NGRX / STORE

Pavan Podila

Sapient

@pavanpodila

minions

PAVAN PODILA

@pavanpodila

blog.pixelingene.com

Sapient

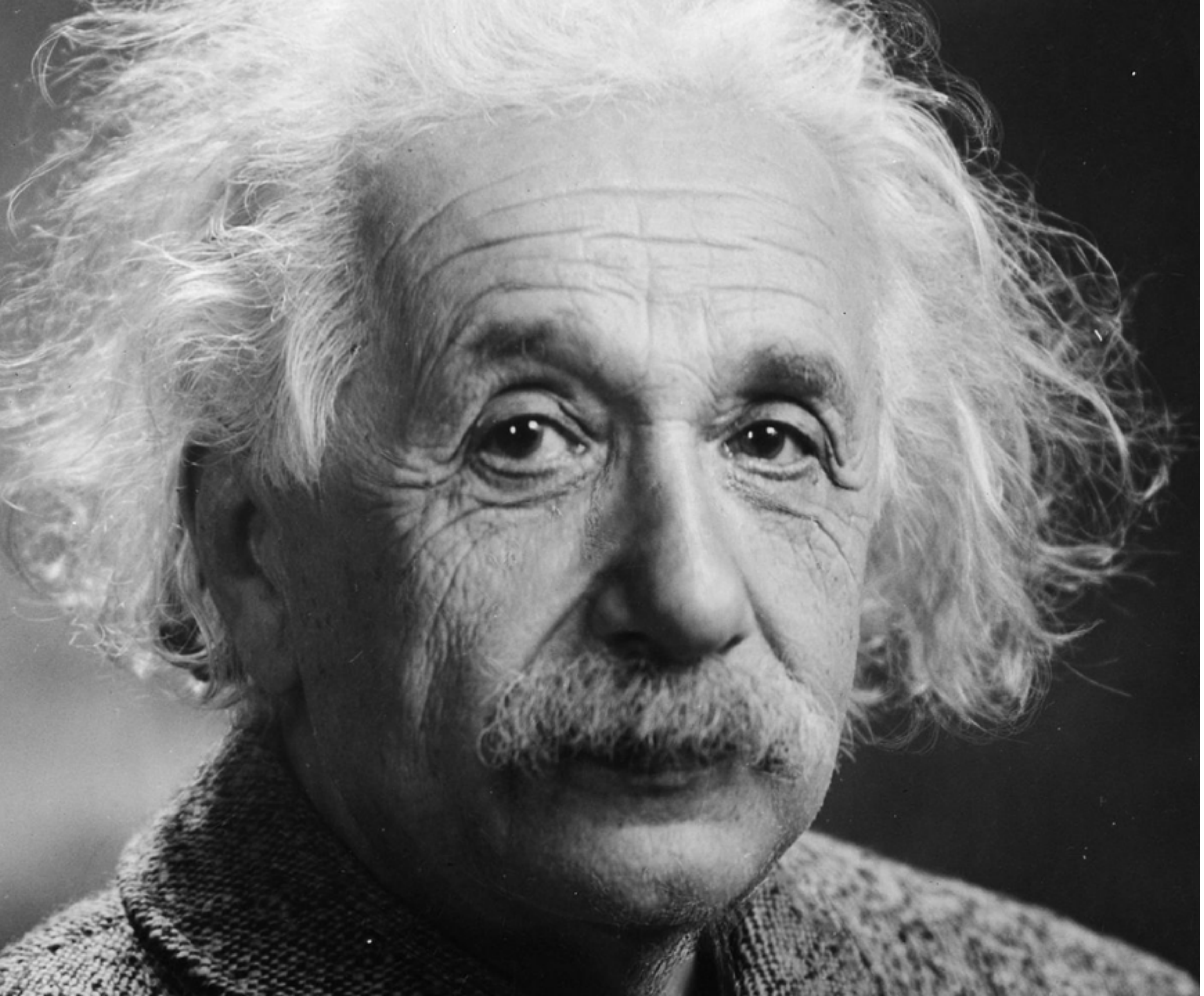
