

REDUX

a predictable state container for JavaScript apps

BY DAN JIPA, TEAM LEADER SOFTVISION

The need

Managing state in big applications is too complicated.

There is no separation of concerns.

Simplified debugging.

! Single Source of Truth

Traditional communication between components

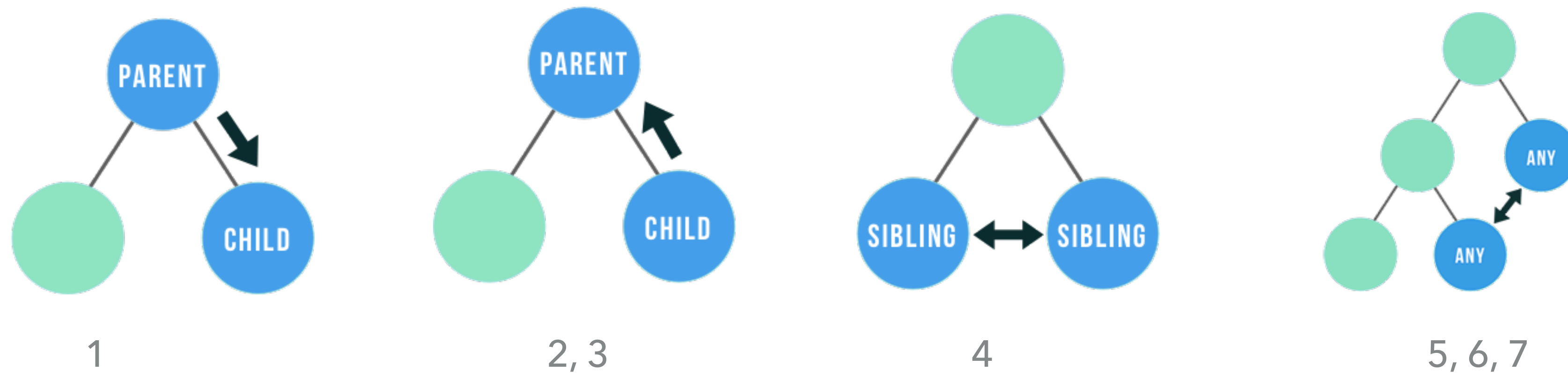


