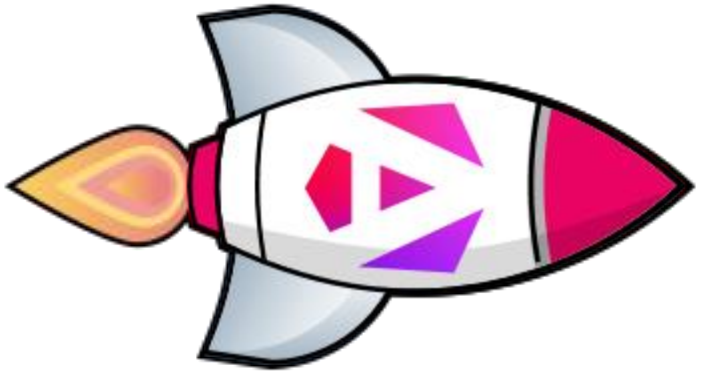
ANGULAR
ARCHITECTS

Synchronisation

Alexander Thalhammer | @LX_T

Outline - Runtime Performance

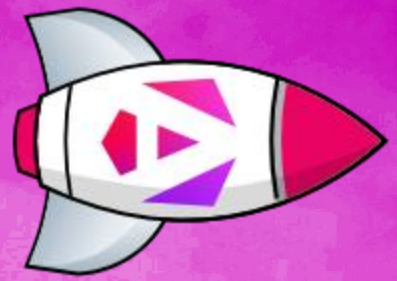


- Synchronization
- Runtime Best Practices

Change Detection

- Out of bound change detection
- Zone pollution by 3rd party libs
- HTML template optimization
  - with state or flags
  - with Angular Pipes
- Going zoneless

Agenda



Change Detection in Angular

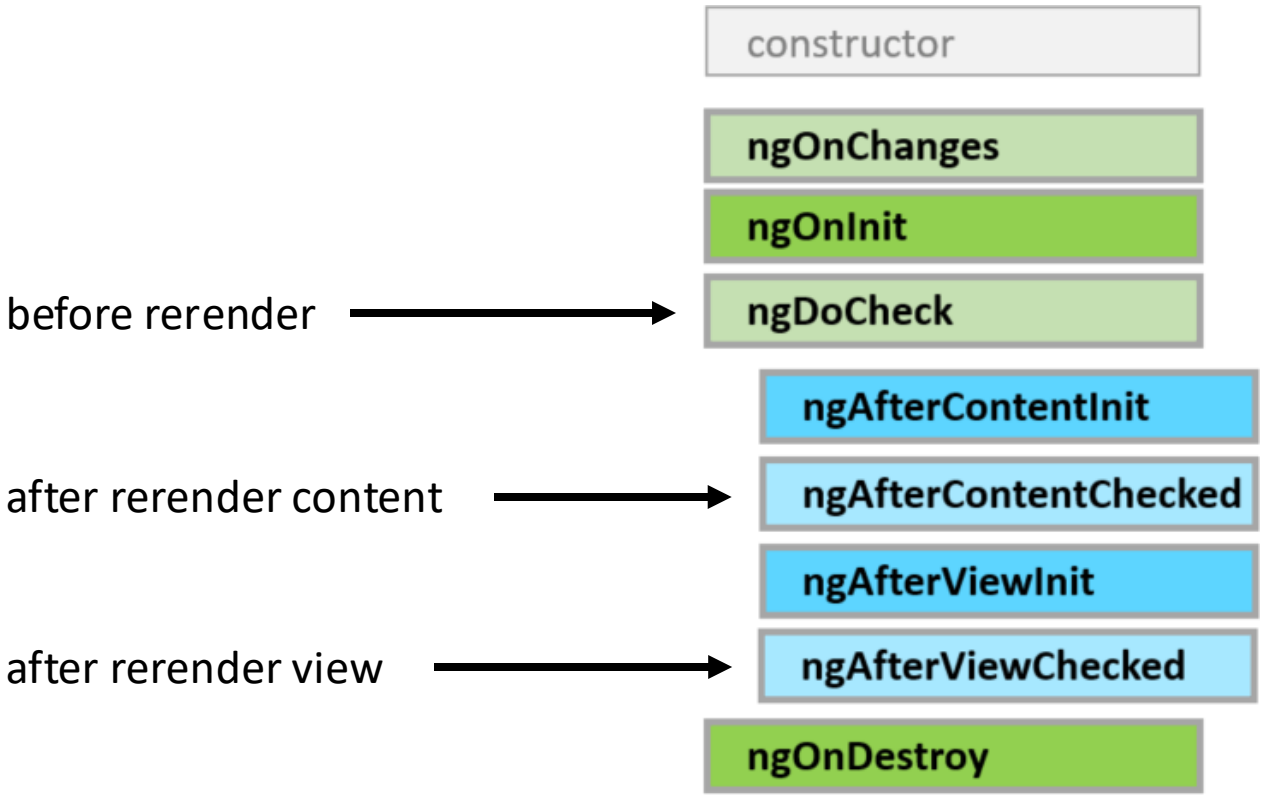
ANGULAR
ARCHITECTS

Out of bound change detection

- Problem: Local state change triggers CD in other comps
  - E.g. Input field keydown triggers CD in parent/child components

