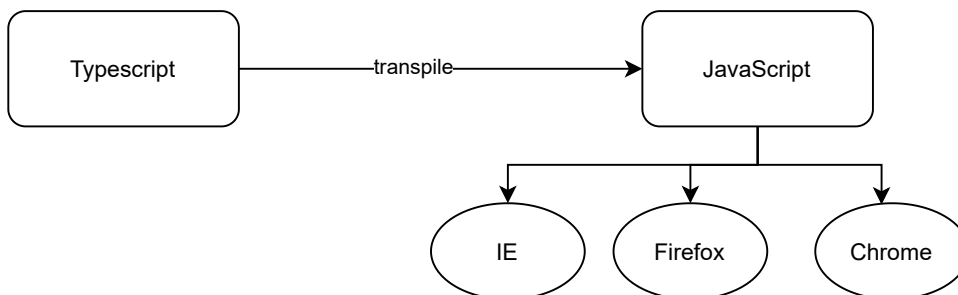


NodeJS - Server side scripting

- Node Runtime Env (NRE)
- Node Native Module - e.g. http, event, utils, fs, events etc
- Node Package Manager (NPM)

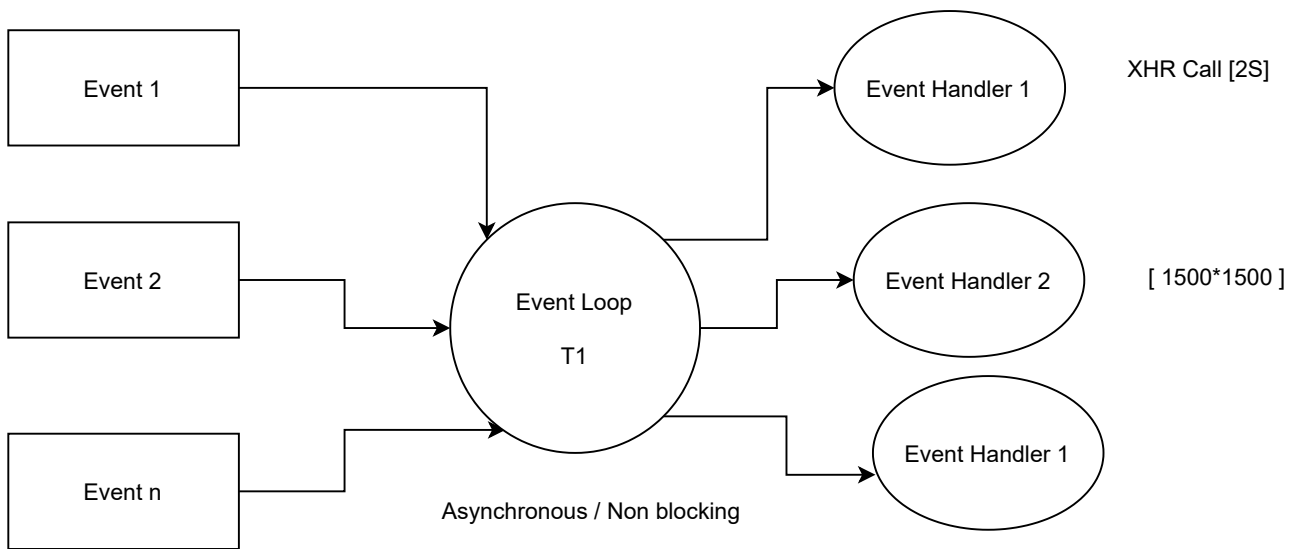
JS Lib and Frameworks -

- jQuery : Lib - DOM manipulation, ajax
- React - data change very frequently; Virtual DOM - diffing algo
- VueJS : data-binding, templates, routing etc - Evan You
- *EmberJS : changes internal API very frequently
- backboneJS : Client side MVC
- knockoutJS : 2 way data-binding, MVVM
- AngularJS (v1) : Library
- Angular (latest version) : Framework
 - 2 way data-binding
 - templates
 - routing
 - SPA
 - MV* (MVC, MVP, MVVM)
 - DOM manipulation
 - many more
- NodeJS - Server-side platform
- ExpressJS - Server-side web app framework for JS app
- JS is Single threaded programming languages
 - async behaviour
- ASP / JSP