Image: <http://andrewfarmer.com/component-communication/>

1. Props

2. Callback functions

3. Event bubbling

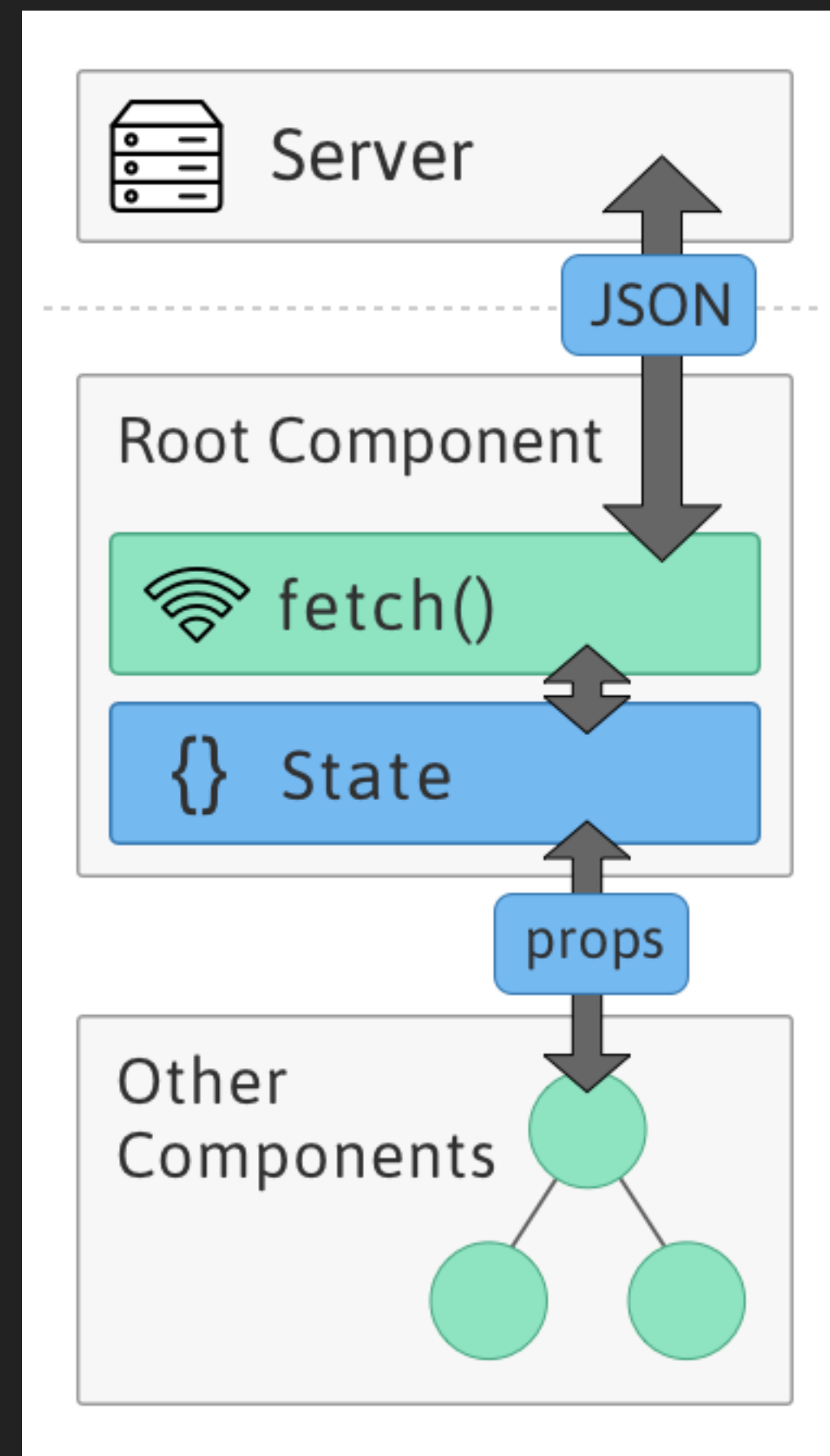
4. Parent component

5. Observer pattern (publish,subscribe)

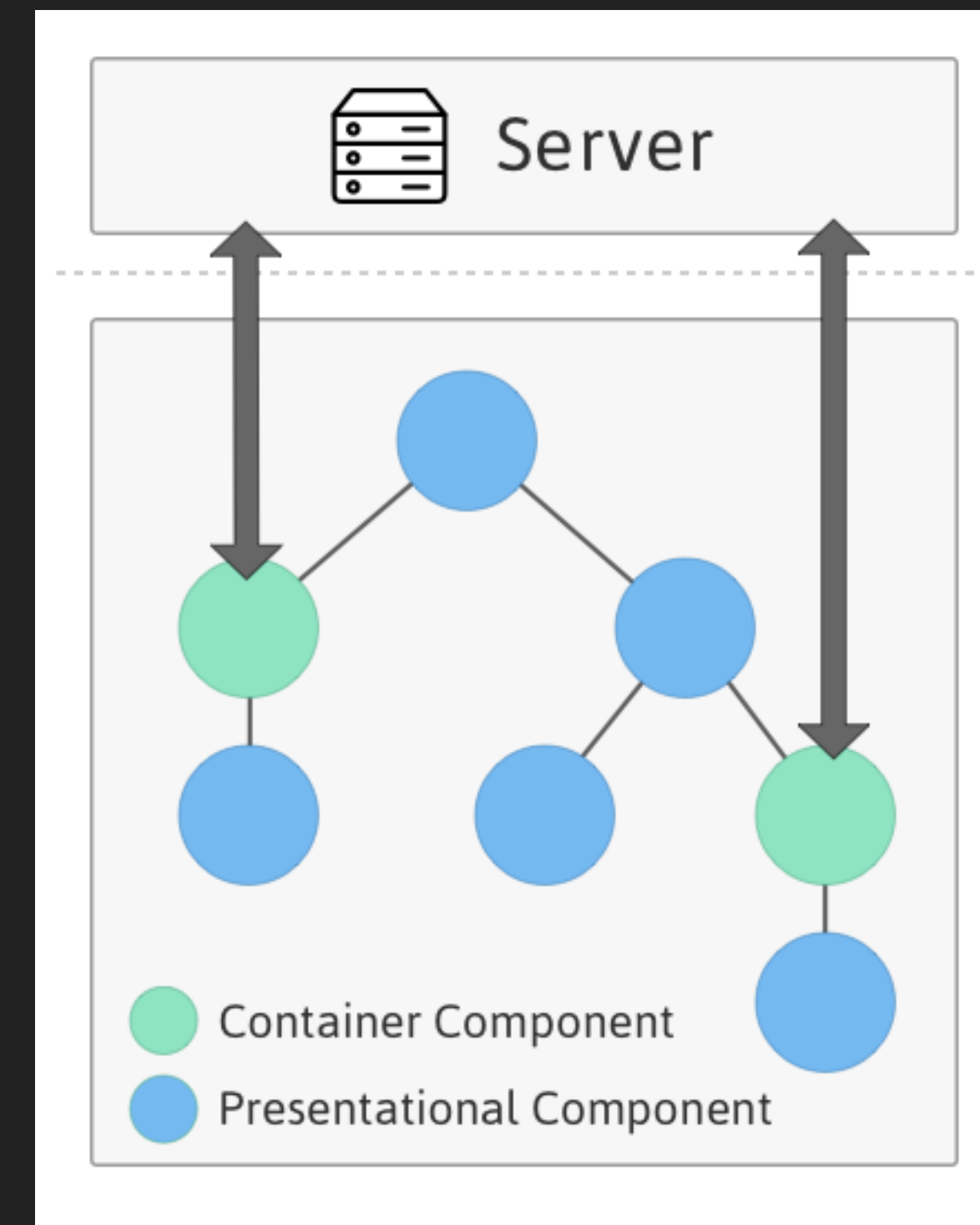
6. Global vars

7. Context

Traditional data fetching



1. Root component



2. Container component