- Identify:
  - Use the (© AngularArchitects) infamous blink() or
  - use the Angular DevTools Profiler
  - console.log() in CD lifecycle hook (e.g. ngDoCheck)

Change Detection – Check Components

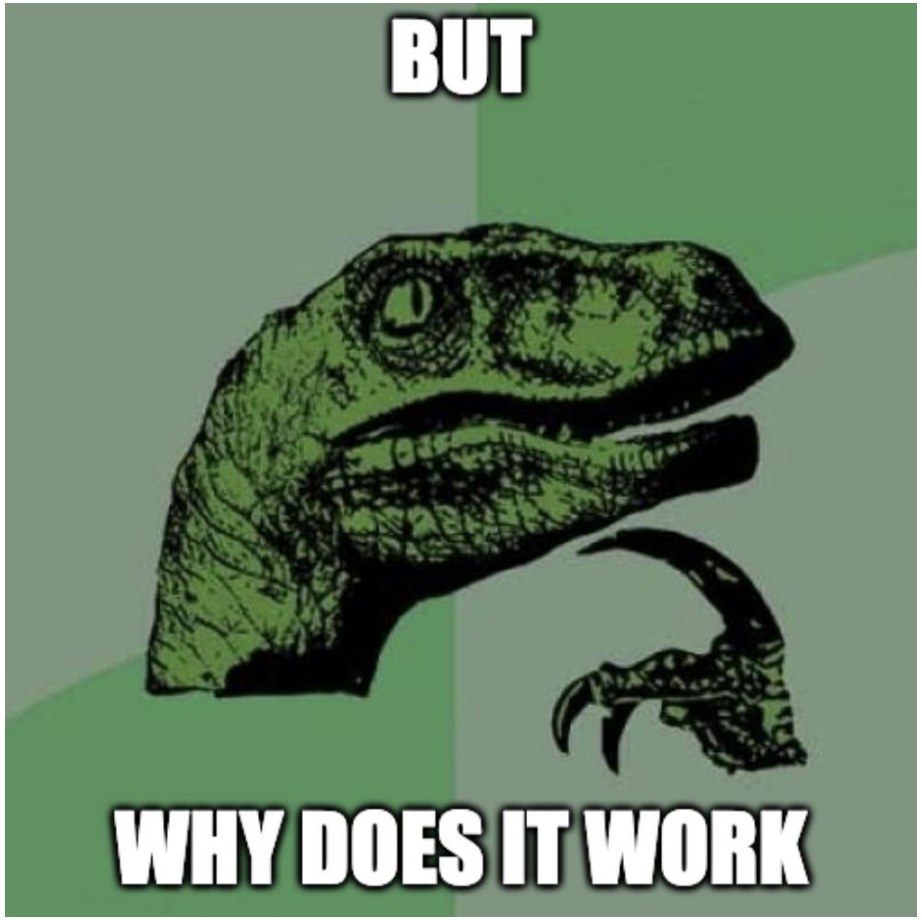


Demo

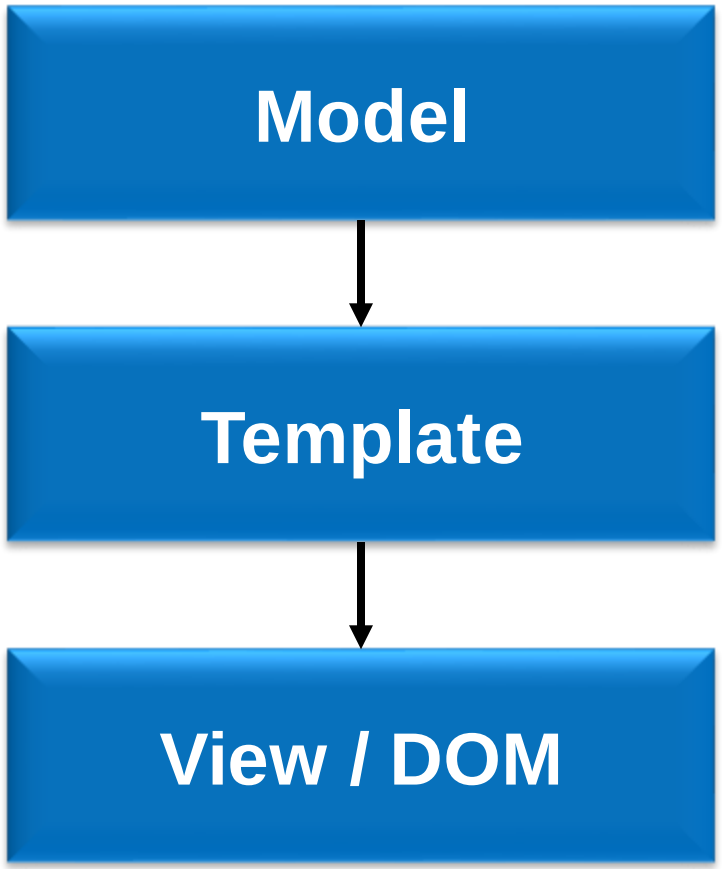
Change Detection

ANGULAR
