

The slide features several decorative red lines and dots. A large, thin red arc curves from the left edge towards the top left. Another large red arc curves from the top right towards the right edge. A smaller red arc is positioned at the bottom right. In the center, there is a complex red line pattern consisting of several loops and dots, resembling a stylized 'S' or a calligraphic flourish. The main title is centered in a large, bold, dark grey font.

State-driven Design mit Redux

Alex Kott
Frontend-Freunde @ Zweitag
17. Januar 2018



Konzept

Konzept





Konzept

Arten von State im Frontend

Location

```
/movies/green-book#summary
```

Control

```
hideWatched: true  
search:      'gre'
```

Data

```
movies: [  
  { title: 'Green Book' },  
  { title: 'Star Wars' },  
]
```

Communication

```
isLoadingMovies: true  
isLoadingUsers:  false
```

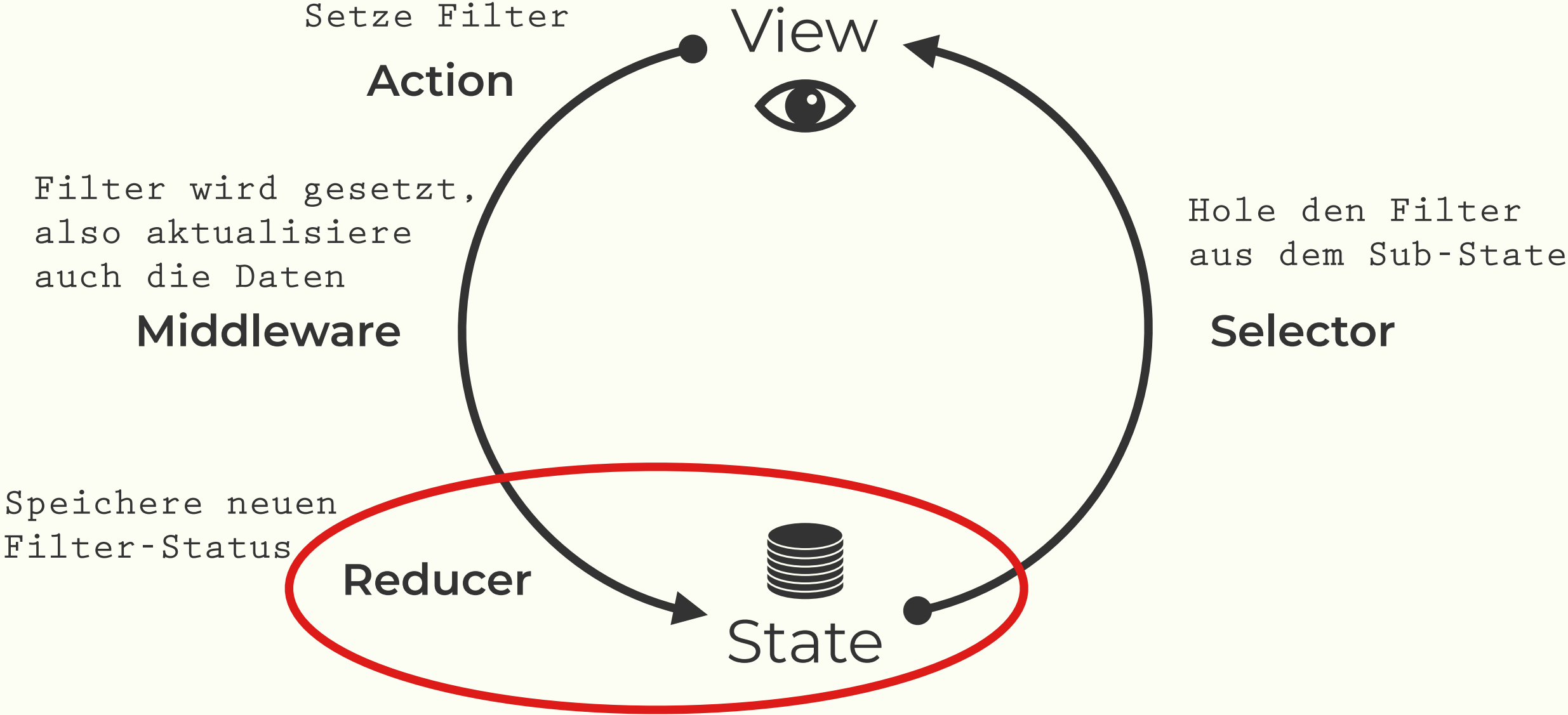
Session

```
isLoggedIn: true  
userName:   alex
```