What do we need to start using Redux with React?

- ▶ *Webpack (optional)*
- ▶ *Babel (optional)*
- ▶ **Redux**
- ▶ **React - Redux**
- ▶ *Middleware (Redux Thunk)*
- ▶ *Request library (Fetch, Axios etc)*

Redux in a few words

Store = {global application state}; dispatch()

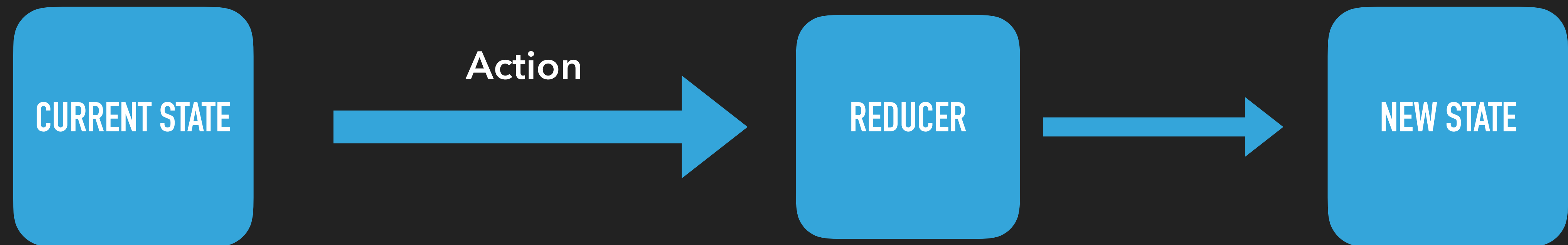
Actions <User, Browser, Server> = the only way to change state