Director, Interactive Development

REDUCKS

World famous Duck Simulator

Redux

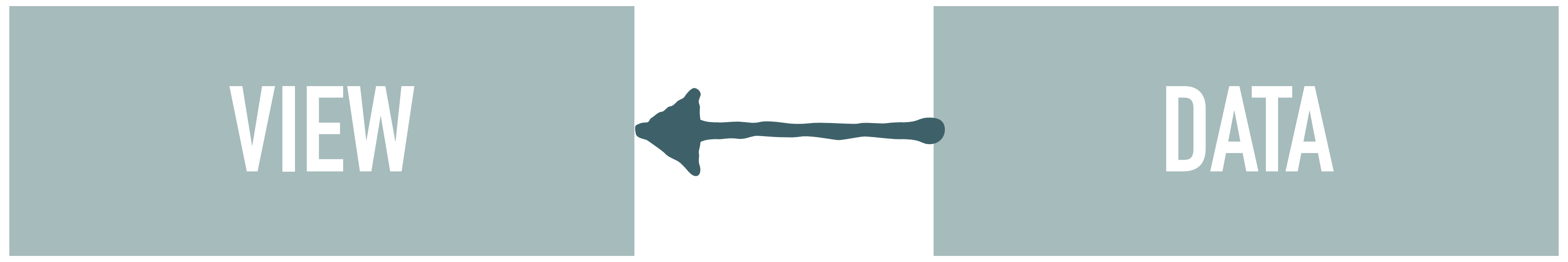
$$UI = fn(state)$$



