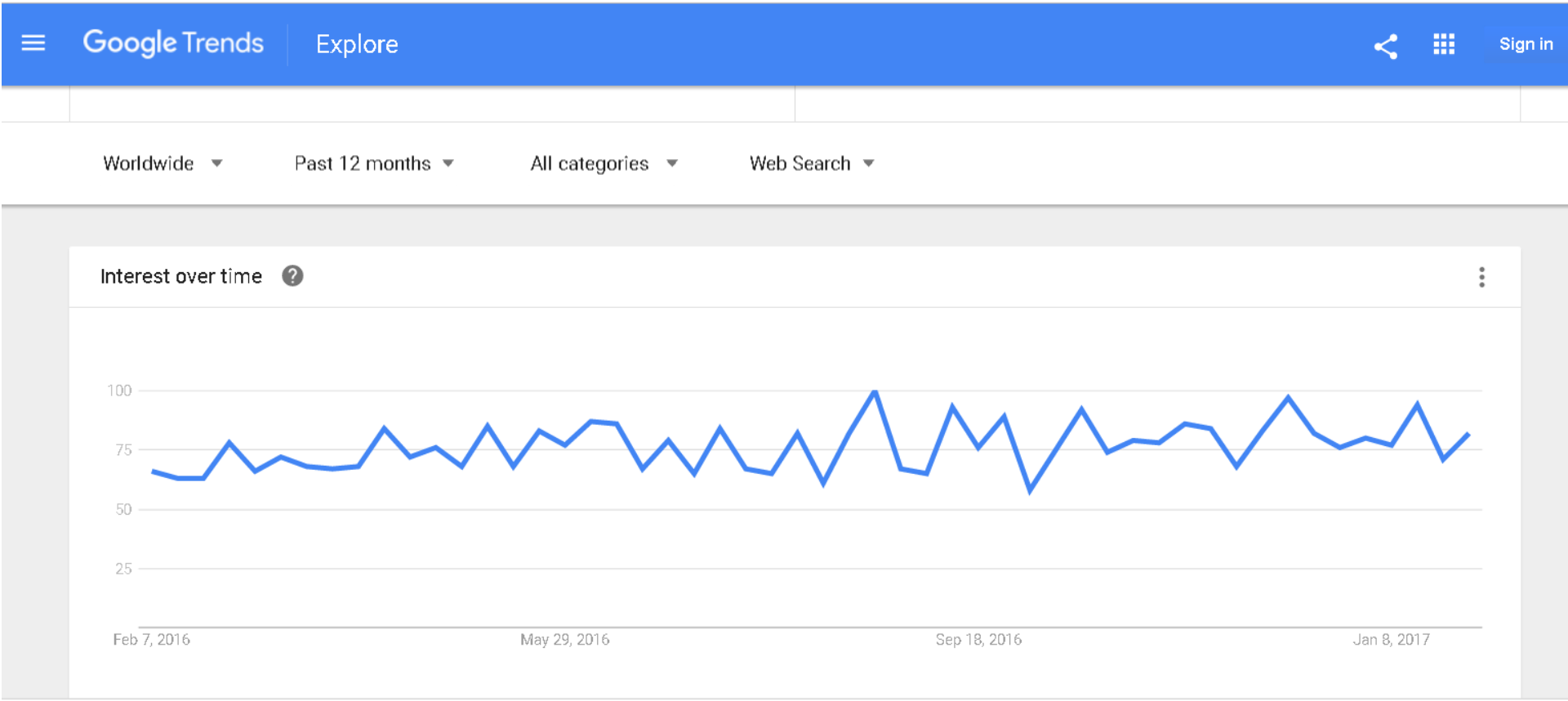
Architecture

MVC Pattern



MVC Implementation

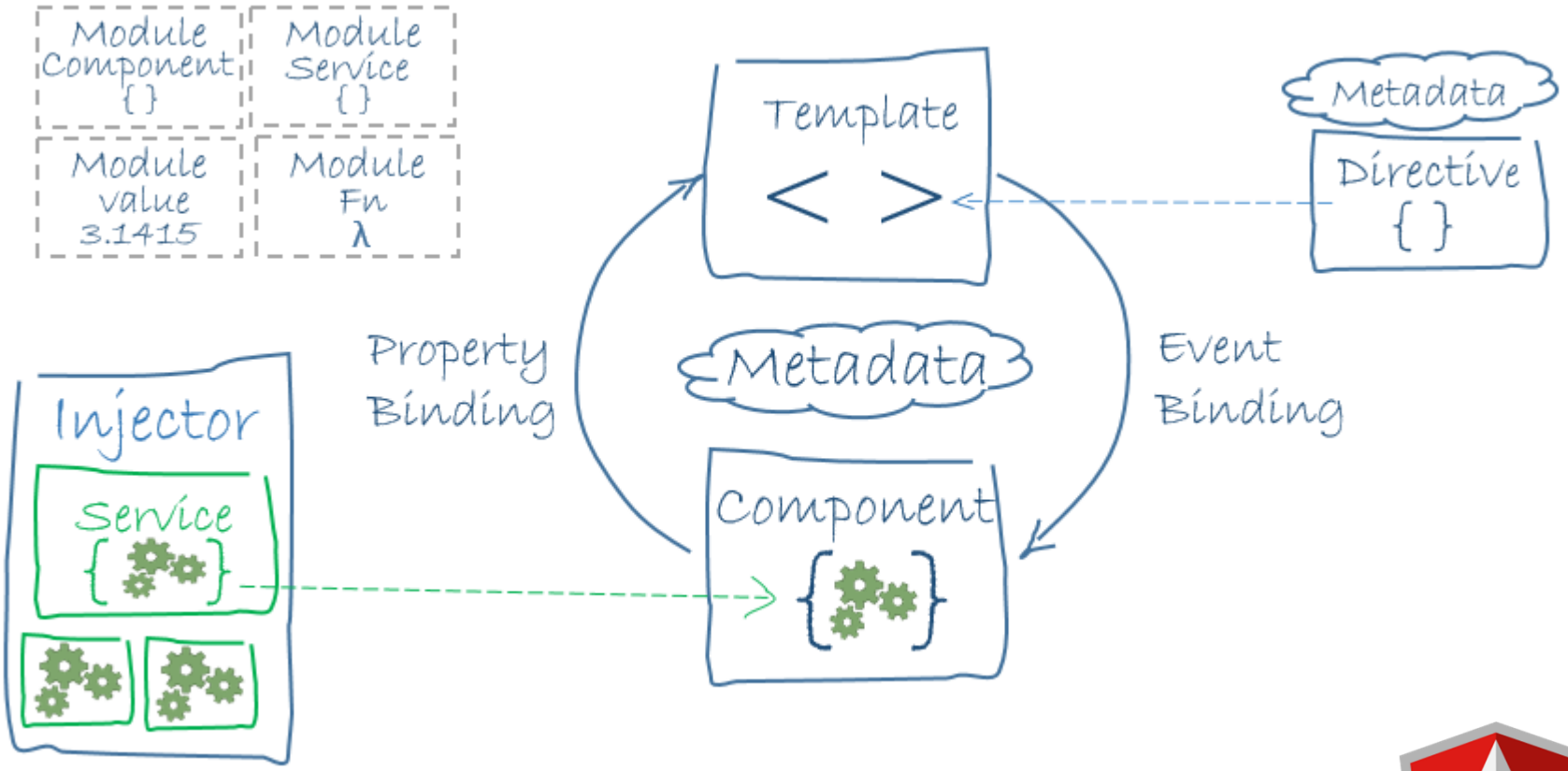


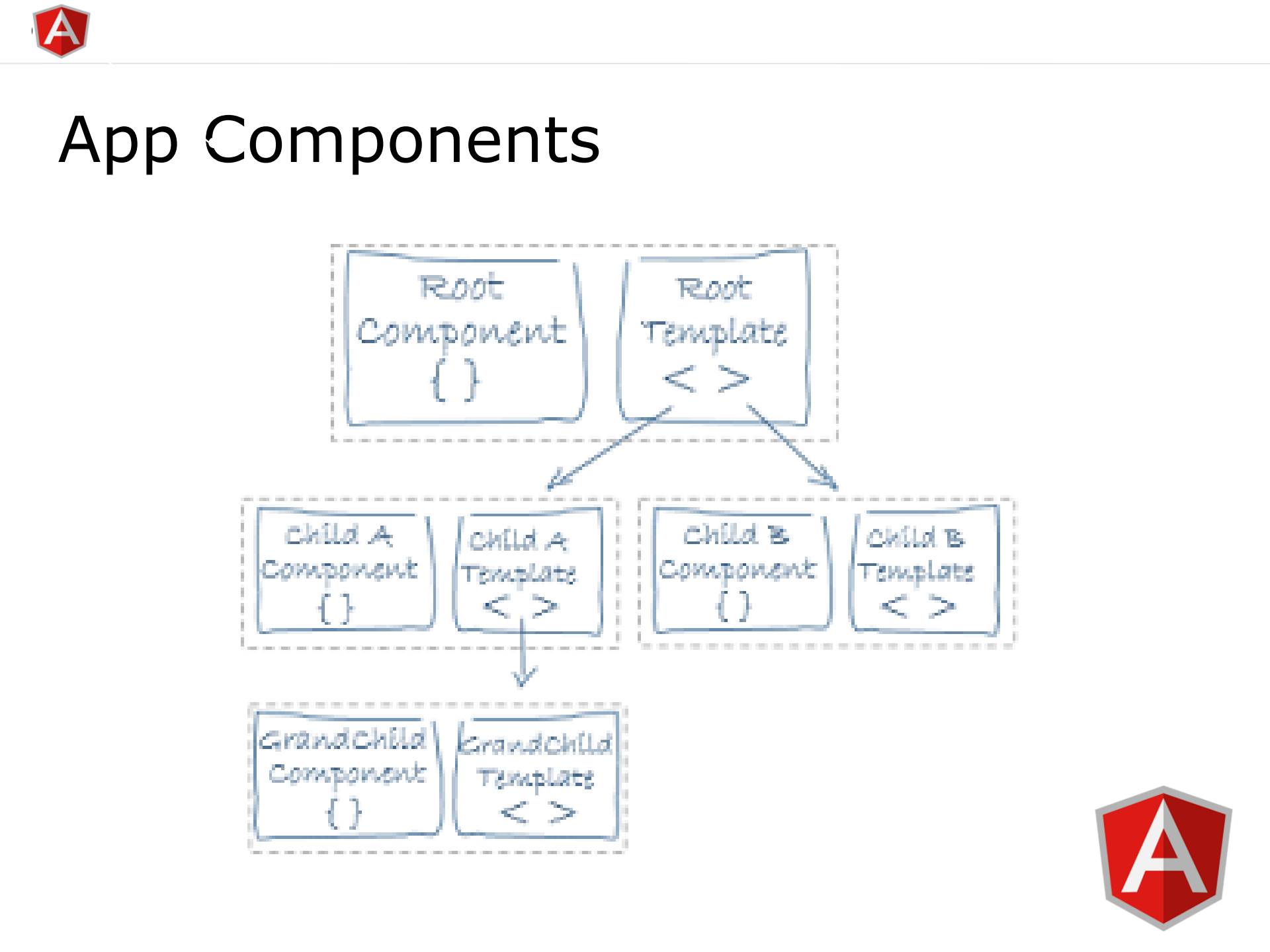

Google Trends
Explore
Sign in
Worldwide
Past 12 months
All categories
Web Search
Interest over time
100
75
50
25
Feb 7, 2016
May 29, 2016
Sep 18, 2016
Jan 8, 2017

Angular 2

Components
Directives
Dependency Injection
Interpolation
2-Way binding
Services
    Promises, Observables
Routing

