ARS

Motivation

- Daten abrufen, wenn diese benötigt werden
- Navigation in der Seite steuern (Bsp. Login)

ARS

Lernziele

- Sie benennen den Aufbau und die Bestandteile einer Resolve-Implementierung.
- Sie beschreiben wie mit Hilfe von Resolve Daten abgerufen werden können.
- Sie erklären, wie Guards verwendet werden, um die Routernavigation zu steuern.

ARS

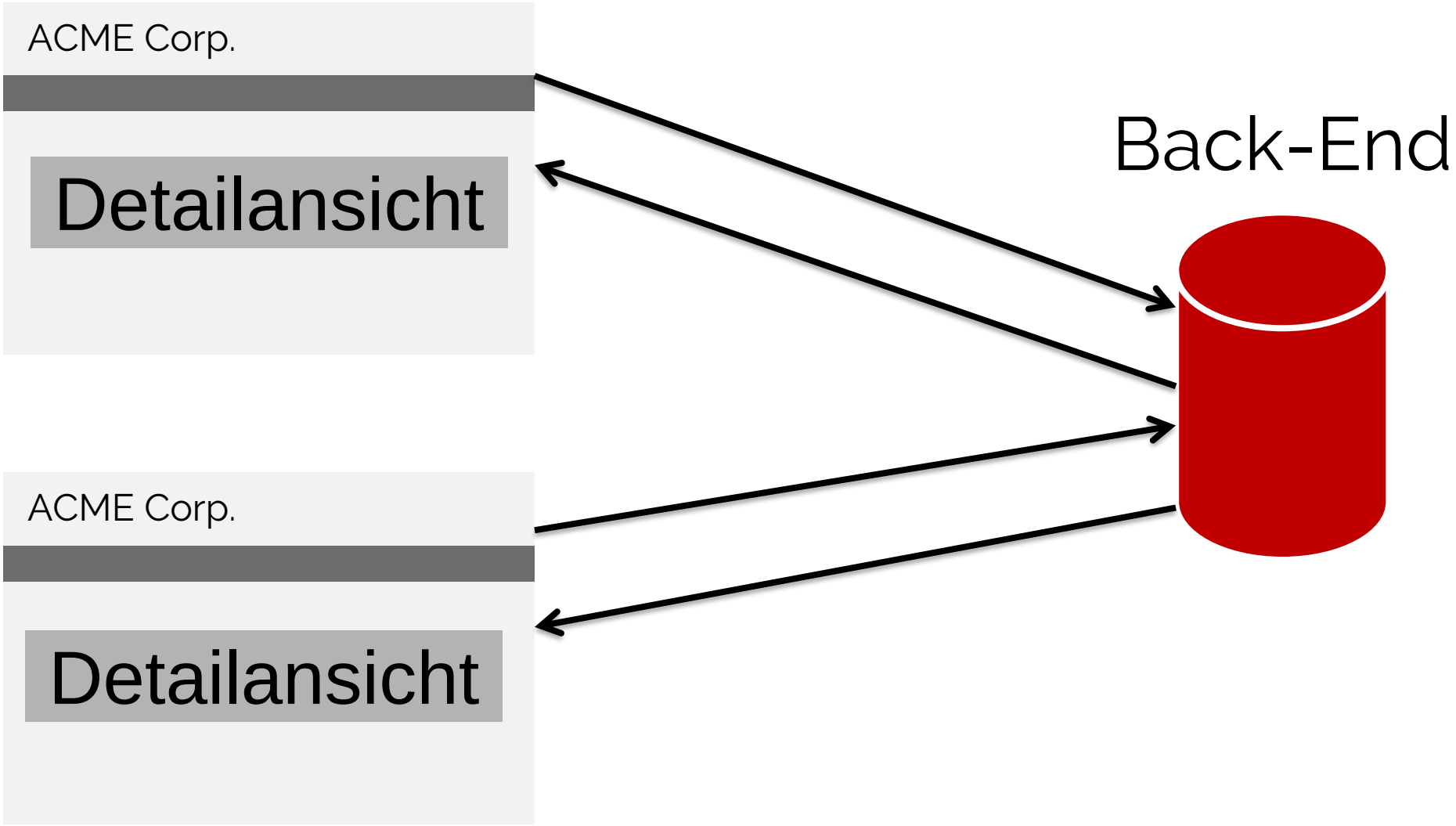
Resolve

Wie gehe ich mit Asynchronität um?

