



ANGULAR
ARCHITECTS
INSIDE KNOWLEDGE

Structure Angular applications with Modules

Alex Thalhammer

Goal

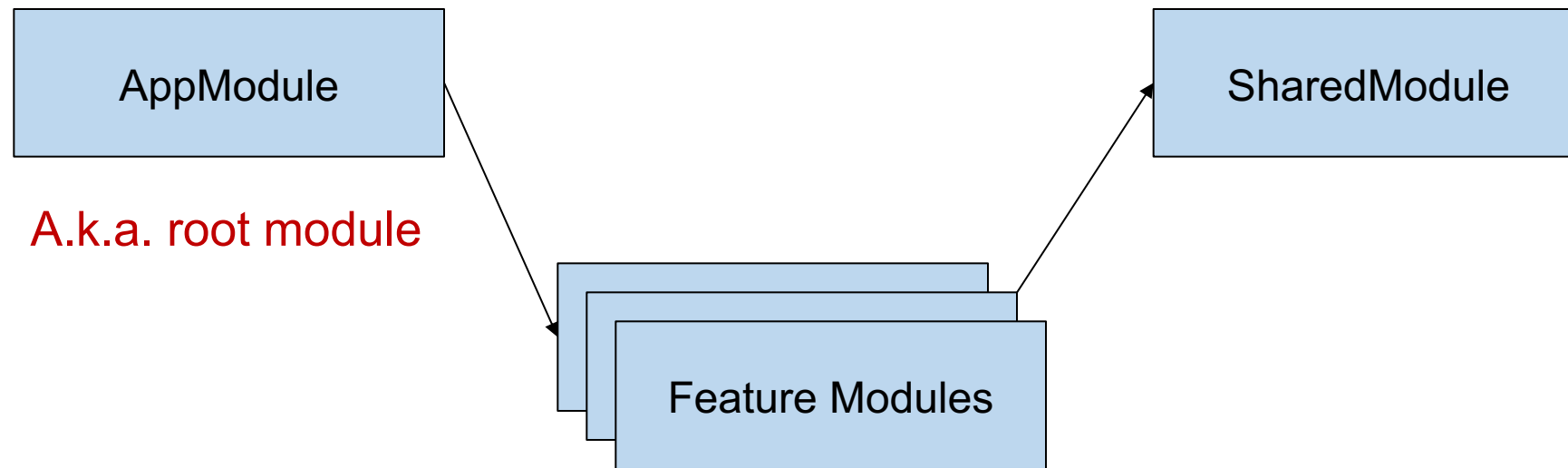
- Learn how to structure an Angular application with Modules
- Know how to use them in your projects