ARCHITECTS

Change Detection



DOM Rendering



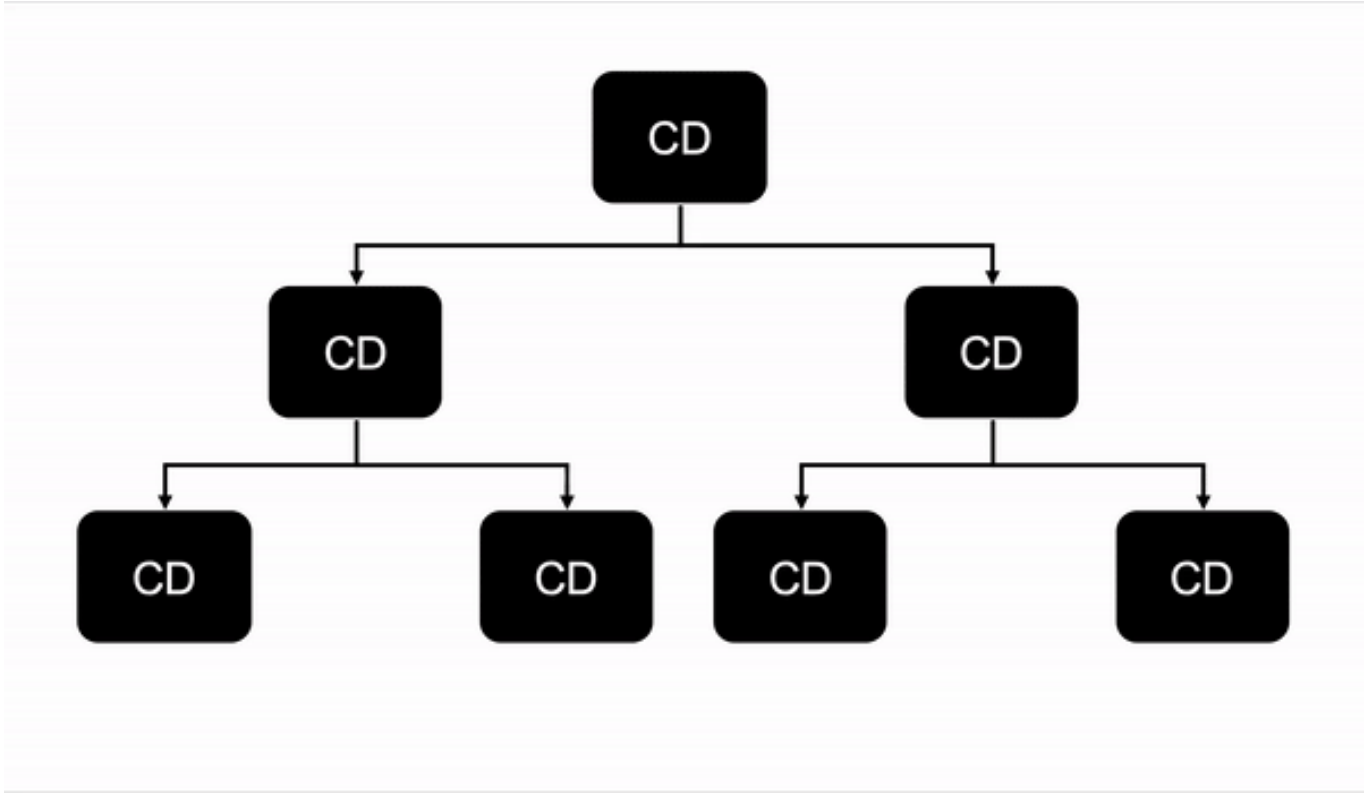
Change Detection

- 1.) User or App changes the model (e.g. input, blur or click)
- 2.) NG CD runs for **every component** (from root to leaves)
- 3.) Check / rerrender the component's view (DOM)

Digression – what triggers CD?