Wie verhindere ich, dass eine leere Seite angezeigt wird, bevor die Daten geladen wurden?

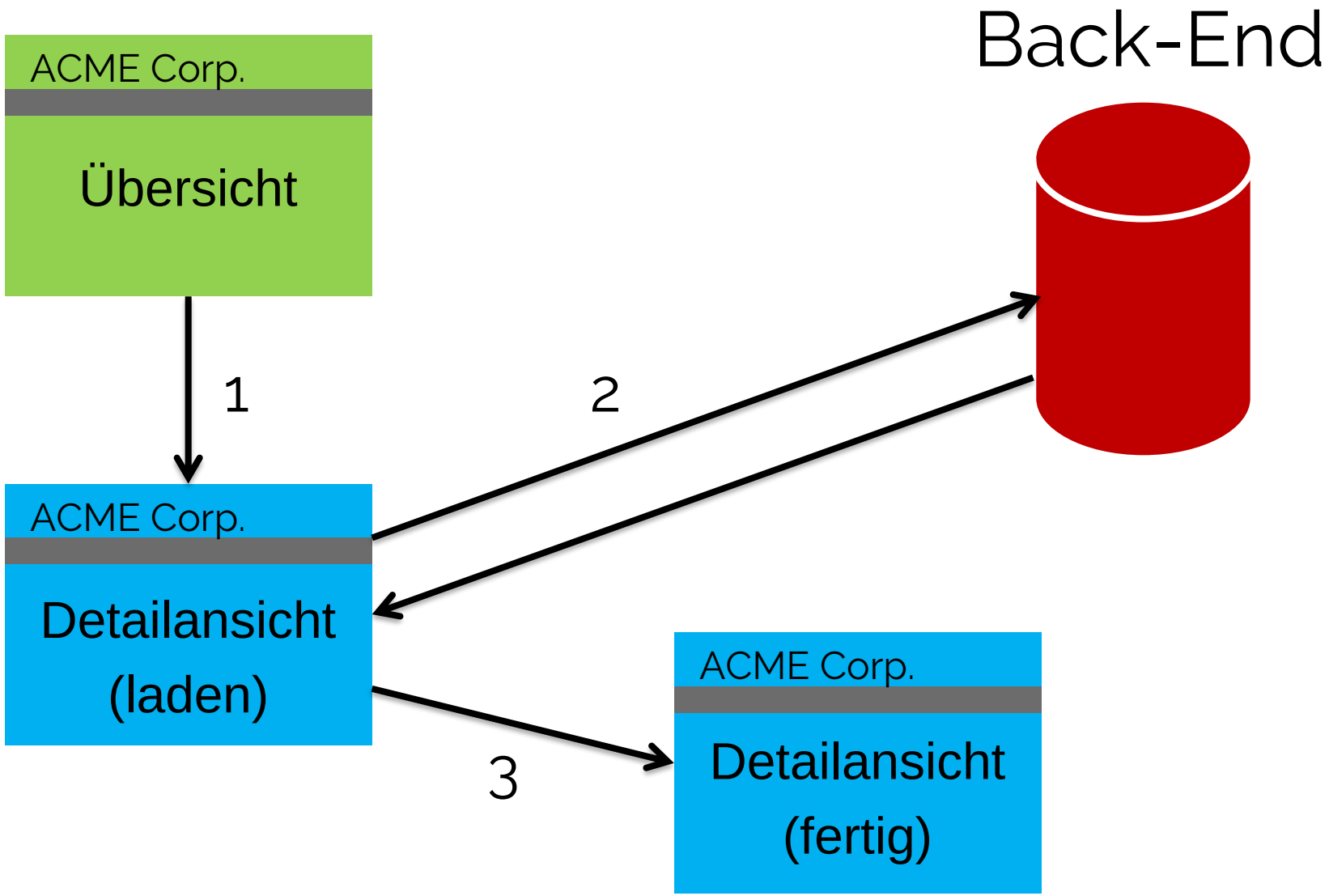
Was passiert, wenn Daten nicht existieren oder Fehler auftreten?