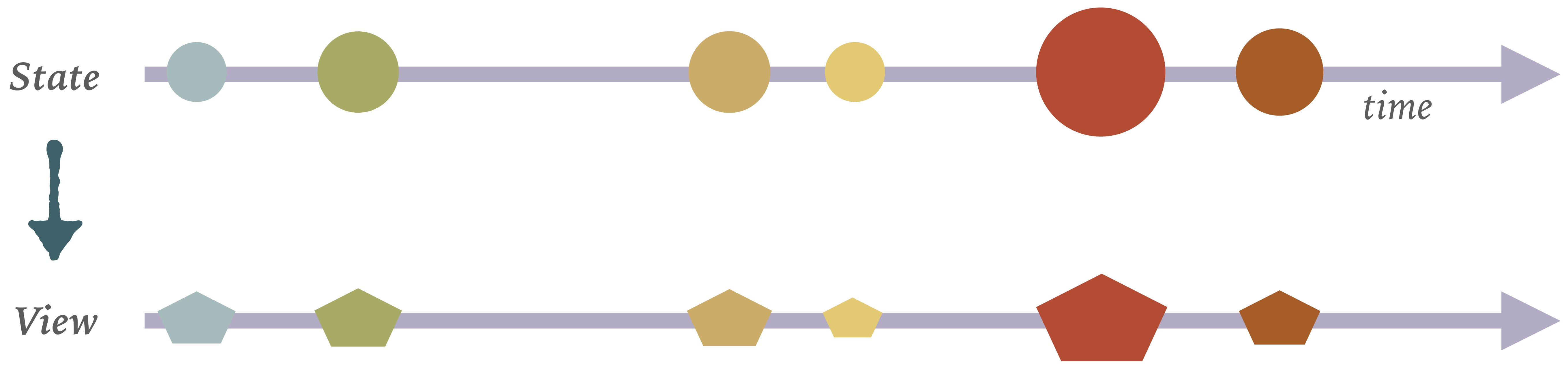
$E=MC^2$

$$UI = fn(state)$$

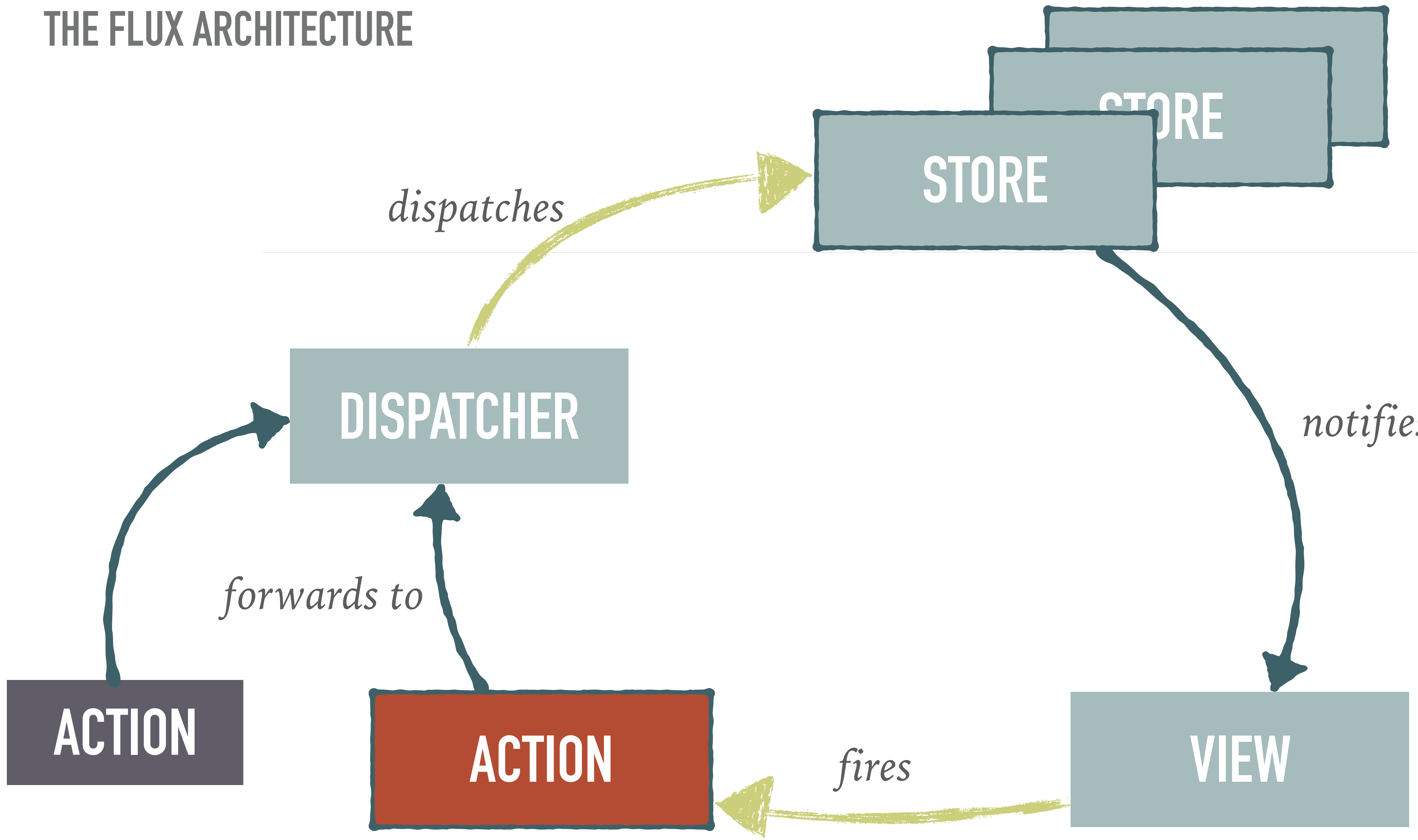
$$UI = fn(state)$$



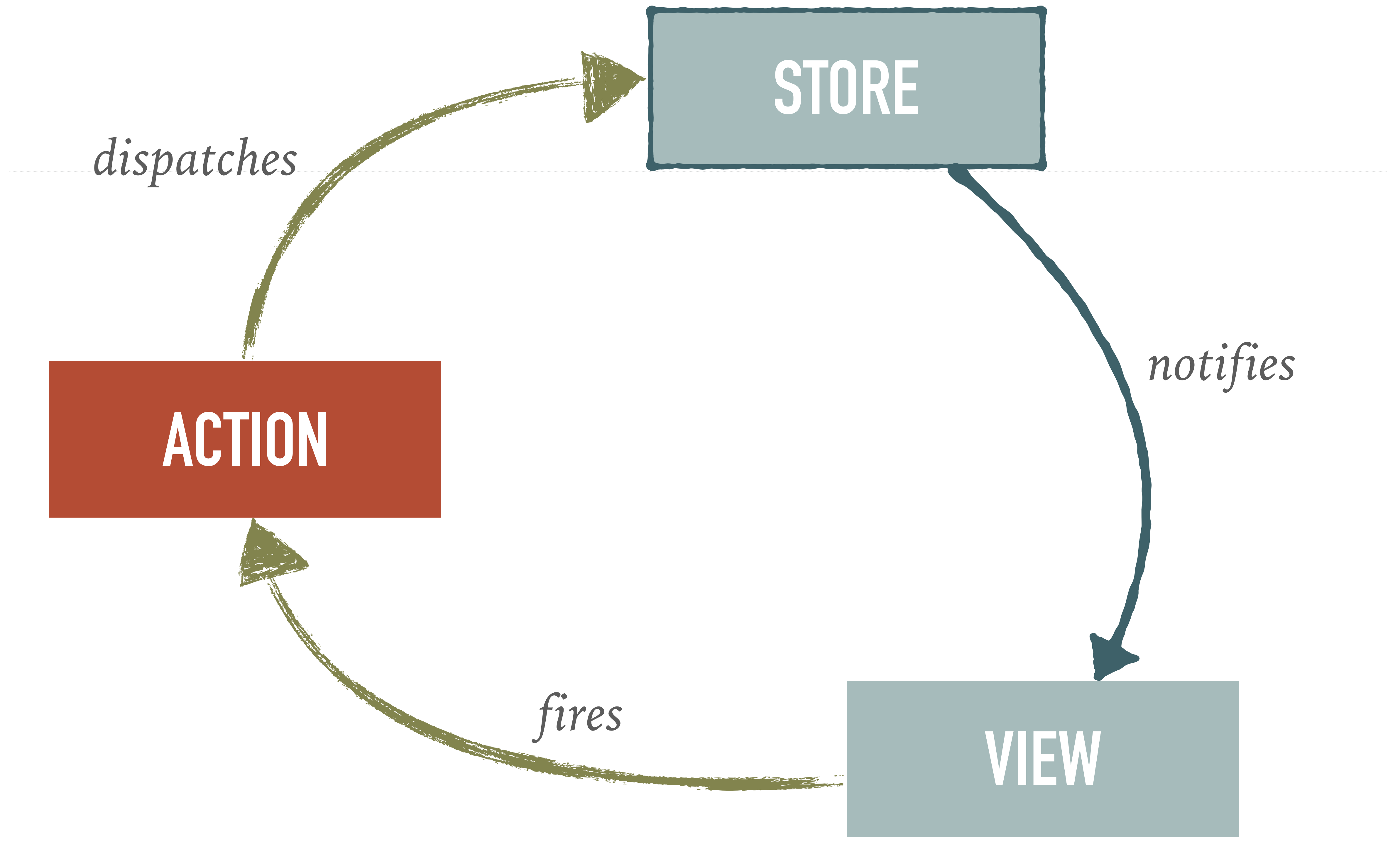