- Many browser events (click, blur, keyup, etc.)
- XMLHttpRequests (AJAX / HttpClient)
- setTimeout() and setInterval()
  - often used as a "hack" to trigger CD
  - meaning: "I have no clue how this works"
- Websockets

Change Detection – From Root To Leaves



Img src: https://mokkapps.de/blog/the-last-guide-for-angular-change-detection-you-will-ever-need/

Demo

Default Strategy & Blink

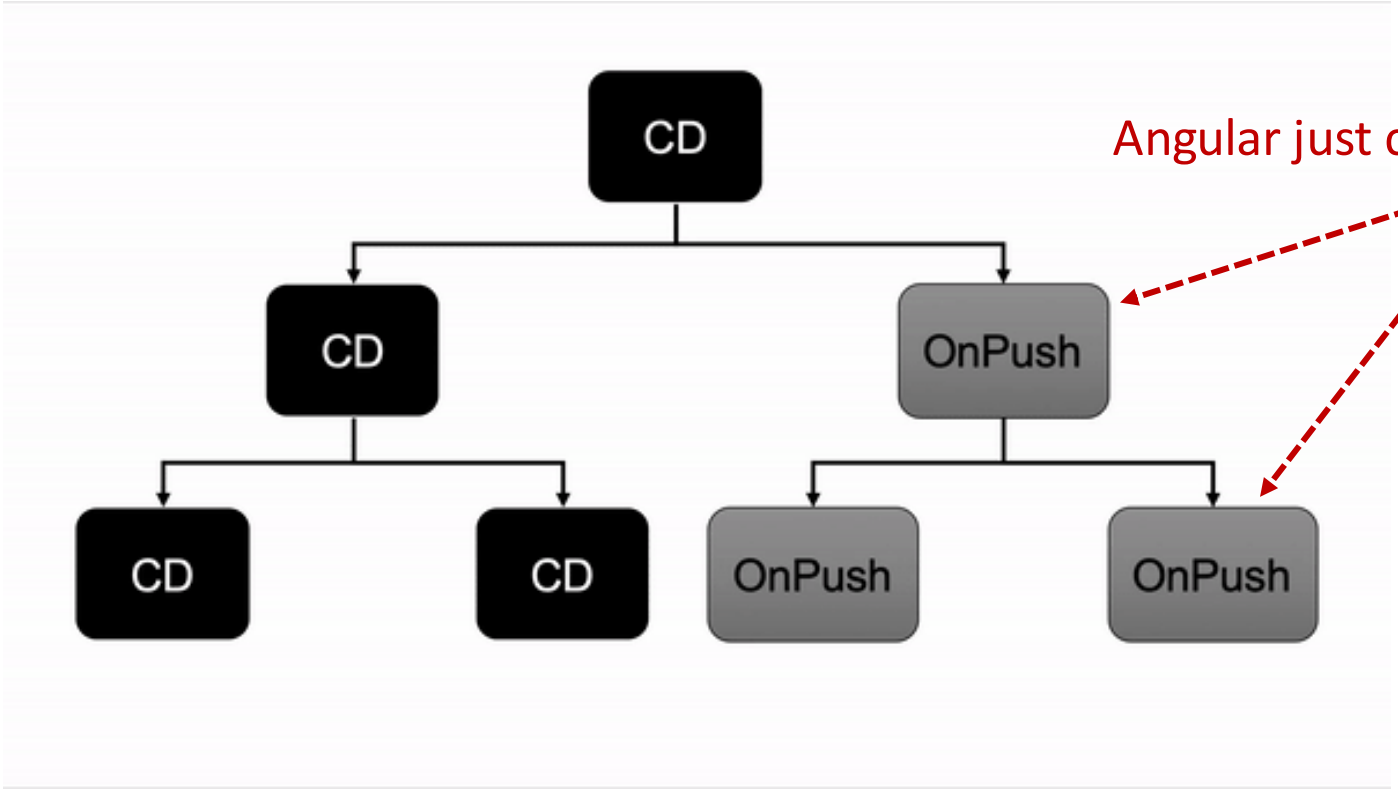
ANGULAR
ARCHITECTS

Performance-Tuning with OnPush

Activate OnPush Strategy

```
@Component({
    […]
    changeDetection: ChangeDetectionStrategy.OnPush
})
export class FlightCardComponent {
    […]
    @Input({ required: true }) flight!: Flight;
}
```