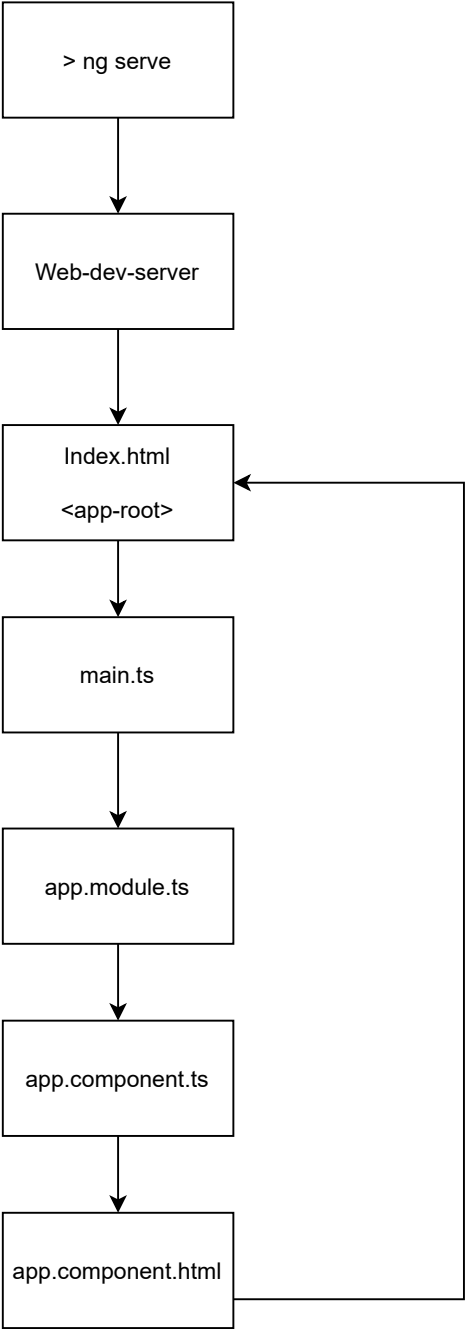
Transpilers :