SNAPSHOTS



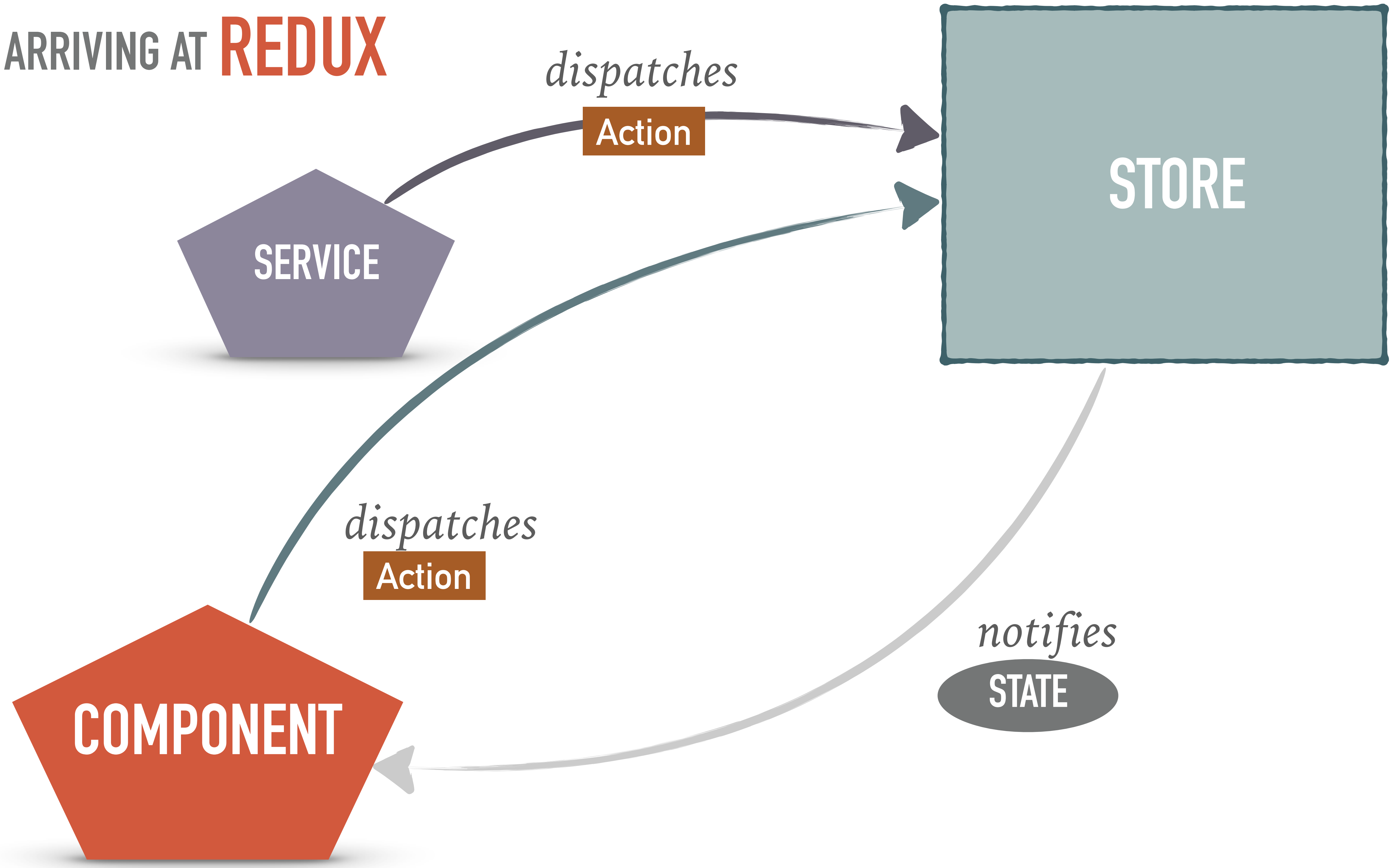
THE FLUX ARCHITECTURE



Simplifying THE FLUX ARCHITECTURE

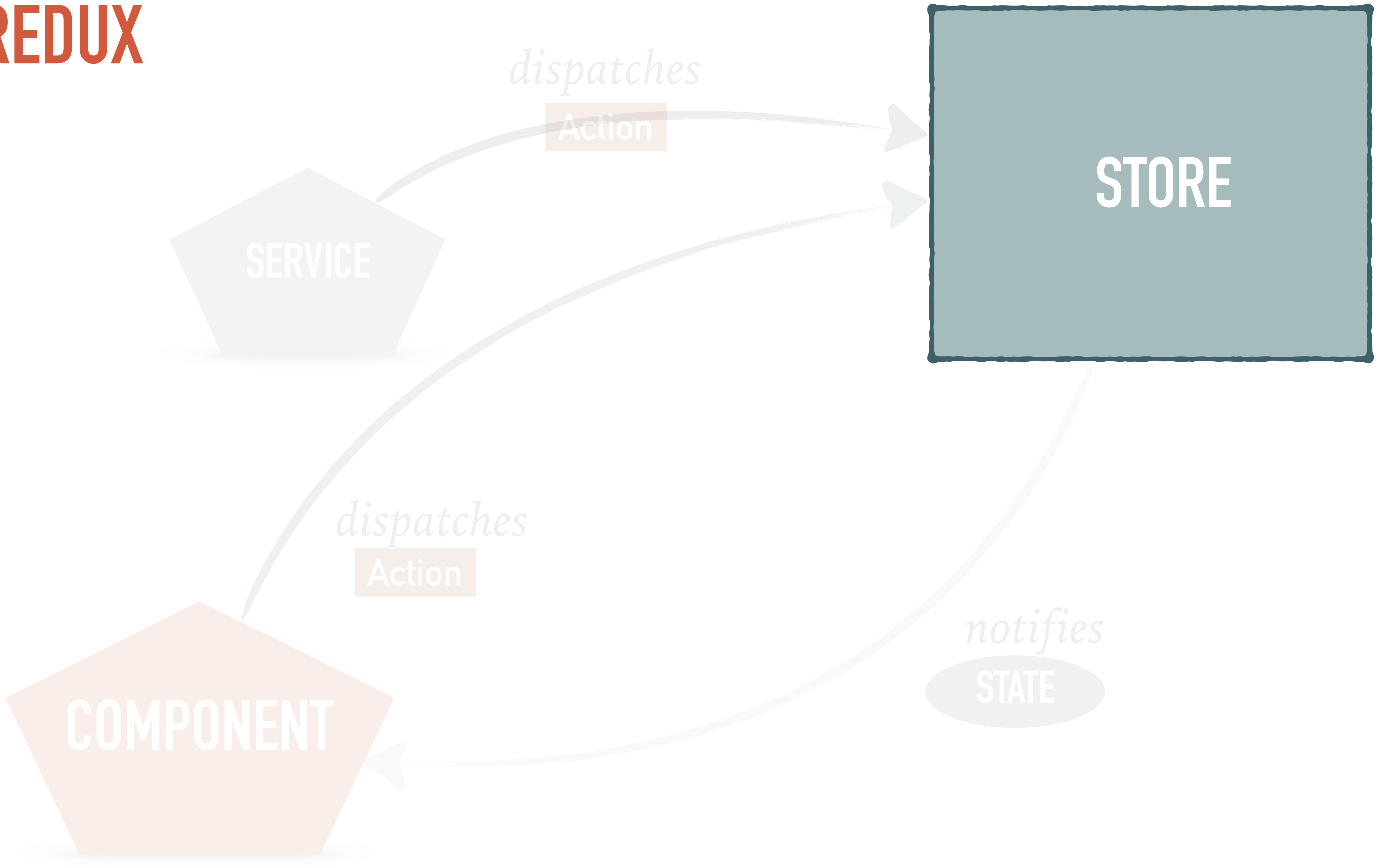