Change Detection – OnPush Strategy

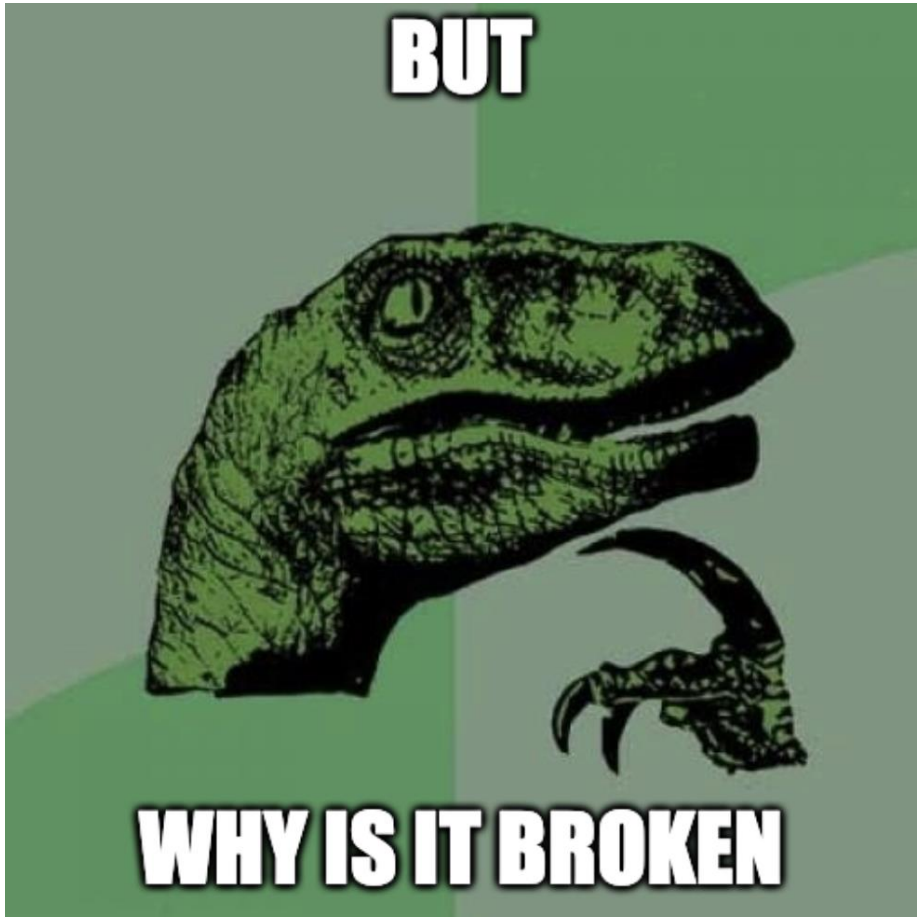


Img src: https://mokkapps.de/blog/the-last-guide-for-angular-change-detection-you-will-ever-need/

Demo

OnPush Strategy

Change Detection



"Notify" about change?

- 1 Raise event within the component or its children (via zone.js)
- 2 Change bound data (**@Input** or input/model signal)
  - OnPush: Angular just compares the object reference!
  - e. g. oldFlight !== newFlight (BTW: like ngOnChanges)
- 3 Emit a bound observable into the **async** pipe | or update a **signal**
  - {{ flights$ | async }} | {{ flights() }}
- 4 Do it manually (cdr.**markForCheck()**)
  - Don't do this at home ;-)
  - But there are reasonable cases (where we can neither use 2 nor 3)
- 5 Attaching or detaching a view using ViewContainerRef

CDR - markForCheck() vs detectChanges()

- Use CDR.markForCheck() to **notify** the CD cycle if using **OnPush**
  - Running up the component tree
  - Useful when you're bypassing the ChangeDetectionStrategy.OnPush e.g. by mutating some data or you've just updated the components model

- Use CDR.detectChanges() to **trigger CD immediately** for this component and it's **children** respecting the its/their CD strategy
  - Running down the component tree
  - Useful when you've updated the model after angular has run it's change detection, or if the update hasn't been in Angular world at all

- For the whole app (from root to leaves) use ApplicationRef.tick()

Set OnPush as default

- Add to angular.json / project.json schematic

```
"@schematics/angular:component": {
  "changeDetection": "OnPush",
  "style": "scss"
},
```

- Add an ESLint rule

```
"@angular-eslint/prefer-on-push-component-change-detection": "warn"
```

- OnPush in every component?
  - well yes, but
  - optional in smart components (and root)

Zone pollution by 3rd party libs (charts)

Demo

Zone Pollution

ANGULAR
ARCHITECTS

Zone pollution by 3rd party libs (charts)