Konzept

Redux Architektur





Konzept

Setup

Setup





Konzept

Setup

Reducers

- reducers
 - data
 - JS index.js
 - JS movies.js
 - filter
 - JS index.js
 - JS search.js
 - session
 - JS index.js
 - JS user.js
 - ui
 - JS index.js
 - JS loading.js
 - JS index.js



```
{
  location: {
    pathname: '/'
  },
  data: {
    movies: []
  },
  filter: {
    search: ''
  },
  session: {
    user: false
  },
  ui: {
    loading: 0
  }
}
```



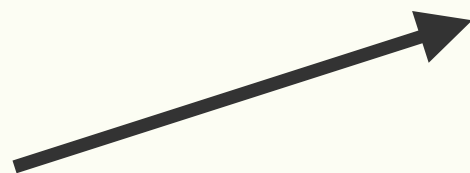
Konzept

Setup

View

```
{
  location: {
    pathname: '/'
  },

  data: {
    movies: []
  },
  filter: {
    search: ''
  },
  session: {
    user: false
  },
  ui: {
    loading: 0
  }
}
```



```
const mapStateToProps = state => ({
  movies: selectors.getMovies(state),
});

const MovieList = ({ movies }) => (
  <ul>
    {movies.map(movie =>
      <li>{movie.title}</li>
    )}
  </ul>
);

export default
  connect(mapStateToProps)(MovieList);
```





Konzept

Setup

Interaktion

Interaktion



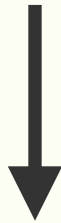


Konzept

Setup

Interaktion

Action



Reducer



Selector

```
const setFilter = value => ({  
  type: 'SET_FILTER',  
  payload: { value },  
});
```

```
const reducer = (previousState, action) => {  
  if (action.type === 'SET_FILTER') {  
    return action.payload.value;  
  }  
  return previousState;  
}
```

```
const getFilterValue = state => state.filter.search;
```



Konzept

Setup

Interaktion

```
const mapStateToProps = state => ({
  value: selectors.getFilterValue(state),
});

const mapDispatchToProps = dispatch => ({
  onChange: input => dispatch(actions.setFilter(input)),
});

const Filter = ({ value, onChange }) => (
  <input
    onChange={event => onChange(event.target.value)}
    value={value}
  />
);

export default connect(mapStateToProps, mapDispatchToProps)(Filter);
```



Konzept

Setup

Interaktion

Optimierung

Optimierung





Konzept

Setup

Interaktion

Optimierung

Action



Request Middleware

Wartet...



Fetch Data



Reducer



Konzept

Setup

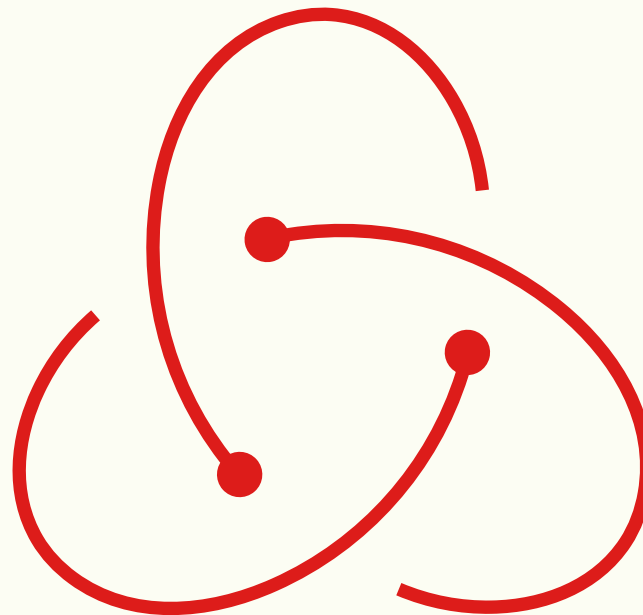
Interaktion

Optimierung

zwen

Zweitag Generator

Zwenerator



redux.js.org

github.com/faceyspacey/redux-first-router

redux-form.com

egghead.io/courses/getting-started-with-redux

medium.com/@dan_abramov/you-might-not-need-redux-be46360cf367