ARRIVING AT REDUX

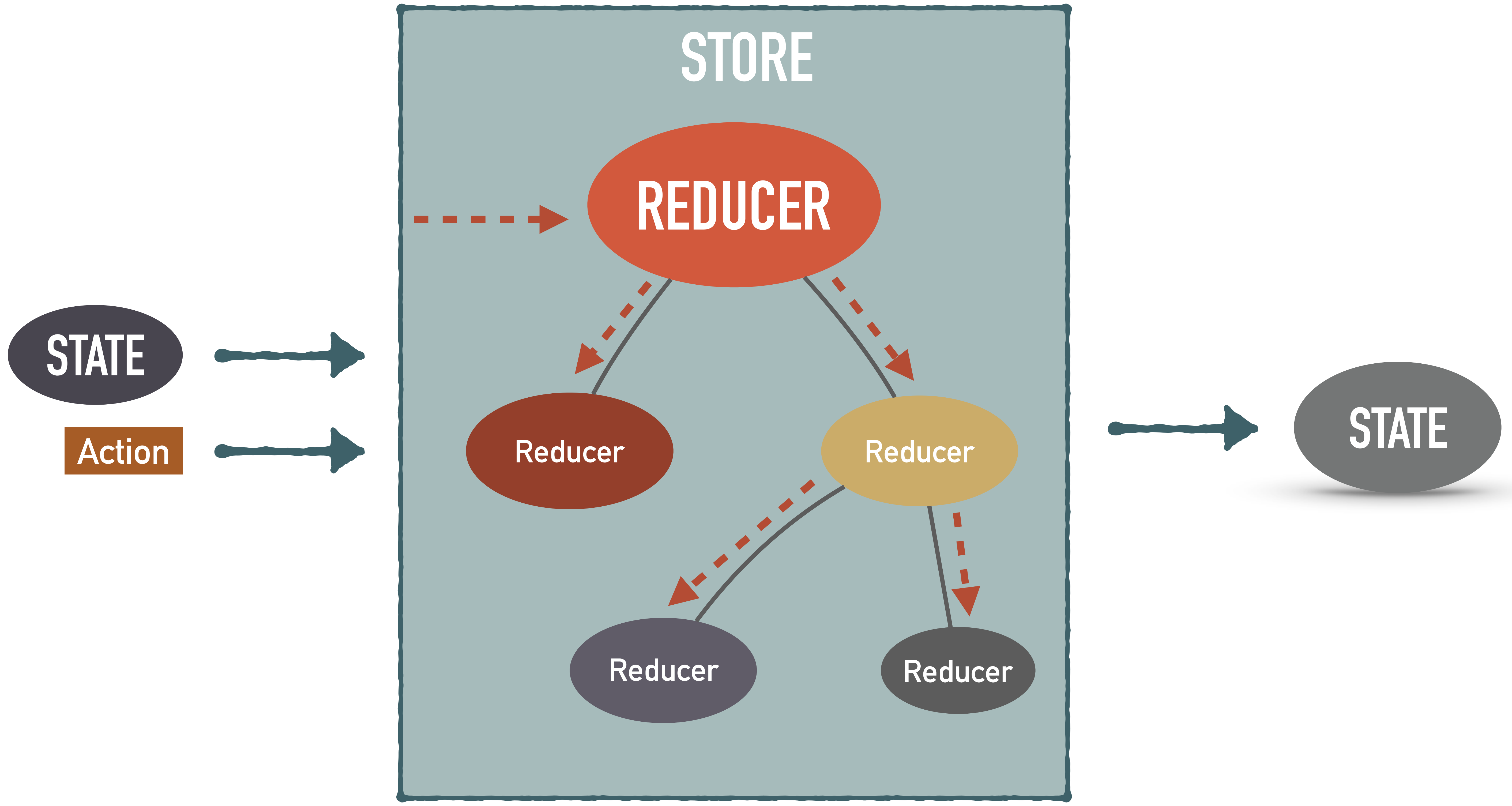


UI = fn(state)

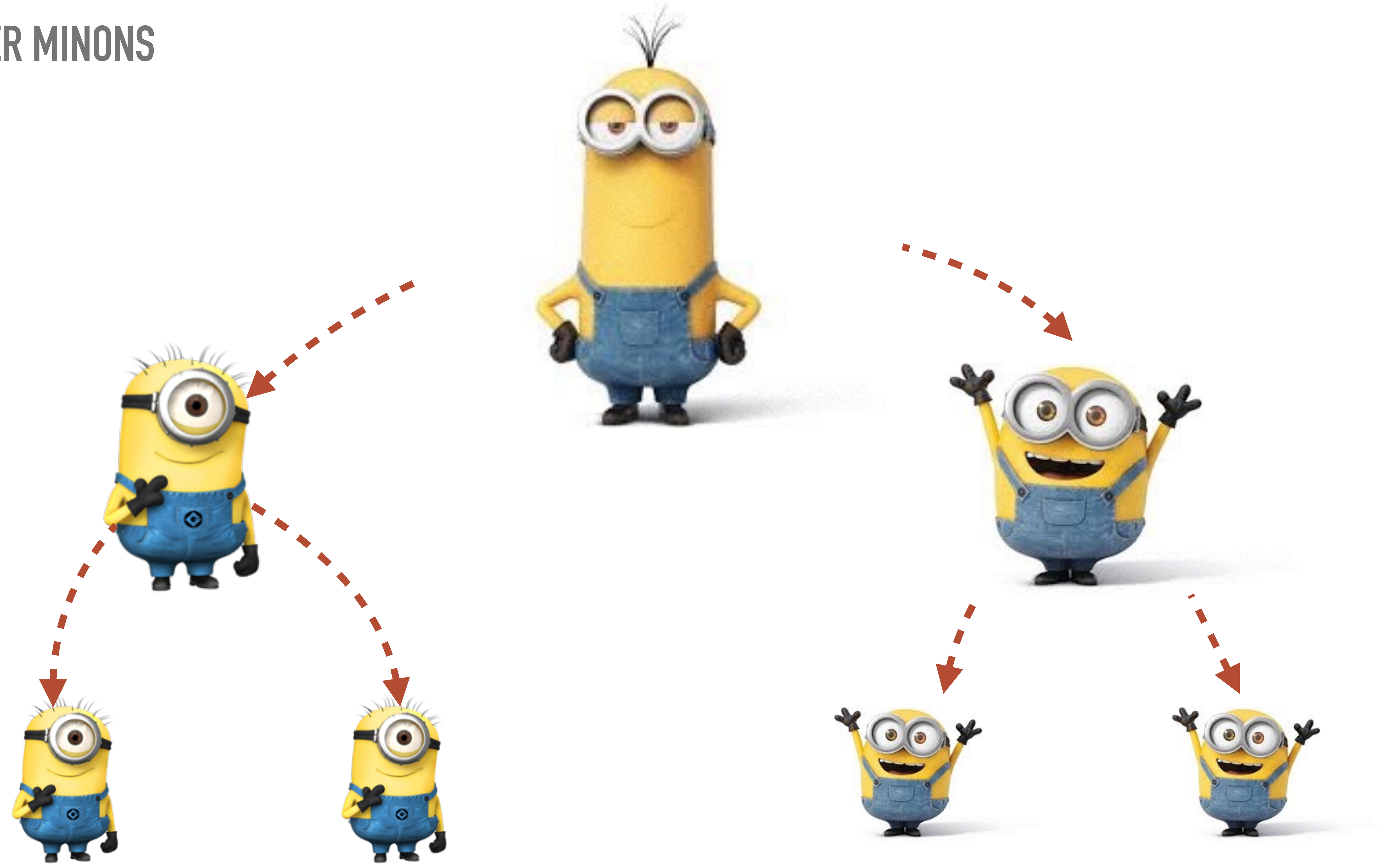
REDUX



UI = fn(state)

