



Angular Advanced short recap – day #1



Peter Kassenaar
info@kassenaar.com

Why devs and enterprises like Angular...



CLI



Router



HTTP



Forms



Animations



i18n



Testing



Language
Services

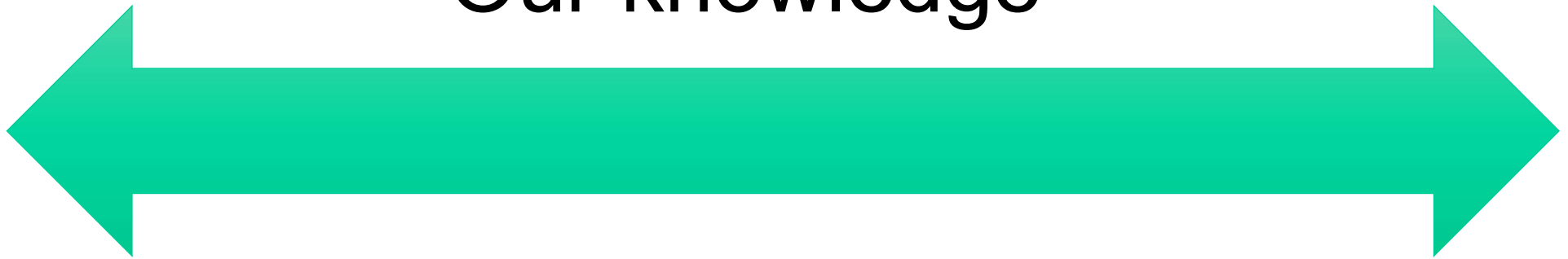


Universal



Material &
CDK

Broadening
Our knowledge



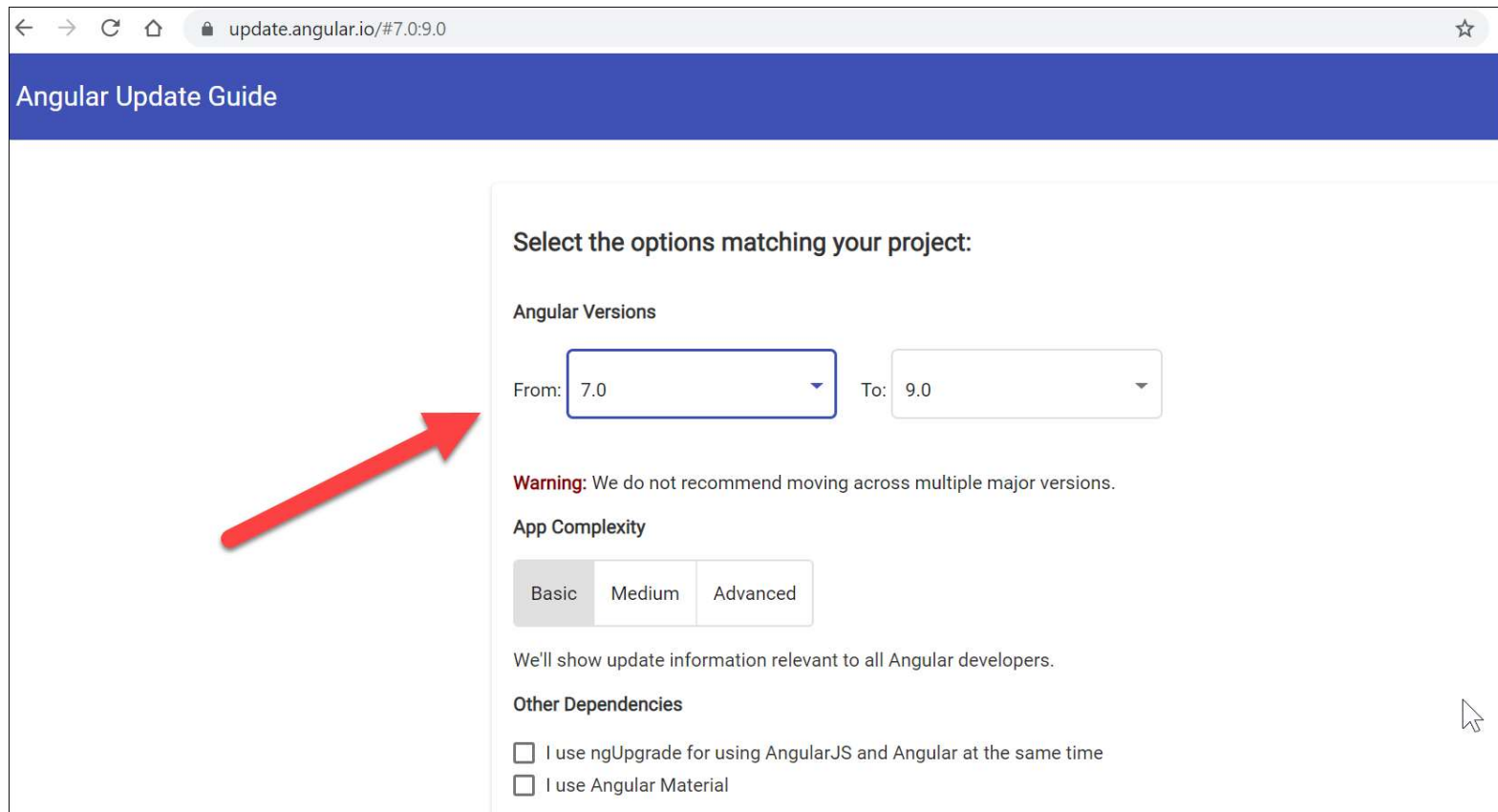


Recap day 1: **Architecture**

- Angular CLI – general **Architecture**
 - Flags, Components, Services, Modules, local/global.
- NG applications with multiple **modules**
 - Using multiple modules in your app
 - Modules and the router
- **Lazy loading** modules
 - `PreloadAllModules`
 - Custom Loading **strategies**
- **Design patterns** – a choice!
 - Content Projection
 - Smart/View Components

On ng update

- DON'T try to update over **major versions** (e.g. from NG 11 → NG 13)



Angular Update Guide

update.angular.io/#7.0:9.0

Select the options matching your project:

Angular Versions

From: 7.0 To: 9.0

Warning: We do not recommend moving across multiple major versions.

App Complexity

Basic Medium Advanced

We'll show update information relevant to all Angular developers.

Other Dependencies

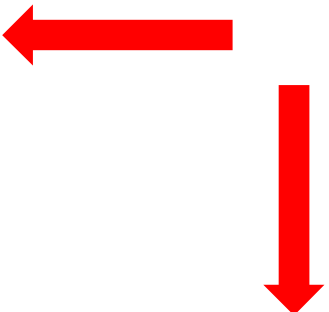
☐ I use ngUpgrade for using AngularJS and Angular at the same time

☐ I use Angular Material

<https://update.angular.io/>

Configuring the router

```
let routerConfig: ExtraOptions = {  
  enableTracing: true  
  preloadingStrategy: PreloadAllModules  
};  
  
@NgModule({  
  imports: [RouterModule.forRoot(routes, routerConfig)],  
  exports: [RouterModule]  
})
```