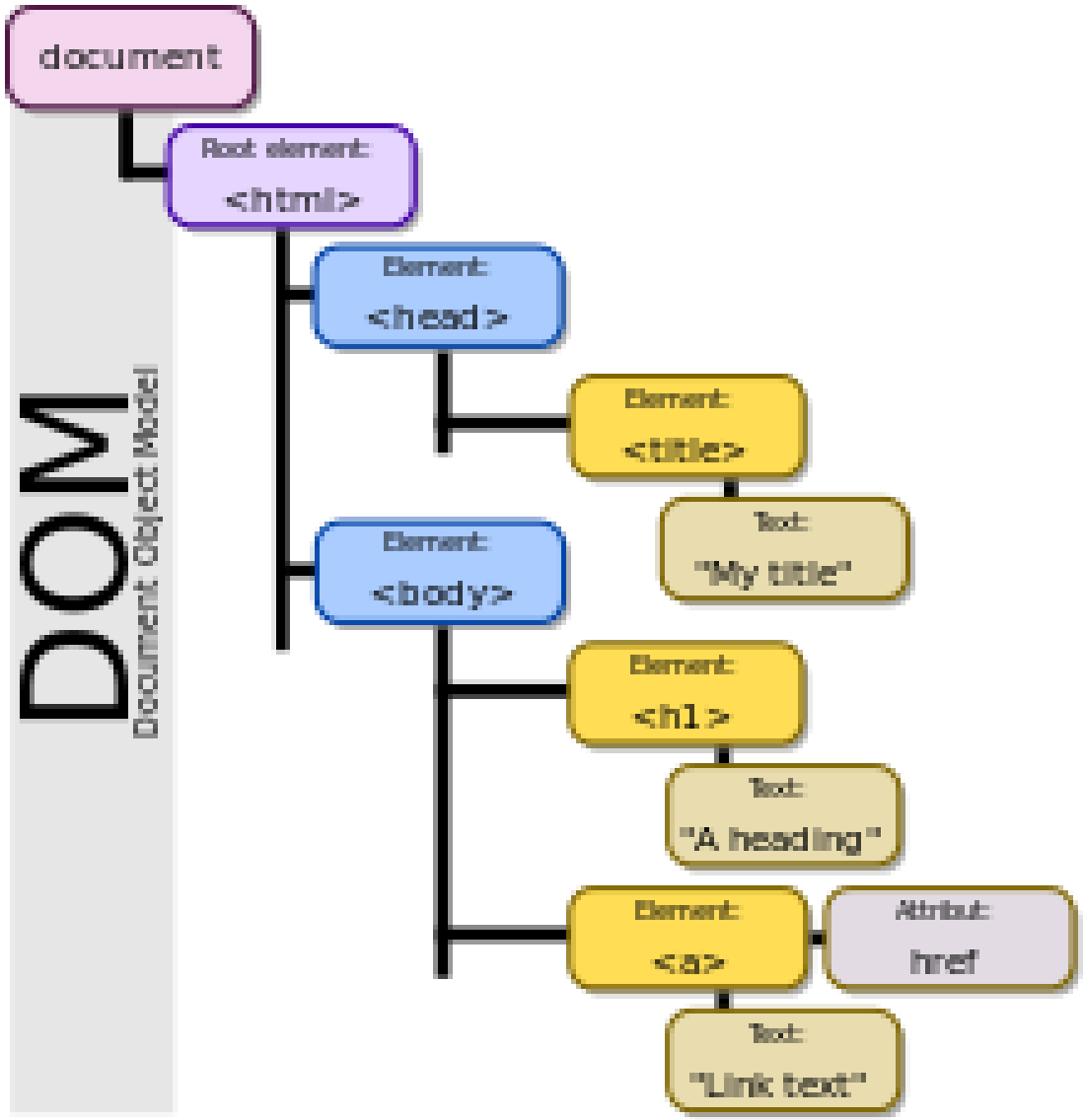
Architecture





App Components

DOM

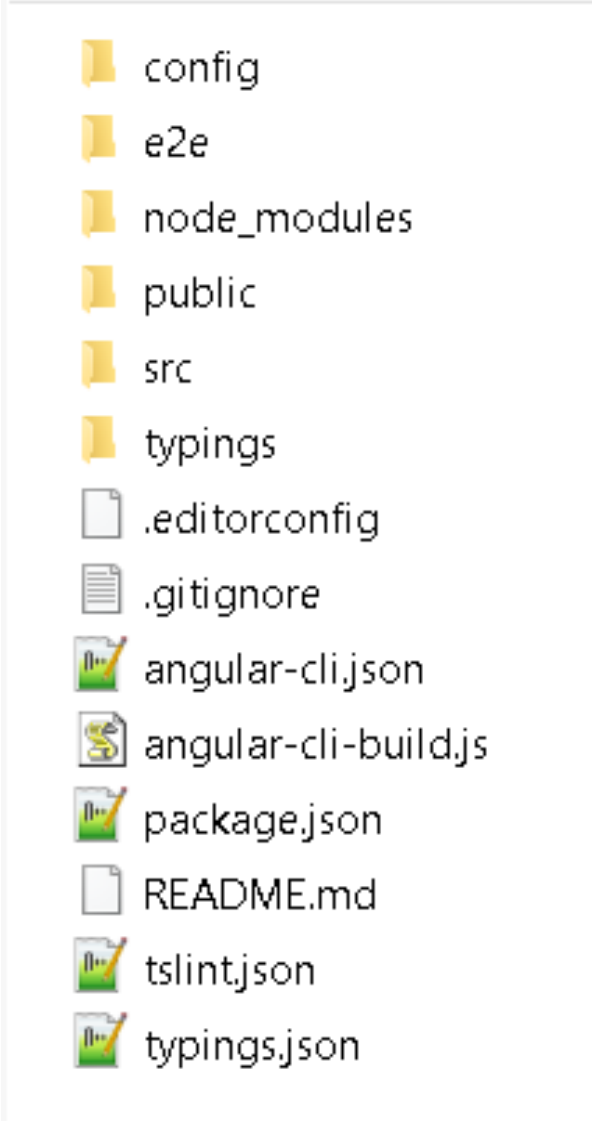

DOM
Document Object Model
document
Root element:
<html>
Element:
<head>
Element:
<title>
Text:
"My title"
Element:
<body>
Element:
<h1>
Text:
"A heading"
Element:
<a>
Attribut:
href
Text:
"Link text"

Lets code

Hello Angular!

Project structure