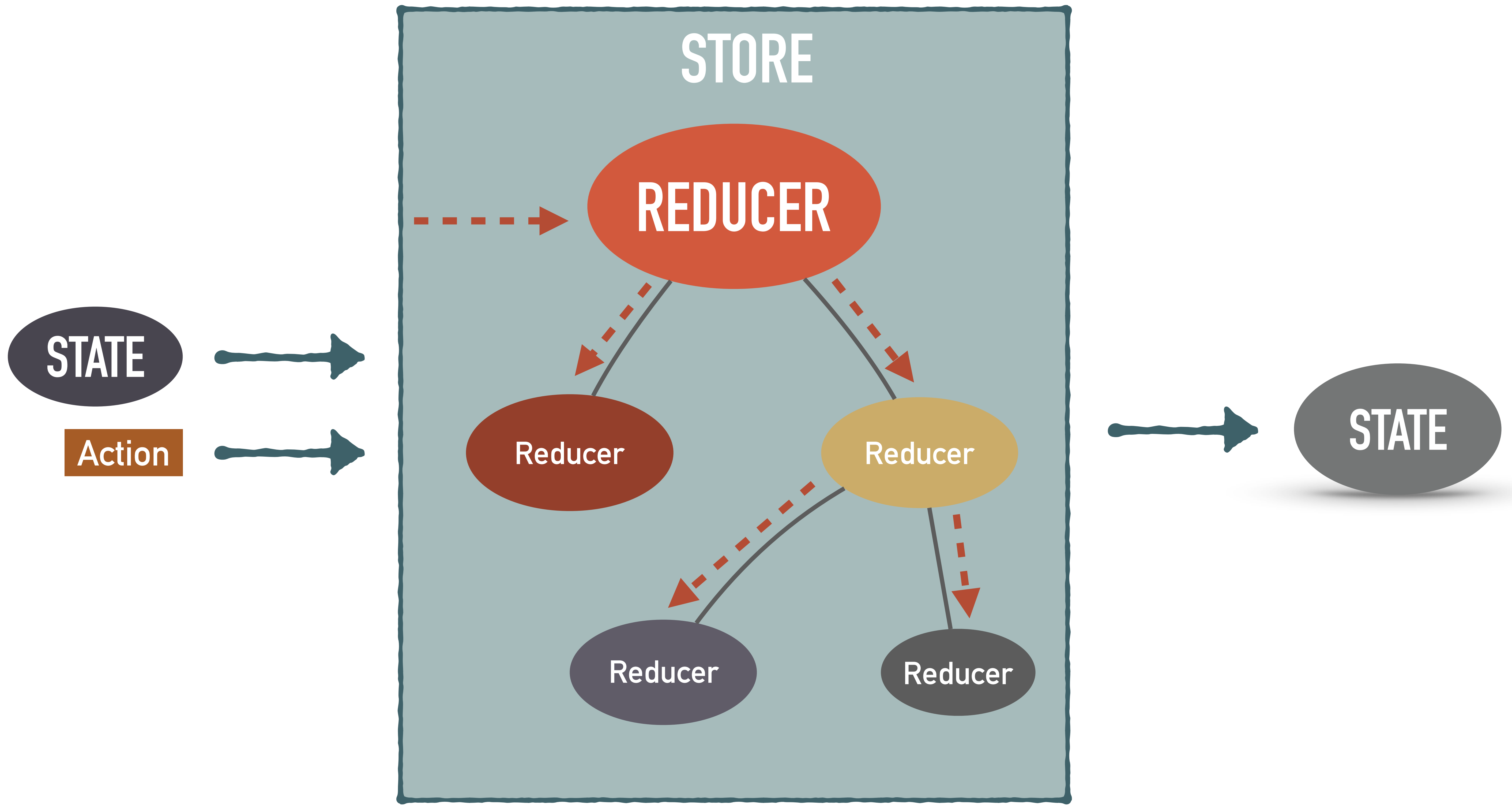

STORE
REDUCER
STATE
Action
Reducer
Reducer
Reducer
Reducer
STATE

REDUCER MINONS



Sapient

@pavanpodila


STATE
Action
STORE
REDUCER
Reducer
Reducer
Reducer
Reducer
STATE

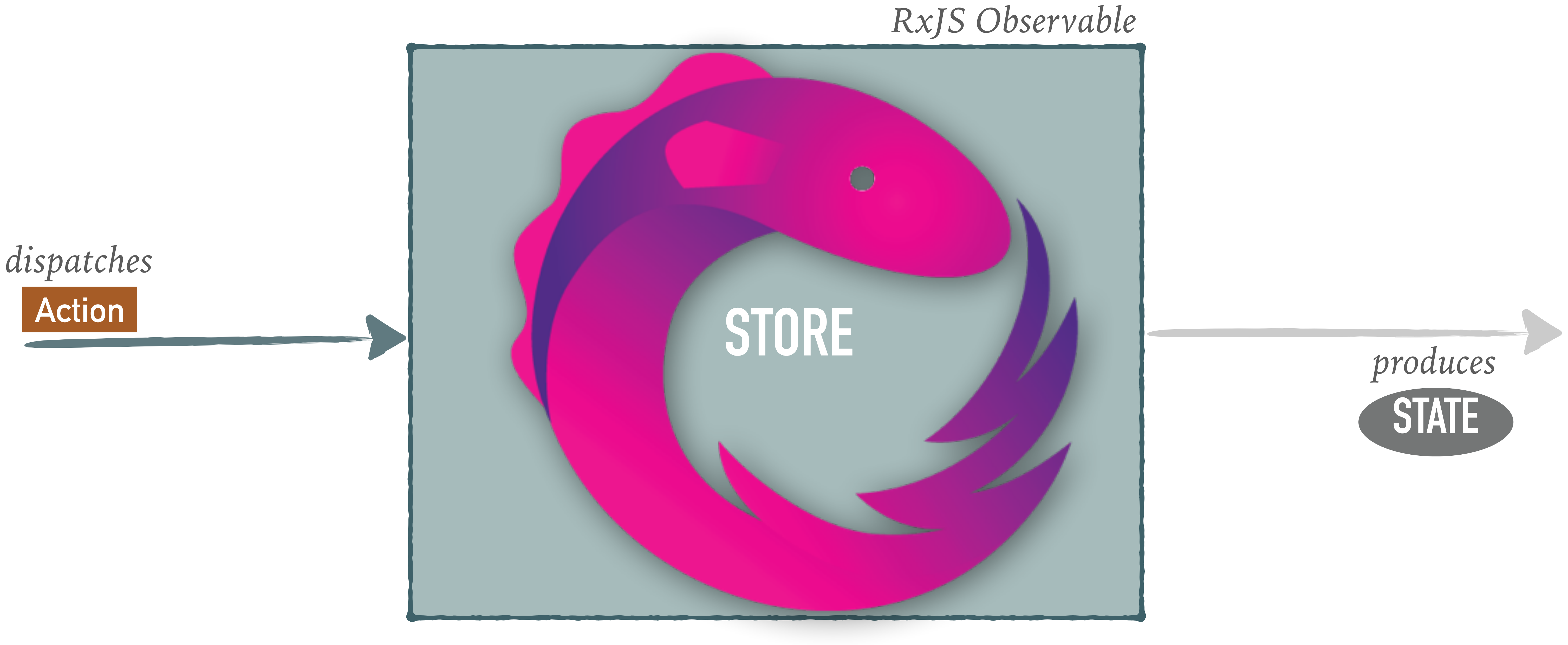
REDUCER

```
function reducer(state, action) {

    switch (action.type) {
        case START_ALBUMS_LOAD:
            return Object.assign({}, state, {loadingAlbums: true});

        case END_ALBUMS_LOAD:
            return Object.assign({}, state, {loadingAlbums: false});

        default:
            return state;
    }

}
```