Current AppModule

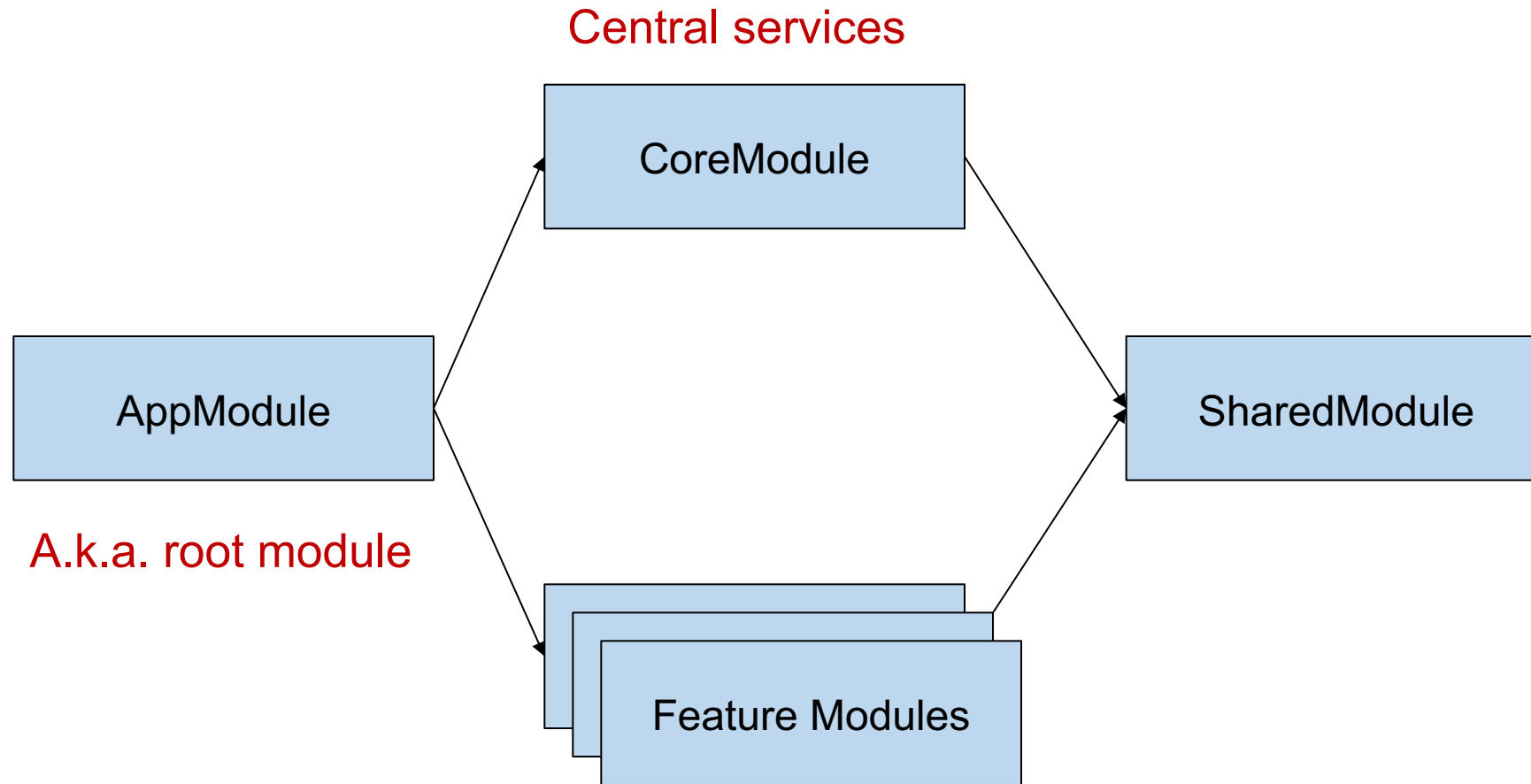
```
@NgModule({  
  imports: [  
    BrowserModule, HttpClientModule, FormsModule  
  ],  
  declarations: [  
    AppComponent,  
    FlightSearchComponent,  
    CityPipe  
  ],  
  bootstrap: [  
    AppComponent  
  ]  
})  
export class AppModule {}
```



Typical module structure



Optional: CoreModule



SharedModule

```
@NgModule({  
  imports: [  
    CommonModule  
  ],  
  declarations: [  
    CityPipe  
  ],  
  exports: [  
    CityPipe  
  ]  
})  
export class SharedModule {}
```

-----> declarations have to be exported to
be used in other Modules!



FlightBookingModule (Feature Module)

```
@NgModule({
  imports: [
    CommonModule, FormsModule, SharedModule
  ],
  declarations: [
    FlightSearchComponent
  ],
  providers: [
    FlightService
  ],
  exports: [
    FlightSearchComponent
  ]
})
export class FlightBookingModule {}
```

----- Provider for this module exclusively
(Angular <= 5 way of providing Services)

----- Everything we want to use in other modules!



AppModule (Root Module)


```
@NgModule({  
  imports: [  
    BrowserModule, HttpClientModule, FlightBookingModule  
  ],  
  declarations: [  
    AppComponent  
  ],  
  bootstrap: [  
    AppComponent  
  ]  
})  
export class AppModule {}
```



Re-exporting of modules

```
@NgModule({  
  imports: [  
    CommonModule, FormsModule  
  ],  
  declarations: [  
    CityPipe  
  ],  
  exports: [  
    CityPipe, FormsModule  
  ]  
})  
export class SharedModule {}
```

Use it here and in the modules that import this module

A diagram consisting of two red dashed arrows. The first arrow originates from the **FormsModule** entry in the `imports` array of the `@NgModule` decorator and points towards the explanatory text. The second arrow originates from the `SharedModule` class declaration and points towards the same explanatory text.

DEMO



LAB