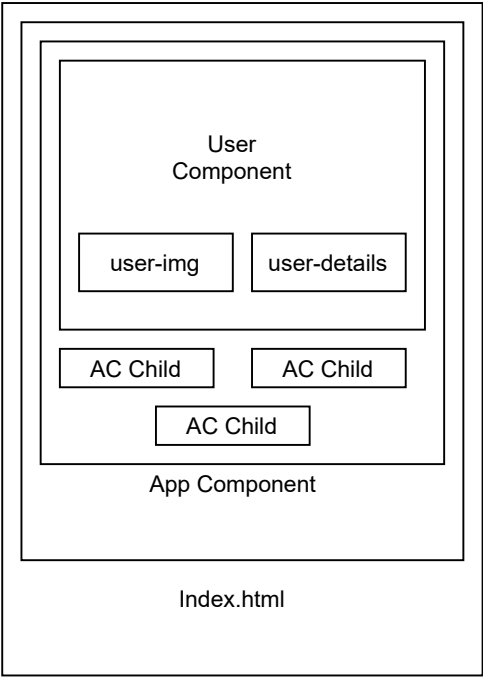
- babel
- typescript compiler
- traceur



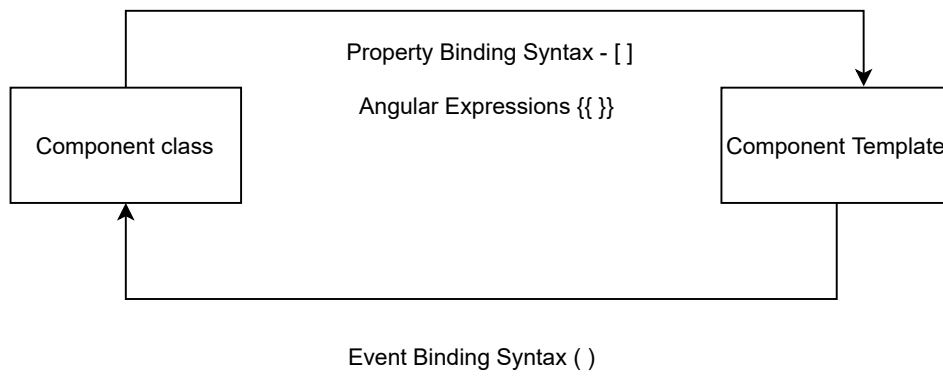
Handling Async Behaviour -

- Callbacks
- Promises
- Async ... await
- Observable





Browser
/ UI



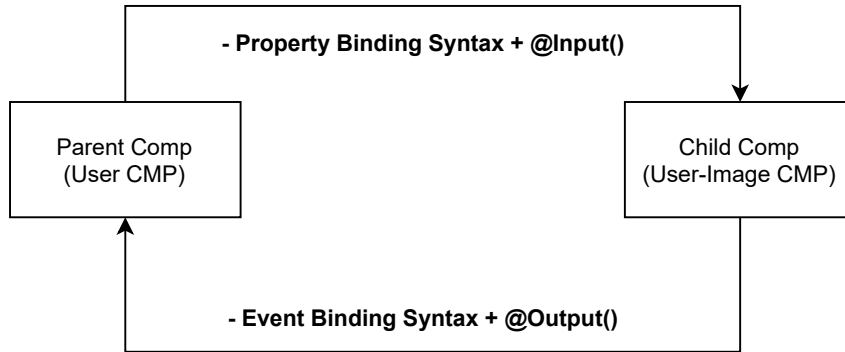
2 way data-binding -

- Property Binding + Event Binding

- Banana in the box [(ngModel)]

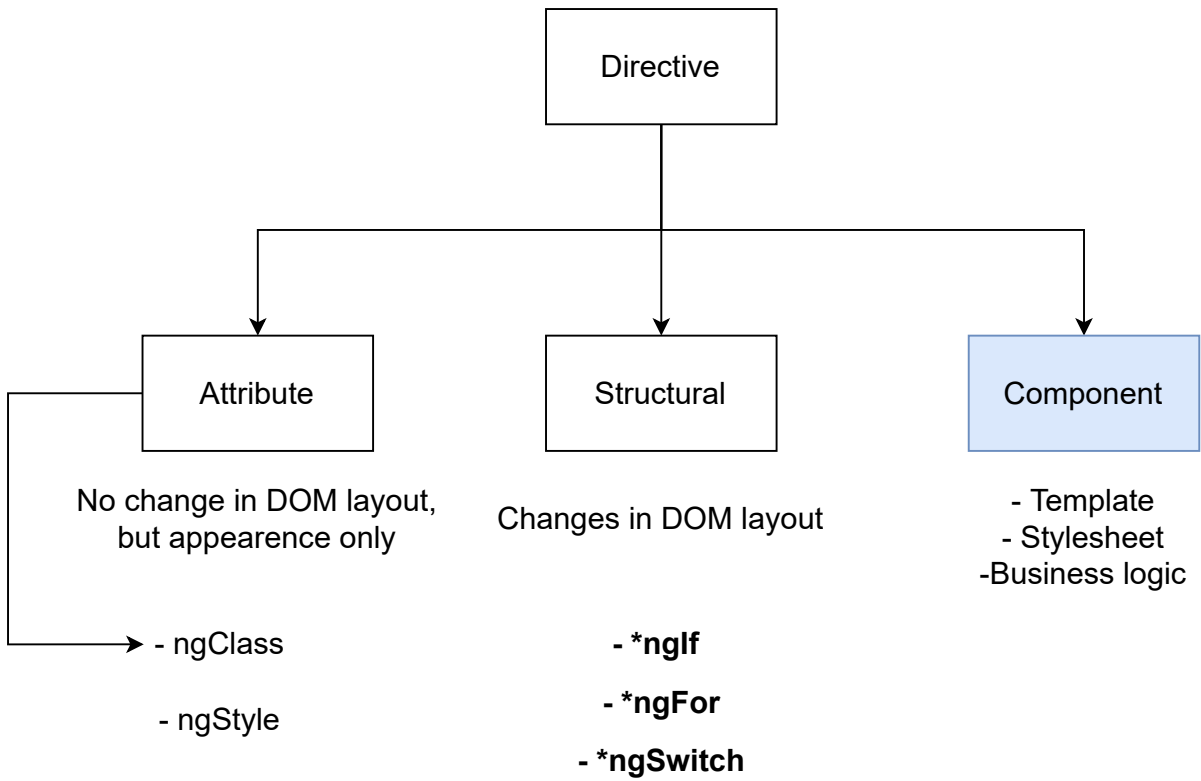
> npm install bootstrap@4 --save

Component Communication



Parent to Child Communication

- Property Binding + @Input() for model only
- Data Projection using <ng-content> for formatted data
- Services (angular building block)