Reload data on reusing a route?

Routing events are actually **Observables**.

Which means we can subscribe! And do something like:

```
this.router.events
  .subscribe(event =>{
    console.log(' router event: ', event);
  })
```

(Don't forget to inject router:

```
constructor(private router: Router) {  })
```

Or, in a more reactive way of programming:

```
this.router.events.pipe(  
    filter(event => event instanceof NavigationEnd),  
    map(...),  
    ...  
) .subscribe(event => {  
    console.log(' router event: ', event)  
});
```

Handle events as desired - logging, setting variables, updating state, ...

Sending data with a route

- Add a `data: { ...}` property to the route
- Send data that you don't want to appear in the URL...
- ...but you want to be accessible in the receiving component

```
...: [  
  {  
    path: 'ikea',  
    component: IkeaComponent,  
    // sending data with the route - mostly data you DON'T want  
    // to appear in the browser URL, for instance:  
    data: {  
      location: 'Utrecht',  
      postcode: '3526 AP',  
      address: 'Van Rensselaerlaan 7'  
    }  
  }  
]
```

Retrieving the data from the route

```
// Fetching data send with the route:  
// Inject the route, subscribe to the data attribute.  
  
address: string;  
postcode:string;  
location:string;  
  
constructor(private route: ActivatedRoute) { }  
  
ngOnInit(): void {  
    this.route.data.subscribe((data:any)=>{  
        this.address = data.address;  
        this.postcode = data.postcode;  
        this.location = data.location;  
    })  
}
```


Questions?