npm start
Great DX !!
Bootstrap App
Root component

Project Structure



app.component.ts

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

@Component({
	selector: 'app-root',
	templateUrl: './app.component.html',
	styleUrls: ['./app.component.css']
})
export class AppComponent {
	title = 'PayPal App';
}
```

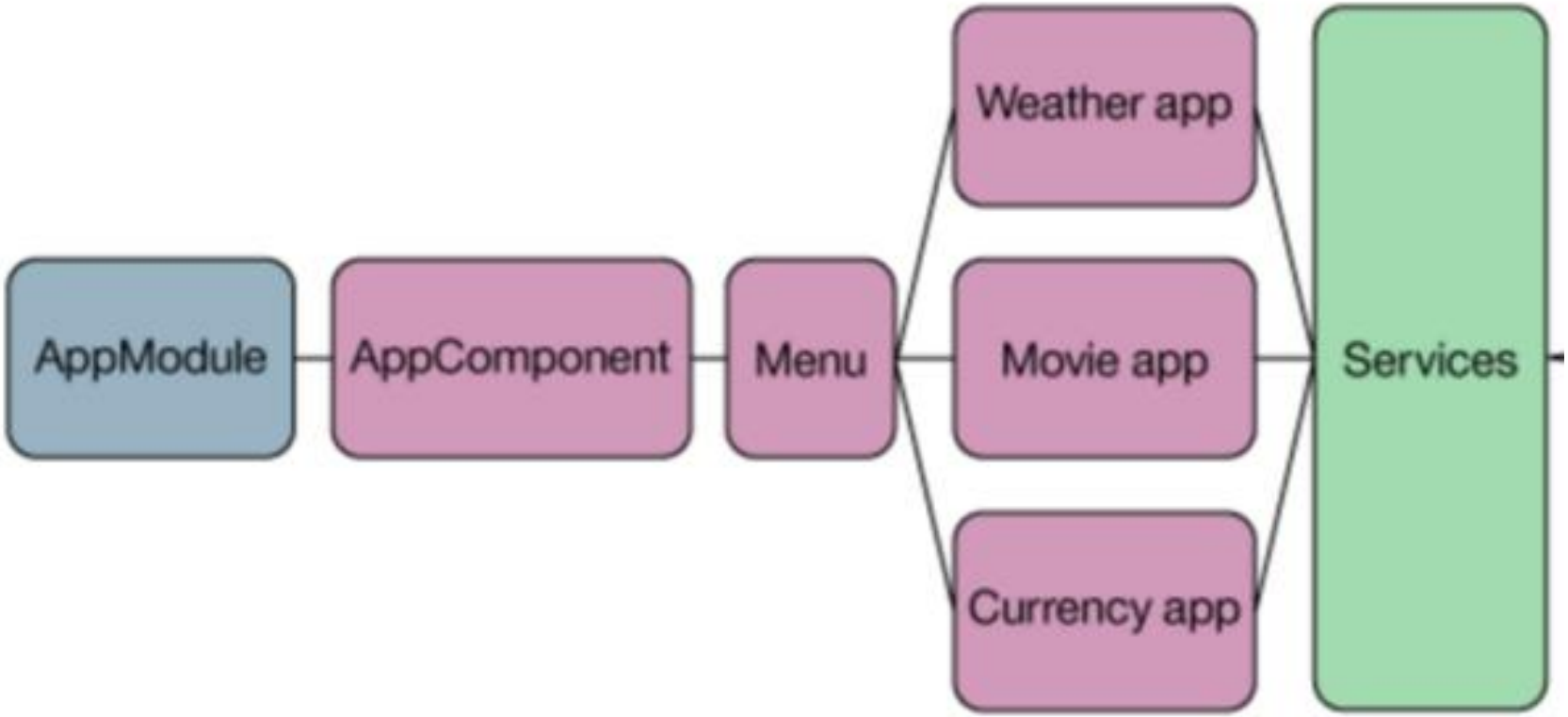
app.component.html