Reducers = mapping actions in state updates

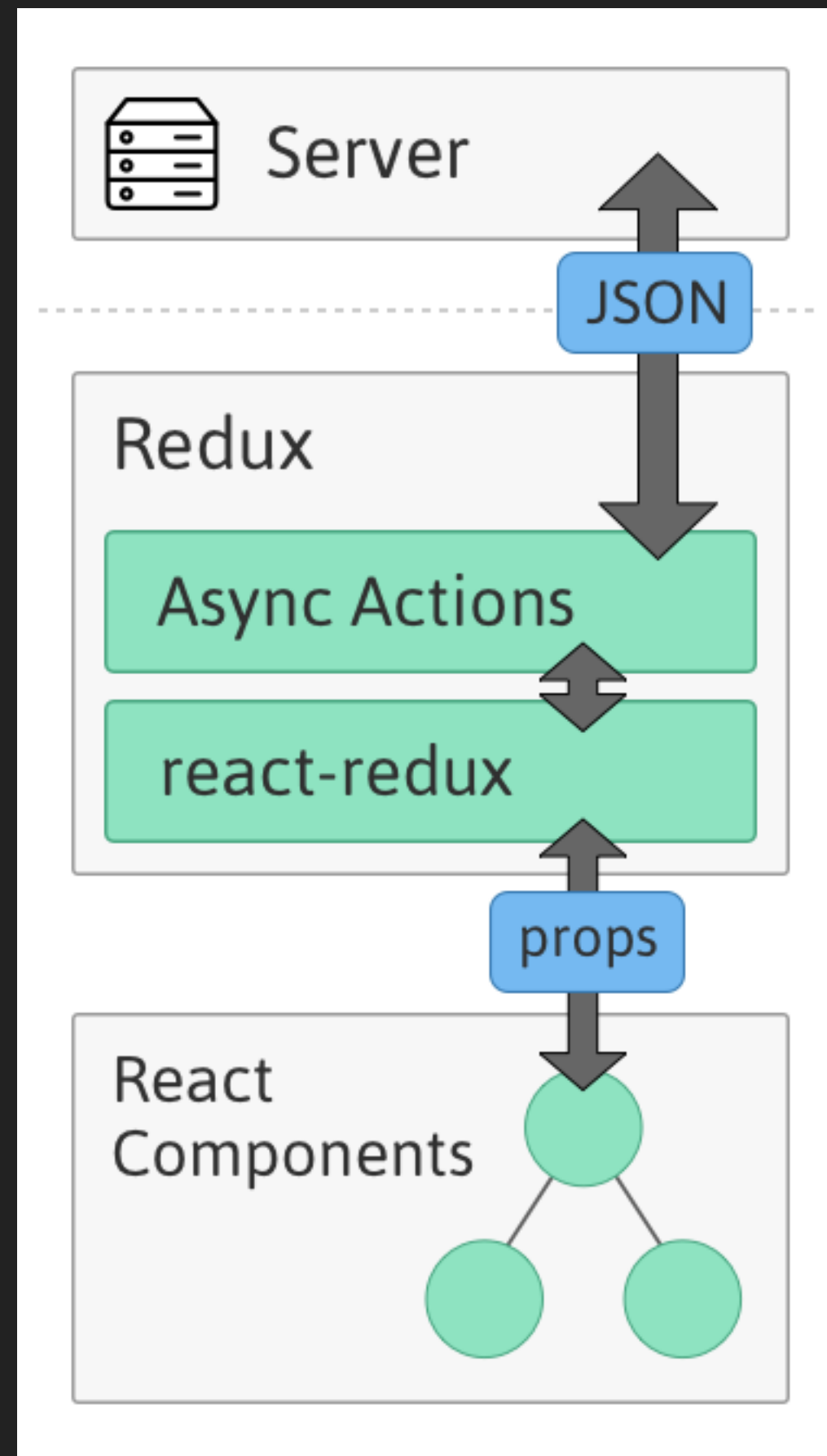
connect() = sending data to low level components via props

high level/container components = components with access to global store

Updating our state



Fetching data with Redux



Creating our initial store

```
import { createStore, applyMiddleware } from 'redux'
import thunk from 'redux-thunk';
import rootReducer from '../reducers'

export default function configureStore(initialState) {
  const store = createStore(rootReducer, initialState, applyMiddleware(thunk));
  return store;
}
```

Modifying it using actions

```
export function translate(text, target) {
  return dispatch => {
    return fetch('https://www.googleapis.com/language/translate/v2?key='+API_KEY+'&source=en&target='+target+'&q='+ text)
      .then(response => response.json())
      .then(response => dispatch(receiveTranslation(text, target, response.data.translations[0].translatedText)))
  }
}
```


LEARN REDUX

[Redux - official docs](#)

[Getting started with Redux - Video lectures by Dan Abramov](#)

[React - Redux - Udemy course \(PAID\)](#)

[A cartoon intro to Redux](#)

[How to manage request state with React & Redux](#)

Thank you!

Let's stay in touch



fb: dan.jipa,
tw: danjipa,
dan.jipa@gmail.com