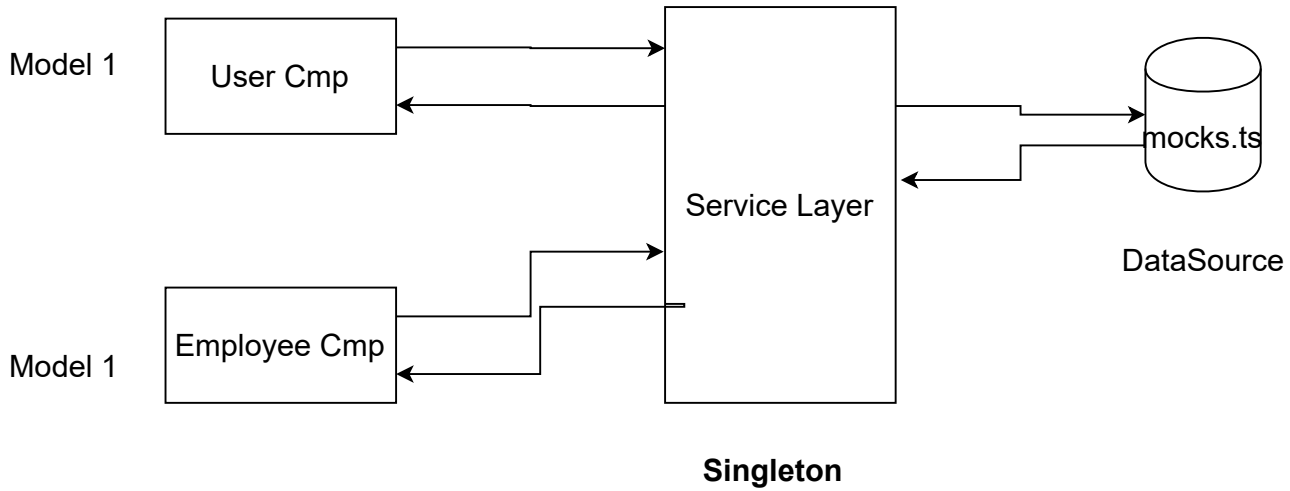
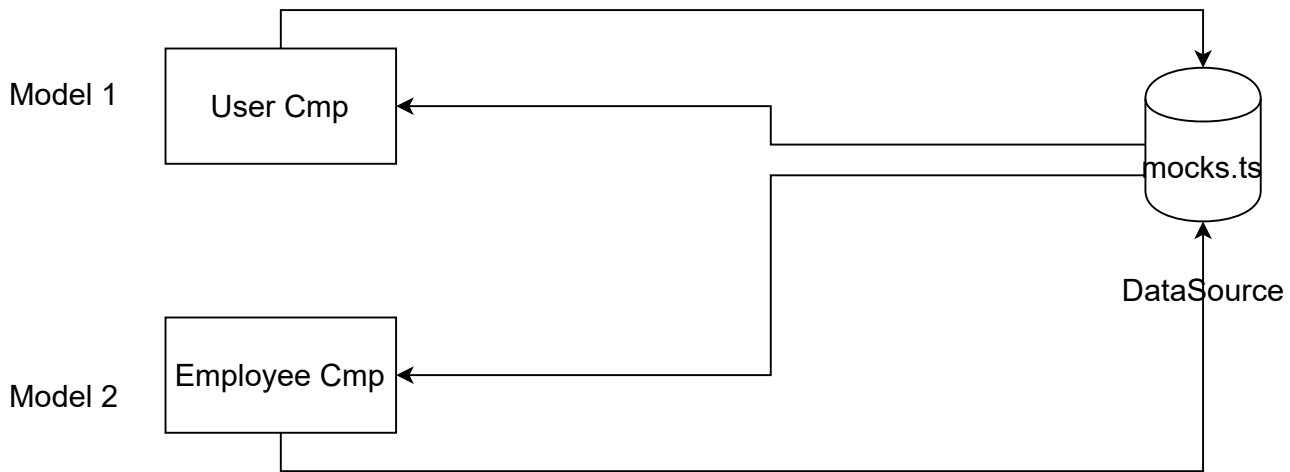


Forms

- Template Driven
 1. HTML5 Validations
 2. ngForm
- Model Driven / Reactive Forms
 1. Angular Validations
 2. FormGroup, FormControl, FormBuilder

Form and Form Controls -

1. ngValid
2. ngInvalid
3. ngDirty
4. ngPristine
5. ngTouched
6. ngUntouched



- Data consistency
- Dependency Injection
- Single Responsibility Principle
- Client-side business logic