```
<h1>
  {{title}}
</h1>
```

app.component.css

```
#Place css for component

h1 {
        color: red;
}
```


AppModule
AppComponent
Menu
Weather app
Movie app
Currency app
Services

WebApp Weather Currency Movie

Yahoo! Weather

Enter City name ...

Enter State. Example CA for California ...

Submit

City, State, Country :
Current Condition :
Current Temperature :

Total # of all the service requests including Weather, Movie, and Currency is : 0

WebApp Weather Currency Movie

Currency Exchange Rates

Enter Currency Symbol. Example: GBP(,AUD,INR)...

Rate Details

Exchange rate relative to Euro in a JSON format: :

Total # of all the service requests including Weather, Movie, and Currency is : 0

WebApp Weather Currency Movie

Open Movie Database

Enter Movie name ...

Movie Details

Poster

Title :
Plot :
Actors :
Directed by :
Rated :
Release Date :

Total # of all the service requests including Weather, Movie, and Currency is : 0

Adding new component

```
ng g c Menu -is --spec false --flat
```

Interpolation

{{ property }}

Update DOM value using
Component property

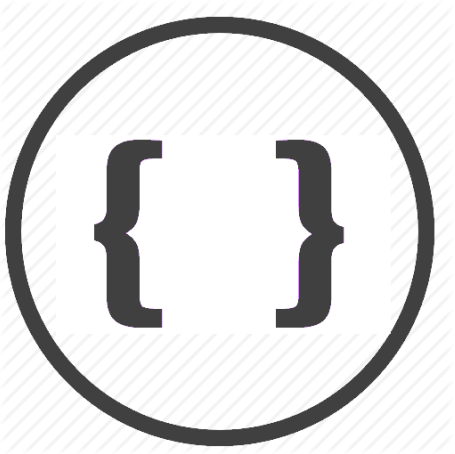
Component decorator

```
@Component({
  selector: 'my-app',
  templateUrl: '/app/app.component.html'
})
```

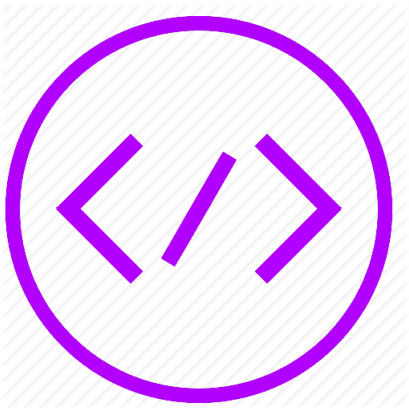
Interpolation

Property binding