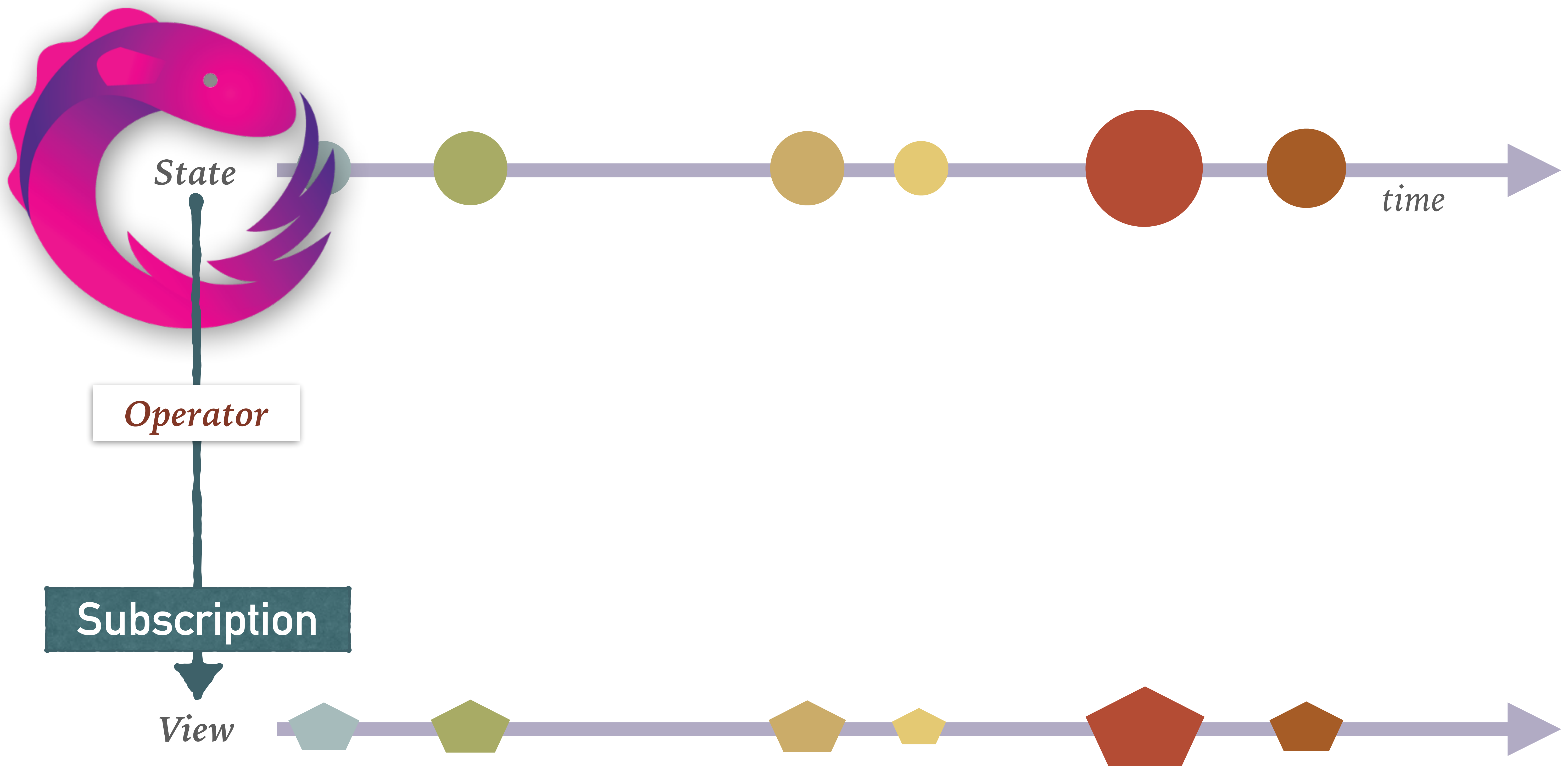
NGRX / STORE

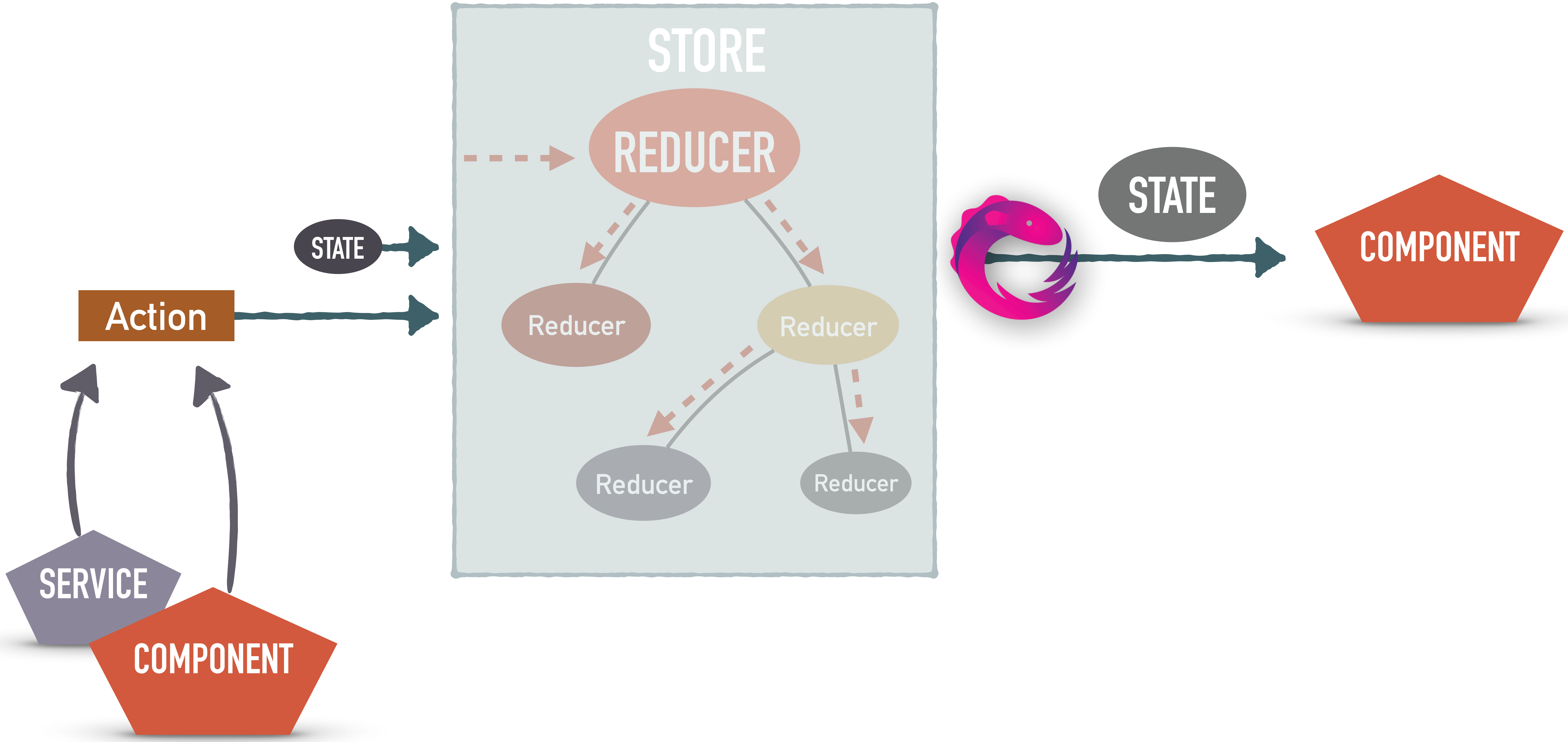
https://github.com/ngrx/store

NGRX / STORE



STREAM OF APPLICATION STATES





STORE
REDUCER
Reducer
Reducer
Reducer
Reducer
STATE
Action
SERVICE
COMPONENT
STATE
COMPONENT

CODE

Sapient

@pavanpodila

SUPPLYING STORE

```
@NgModule({
    imports: [
        RouterModule, FormsModule, BrowserModule,
        StoreModule.provideStore(reducer),
        routing
    ],
    providers: [PhotoService],
    declarations: [
        ShellComponent, DashboardComponent, UnknownComponent,
        AlbumComponent, PhotoComponent, AlbumDetail
    ],
    bootstrap: [ShellComponent]
})
export class AppModule {
}
```

INJECTING STORE

```
@Component({
    selector: 'photo',
    template: require('./photo.html')
})
export class PhotoComponent implements OnInit, OnDestroy {

    constructor(private route: ActivatedRoute,
                private store: Store<AppState>,
                private service: PhotoService) {
    }
}
```

Sapient

@pavanpodila

DISPATCHING ACTIONS

```
@Injectable()
export class PhotoService {

    constructor(private store: Store<AppState>) {
    }

    /* ... */

    loadAlbums() {
        this.store.dispatch({
            type: START_ALBUMS_LOAD
        });
    }
}

```

Sapient

@pavanpodila

Container

SLICING STATE IN COMPONENTS

```
this.subscription = this.store.select(x => x.albums.selectedPhoto)
    .subscribe((photo: Photo) => {
        this.photo = photo;
    });

this.loadSubscription = this.store.select(x => x.operation.loadingPhoto)
    .subscribe(flag => {
        this.loading = flag;
    });
```

DEMO

RXJS *https://www.learnrxjs.io*

NGRX / STORE *https://github.com/ngrx/store*

REDUX-DEVTOOLS *https://github.com/zalmoxisus/redux-devtools-extension*

DEMO *https://github.com/pavanpodila/angular-nyc-ngrx-store*

PAVAN PODILA

@pavanpodila

blog.pixelingene.com