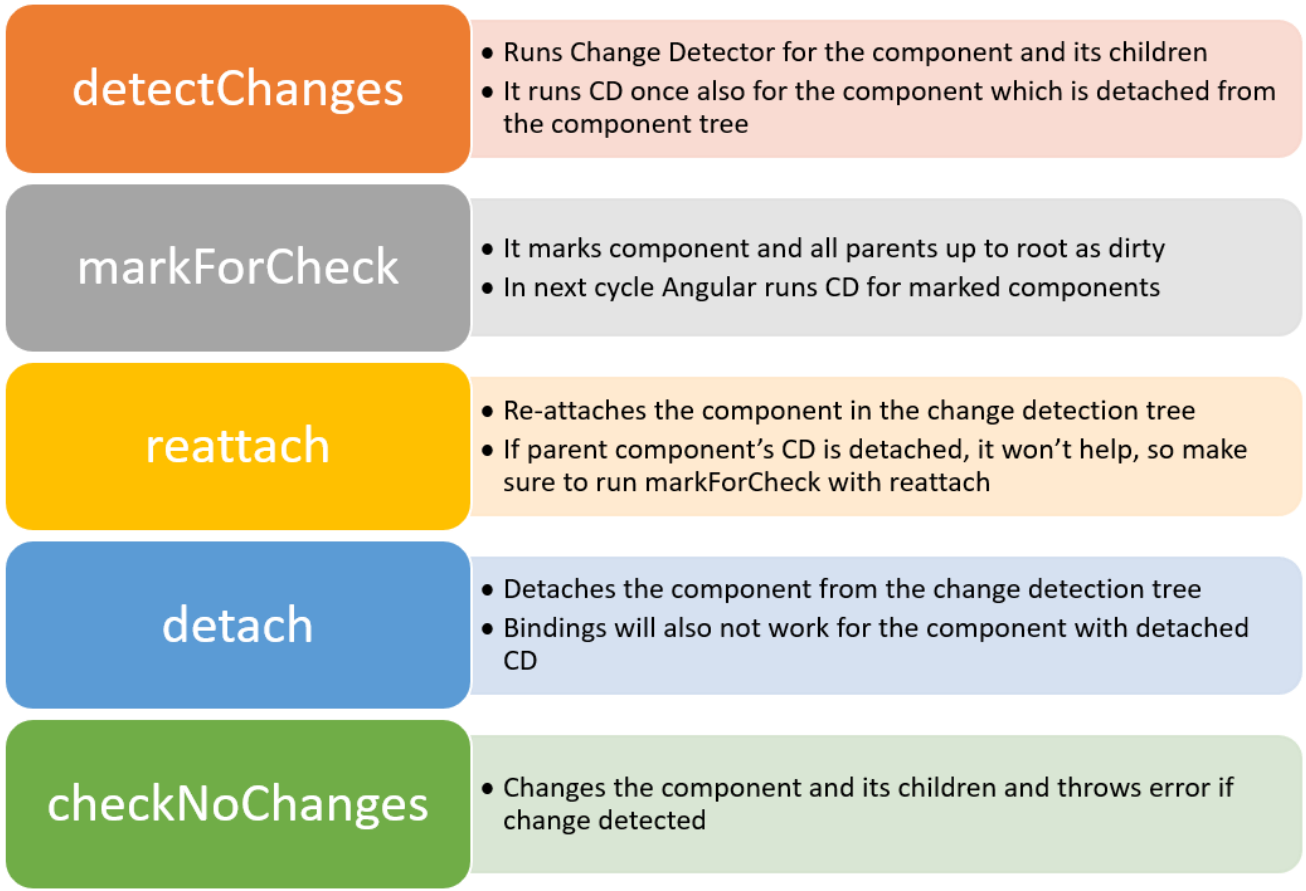
- Problem: Callbacks that trigger redundant change detection cycles
- Identify: Use the infamous blink() or the Angular DevTools Profiler
  - E.g. MouseEvent listeners
  - **requestAnimationFrame()** or
  - **setInterval()**
  - a live watch
- Solution: Run outside of NG Zone
  - Inject (private ngZone: NgZone)
  - Call this.ngZone.runOutsideAngular(() => doStuff)
  - https://angular.io/guide/change-detection-zone-pollution
- Alternative: Using cdr.detach() for components