```
{{value}}
```



Property



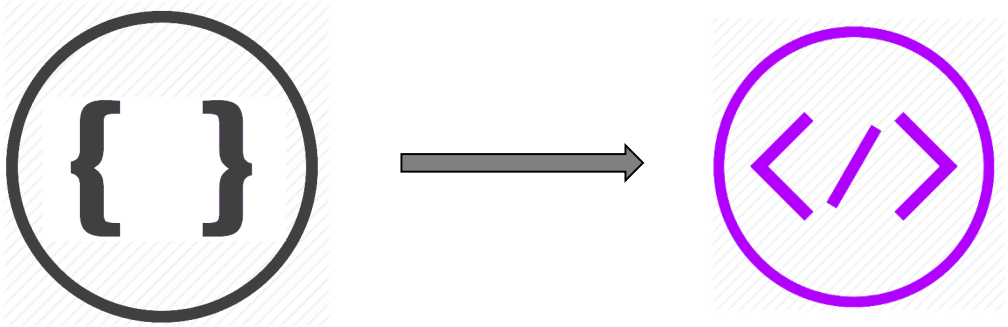
xxx.ts

xxx.html

Property binding

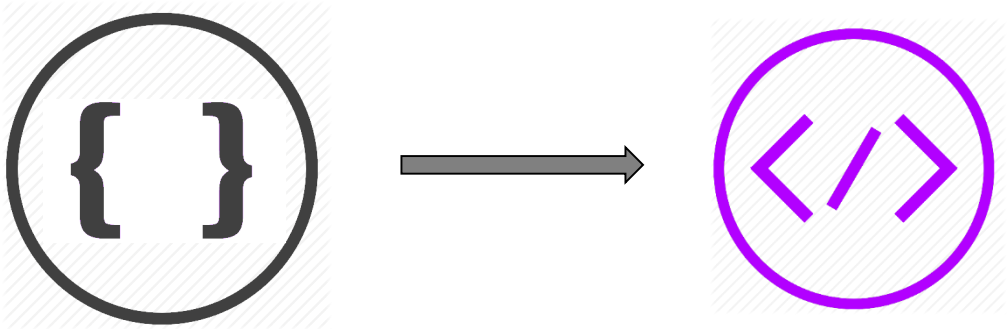
```
<img src="{{ imageUrl }}" />
<img [src]="imageUrl" />
<img bind-src="imageUrl" />
```

We will see forms 1 often, 2 we will see an example in YouTube app



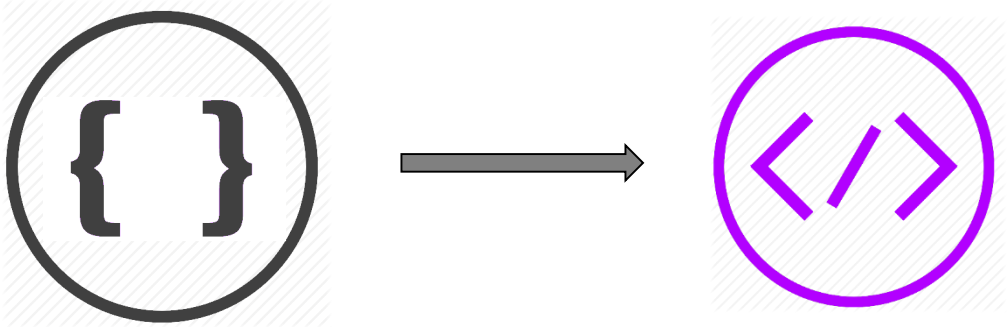
Class binding

```
<button
    class="btn btn-primary"
    [class.active]="expression" >
    Submit
</button>
```

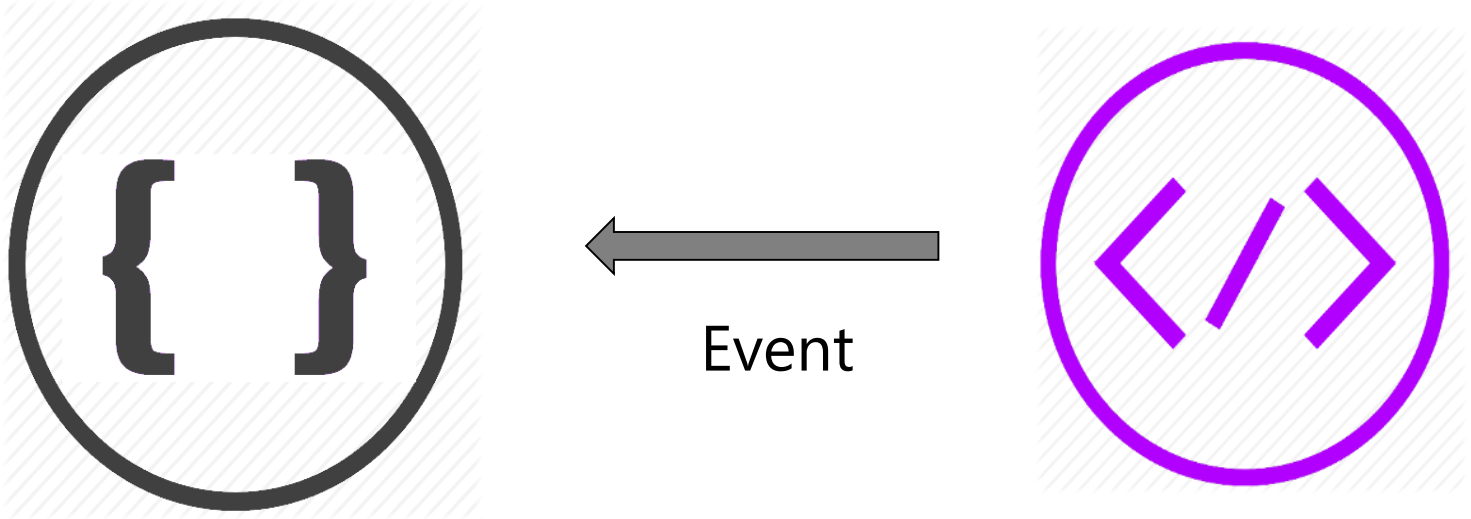


Style binding