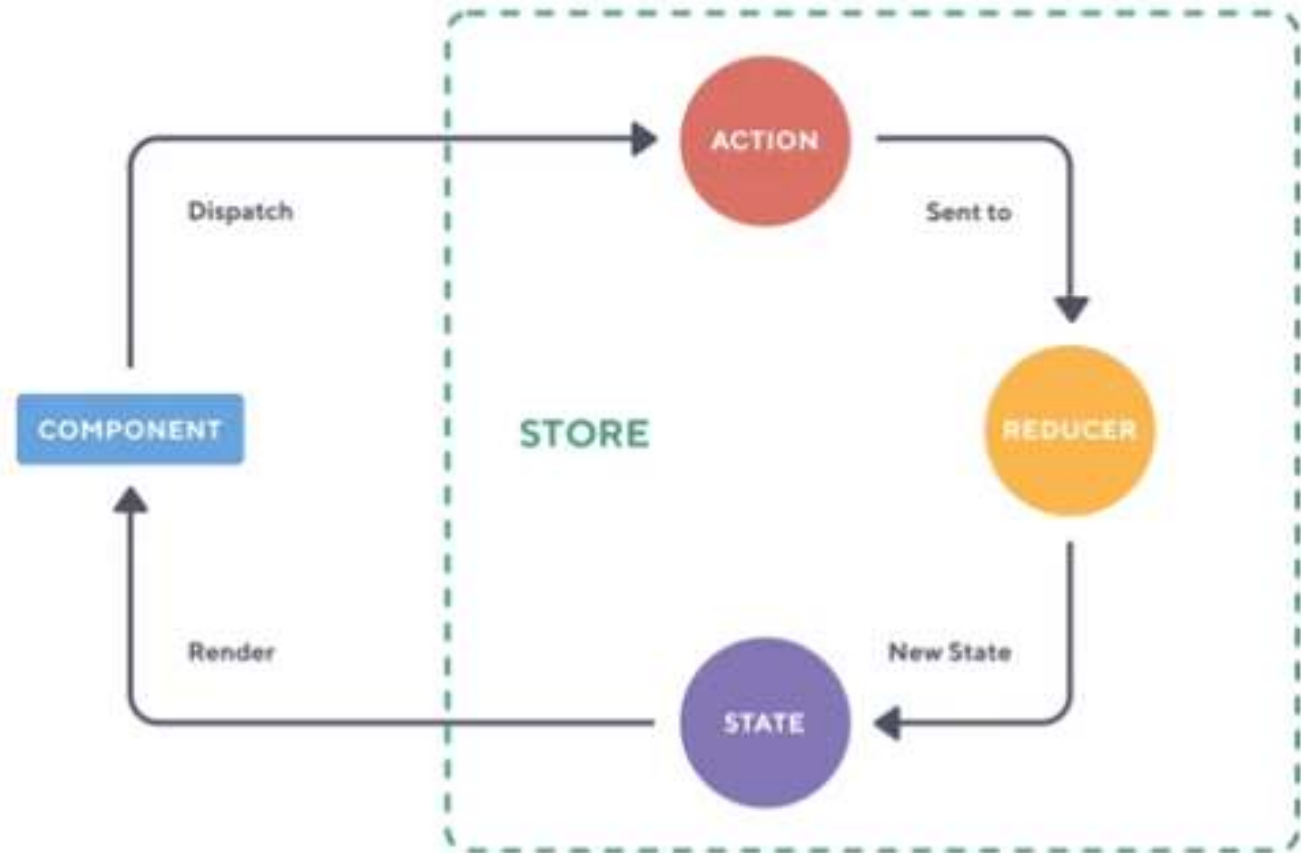
Agenda - 3 days - Thematic



- Day 1: Architecture
 - Composing Applications with multiple modules
 - ...
- Day 2: **Store & Observables**
 - Observable **Design pattern**
 - Creating **observables** from scratch
 - **State management** and @ngrx/store w/ observables
 - **Concepts**, State, Action, Reducer, Dispatcher, Effect, Http

REDUX ARCHITECTURE

One-way dataflow



<https://platform.ultimateangular.com/courses/ngrx-store-effects/lectures/3788532>

Next up

- Day 3: **Miscellaneous**
 - Dynamic Forms
 - Reusable/configurable Components, Services
 - Angular monorepo's / micro frontends?
 - Shared libraries in projects/multiple applications
 - Publishing Angular libraries to NPM
 - More on Angular Schematics
 - Your turn : Q & A, specific issues
 - ...



Angular Advanced, day #3



Vrijdag, 09:00 hr

Peter Kassenaar –
info@kassenaar.com