Demo

Fixed Zone Pollution

ANGULAR
ARCHITECTS

ChangeDetectorRef API, once more



Img src: https://www.telerik.com/blogs/simplifying-angular-change-detection/

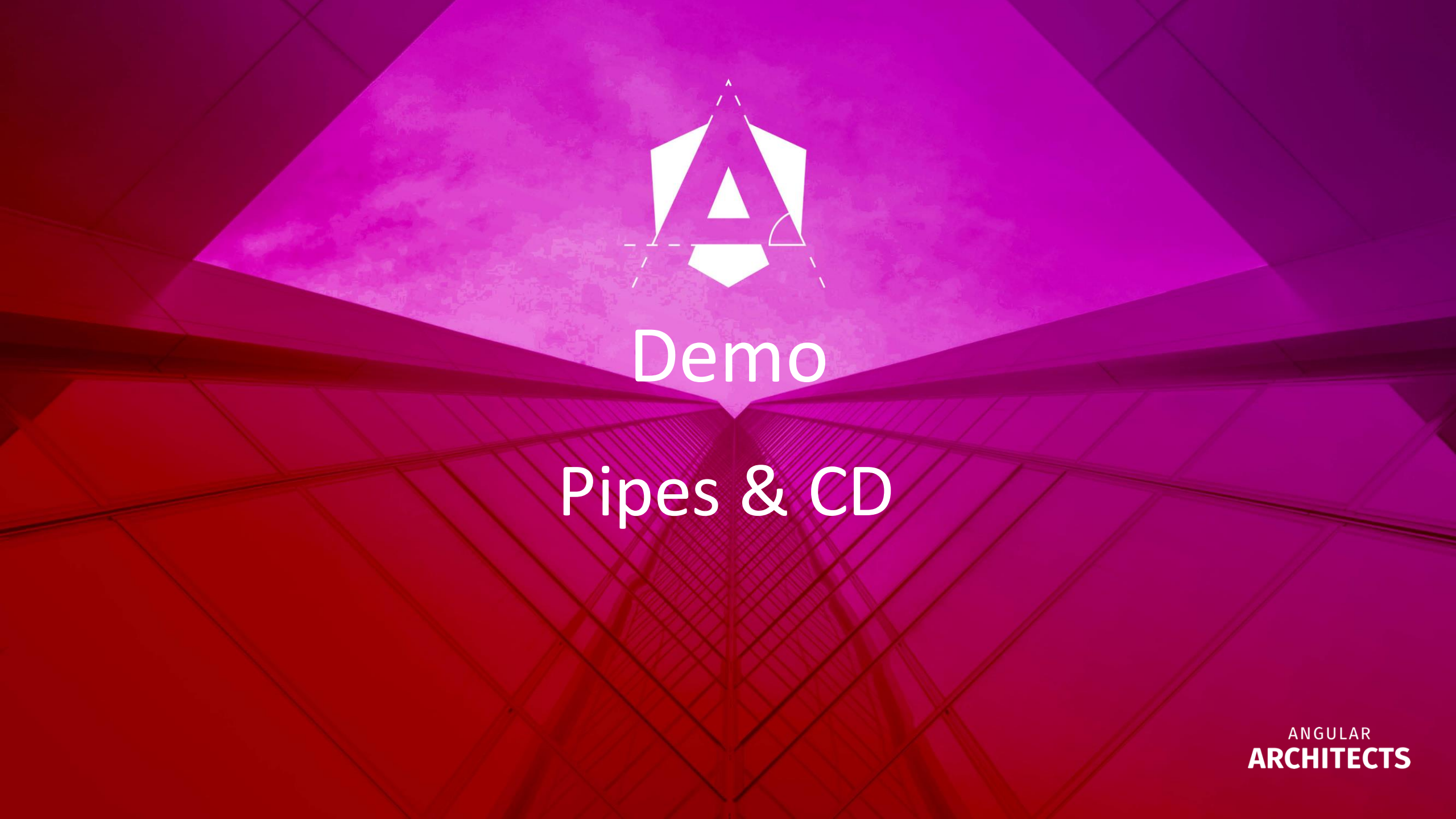
Lab 06 Change Detection

Optimization with state or flags

- Problem: Redundant calculations for conditions
- Identify: Methods being executed in **@if** (*ngIf) statements
- Solution:
  - Use StateManagement like subjects or
  - use signals or
  - use boolean flags or strings, that only change when they should

Optimization with Angular Pipes

- Problem: Redundant calculations / transforming / formatting
- Identify: Methods in html templates
- Solution: Use (pure) Angular Pipes