```
<button
    class="btn btn-primary"
    [style.backgroundColor]=
    "isActive ? 'blue' : 'grey'">
    Submit
</button>
```

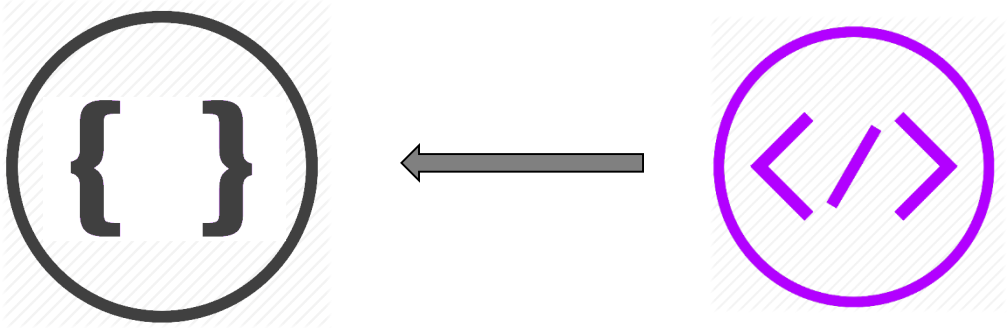


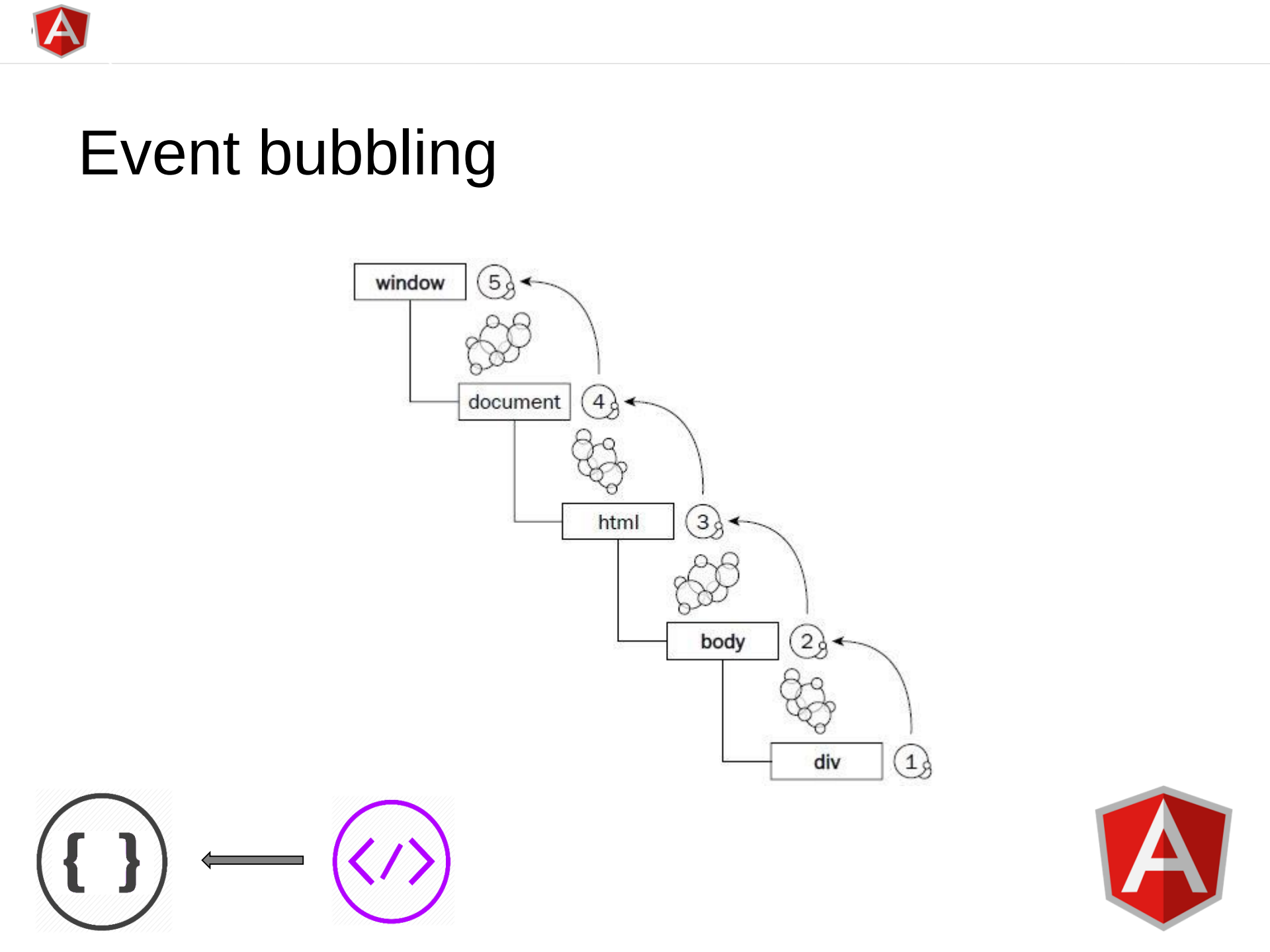
Event binding



Event binding

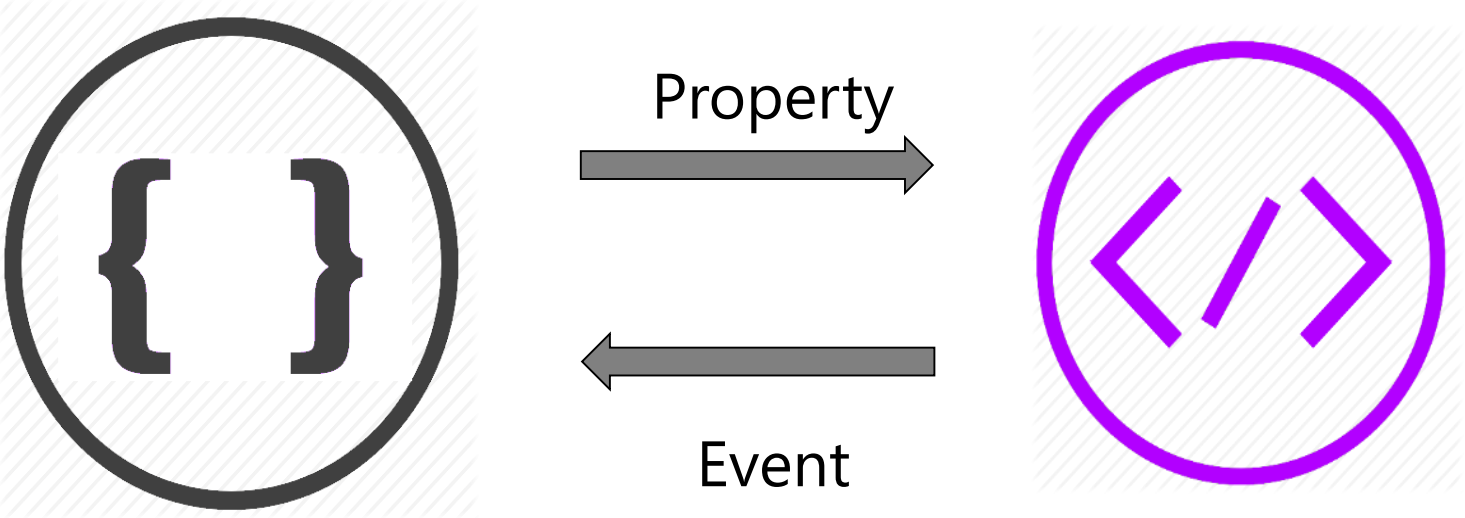
```
<button
    class="btn btn-primary"
    (click)= “onSubmit()">
    Submit
</button>
```




Event bubbling
window
5
document
4
html
3
body
2
div
1

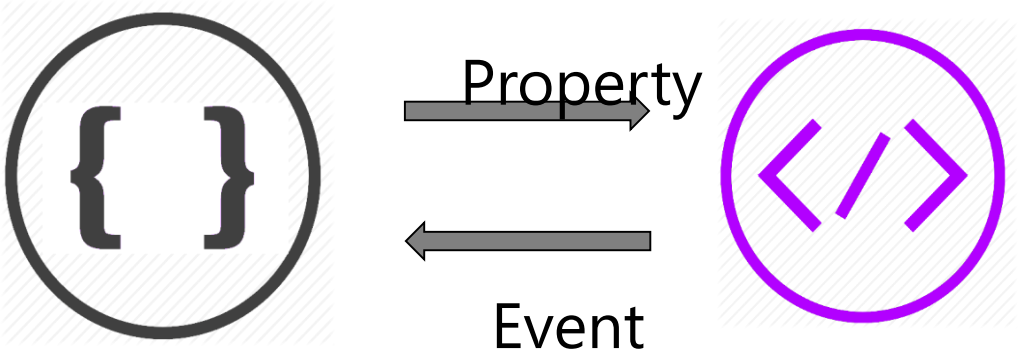
2-Way binding

ngModel



2-Way binding

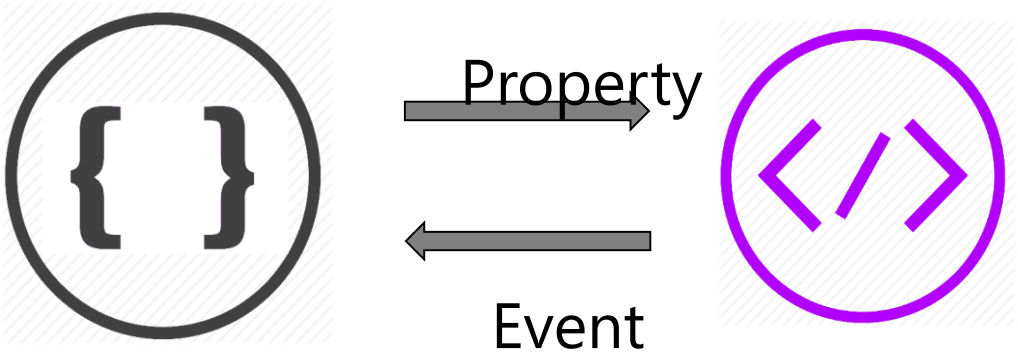
```
<input type="text"
   [value]="exp1"
   (input)="exp2" />
```



2-Way binding

```
<input type="text"
   [(ngModel)]="property" />
```

Note: FormsModule is needed!



That was a very gentle intro to Forms...

Summary

Interpolation

```
<h1>{{ title }}</h1>
```

Property binding

```
<img [src]="imageUrl" />
<img bind-src="imageUrl" />
```

Class binding

```
<li [class.active]="isActive" />
```

Style binding

```
<button [style.backgroundColor]="isActive ? 'blue' : 'gray'">
```

Summary

Event binding

```
<button (click)="onClick($event)">
<button on-click="onClick($event)">
```

Two-way binding

```
<input type="text" [(ngModel)]="firstName">
<input type="text" bindon-ngModel="firstName">
```

Complex Class

```
class Child  {
   property:Type;

   constructor(arg1:Type) {
     super(arg1);
   }

   methodProperty:(arg1:Type) => ReturnType;
}
```

Further Reading:

https://angular.io

Thank You!