Resolve - Ablauf



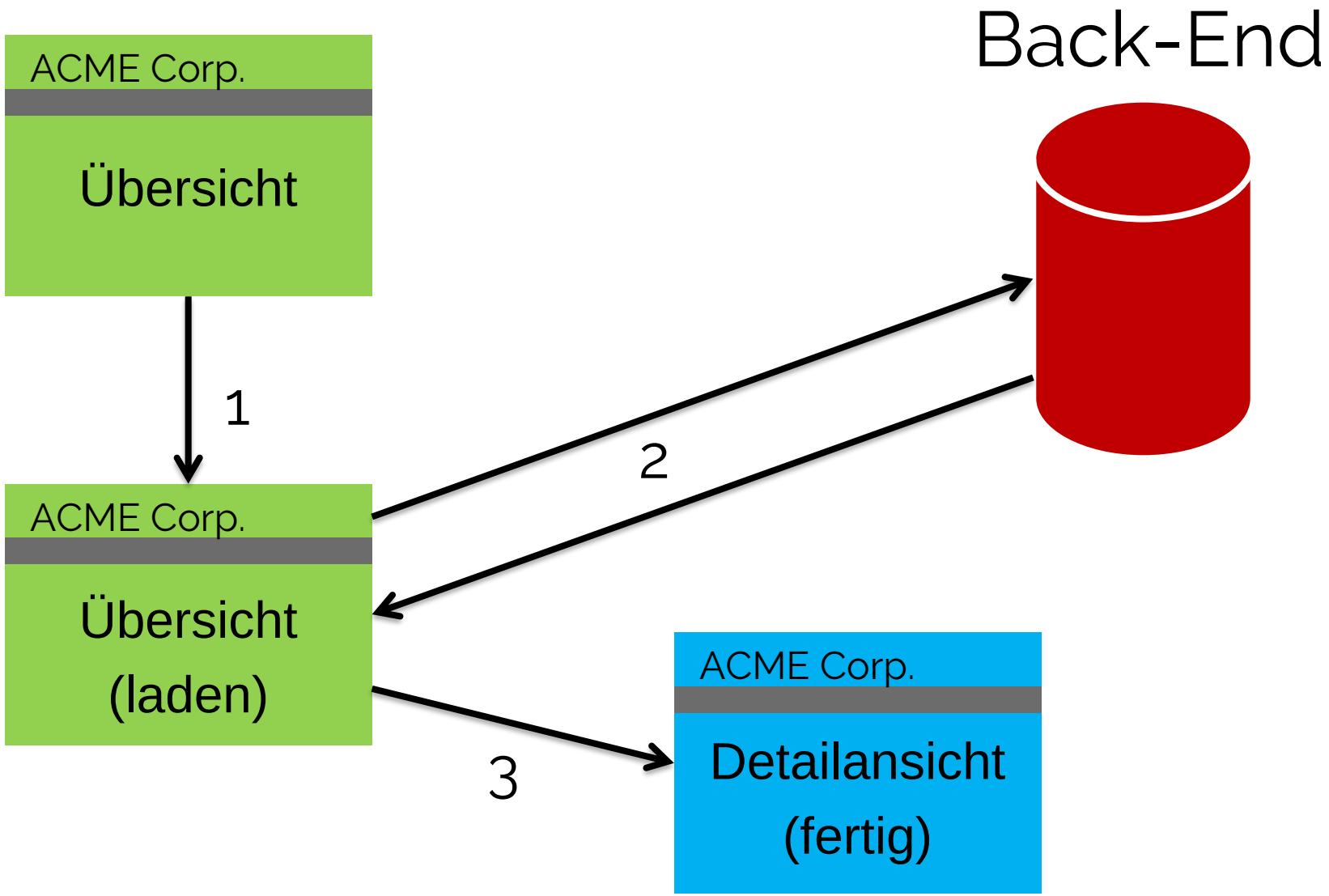
ARS

Resolve – Ablauf ohne Resolve



ARS

Resolve – Ablauf mit Resolve



ARS

Resolve - Definition

```
export interface Resolve<T> {
  resolve(
    route: ActivatedRouteSnapshot,
    state: RouterStateSnapshot
  ): Observable<T> | Promise<T> | T;
}
```

ARS

Resolve - Beispiel

```
@Injectable()
export class MyDataResolver implements Resolve<MyData> {
  constructor(private dataService: MyDataService) {}

  resolve(route: ActivatedRouteSnapshot) {
    const id = Number(route.paramMap.get('id'));

    return this.dataService.getById(id);
  }
}
```

ARS

Resolve – Routerkonfiguration - Routen

```
const routes: Routes = [
  {
    path: 'todos/:id',
    component: DetailComponent,
    resolve: {
       mydata: MyDataResolver
    }
  }
];
```

ARS

Resolve – Routerkonfiguration - Modul

```
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule],
  providers: [MyDataResolver]
})
export class AppRoutingModule {}
```

ARS

Resolve – Datenzugriff

```
@Component({ /* ... */ })
export class MyDataDetailComponent implements OnInit {
  constructor(private route: ActivatedRoute) {}

  ngOnInit() {
    this.route.data.subscribe(routeData => {
      const mydata = routeData.mydata;

      console.log(mydata);
    });
  }
}
```

ACME Corp.

Startseite

ACME Corp.

Admin-Seite

ohne Guard

mit Guard

eingeloggt

nicht eingeloggt

Bild: https://commons.wikimedia.org/wiki/File:Coldstream_Guard_July_06.jpg

ARS

Guards

Guards verhindern oder erlauben Routenänderungen!

CanActivate

Darf auf die Seite navigiert werden?

CanDeactivate