Demo

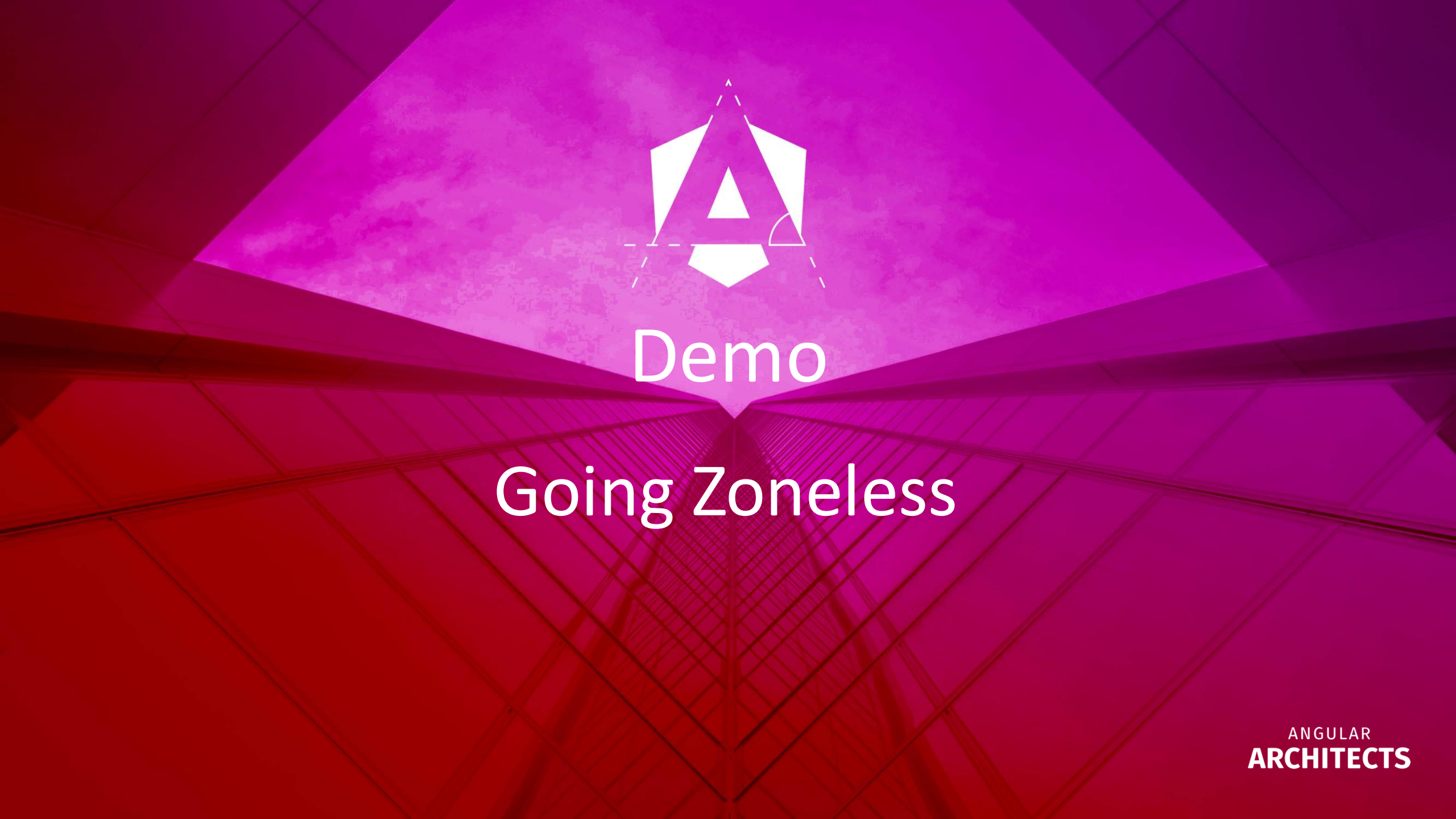
Pipes & CD

ANGULAR
ARCHITECTS

Zoneless Angular

- No zone.js event bindings

- Need to make sure to **notify** Angular about changes
  - (see notification options above: 2, 3 or 4)
  - To trigger Change Detection and thus DOM updates



Demo

Going Zoneless

ANGULAR
ARCHITECTS

Change Detection

- **Out of bound change detection**
- **Zone pollution by 3rd party libs**
- HTML template optimization
  - with state or flags
  - with **Angular Pipes**
- Going zoneless

Recap

References

- Minko Gechev (@mgechev) for Angular on YouTube
  - https://www.youtube.com/watch?v=FjyX_hkscII
  - https://www.youtube.com/watch?v=f8sA-i6gkGQ
  - New in NG 17, 18+: https://www.youtube.com/watch?v=2M17gRQbgfI
- Resolving Zone Pollution
  - https://angular.io/guide/change-detection-zone-pollution
- Angular Performance Optimization using Pure Pipe
  - https://www.youtube.com/watch?v=YsOf90RZfss

ANGULAR
ARCHITECTS