Darf die Seite verlassen werden?

Weitere Guards: **CanActivateChildren**, **CanLoad**

ARS

CanActivate - Definition

```
export interface CanActivate {
  canActivate(
    route: ActivatedRouteSnapshot,
    state: RouterStateSnapshot
  ): Observable<boolean> | Promise<boolean> | boolean;
}
```

ARS

CanActivate - Beispiel

```
/**
 * Verhindert die Navigation,
 * wenn der Nutzer nicht eingeloggt ist
 */
@Injectable()
export class AuthGuard implements CanActivate {
  constructor(private loginService: LoginService) {}

  canActivate(): boolean {
    return this.loginService.isLoggedIn();
  }
}
```

ARS

CanActivate – Routerkonfiguration - Routen

```
const routes: Routes = [{
  path: 'admin',
  component: AdminComponent,
  canActivate: [ AuthGuard ]
}];

@NgModule({
  imports: [ RouterModule.forRoot(routes) ],
  providers: [ AuthGuard ]
})
export class AppRoutingModule {}
```

ARS

CanDeactivate - Beispiel

```
/**
 * Verhindert die Navigation,
 * wenn Änderungen nicht gespeichert wurden
 */
@Injectable()
export class FormSavedGuard implements
CanDeactivate<MyFormComponent> {
  canDeactivate(component: MyFormComponent) {
    return component.didSave();
  }
}
```

ARS

Zusammenfassung

Um was ging es in diesem Modul?

- Resolve
- Guards
  - CanActivate
  - CanDeactivate

Wozu brauche ich das? Was will ich damit machen?

- Abrufen von Daten wenn diese benötigt werden
- Verhindern / Erlauben von Navigation

ARS

Kontrollfragen

- Wofür wird Resolve benötigt?
- Wie funktioniert der Zugriff auf die in Resolve geladenen Daten?
- Wie werden Guards eingesetzt?
- Welche Guards kennen Sie?