Redux with React JS

Introduction: redux can be used with any frontend frameworks like react, angular or vue js.

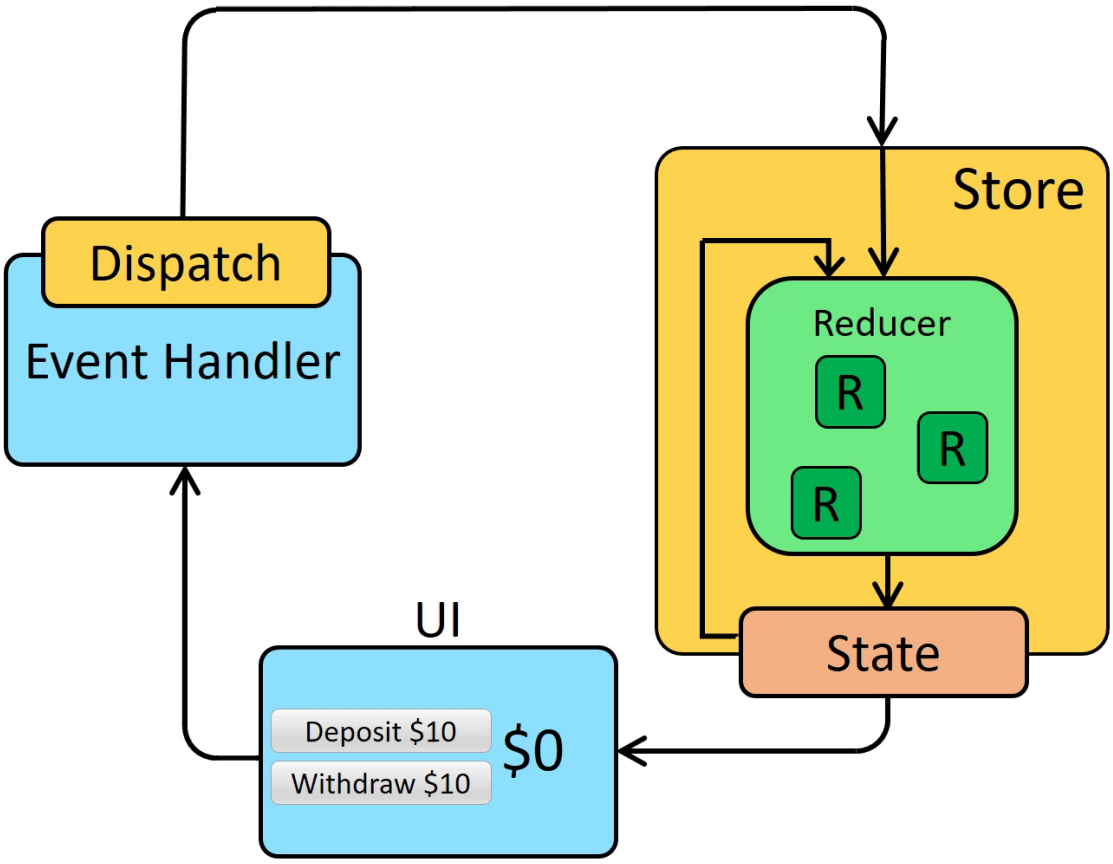
Redux is an open-source JavaScript library for managing and centralizing application state. It is most commonly used with libraries such as React or Angular for building user interfaces.

React Redux is the official React binding for Redux. It allows React components to read data from a Redux Store, and dispatch **Actions** to the **Store** to update data. Redux helps apps to scale by providing a sensible way to manage state through a **unidirectional** data flow model. React Redux is conceptually simple. It subscribes to the Redux store, checks to see if the data which your component wants have changed, and re-renders your component.

1. React Redux is the official UI bindings for react Application. It is kept up-to-date with any API changes to ensure that your React components behave as expected.
2. It encourages good 'React' architecture.
3. It implements many performance optimizations internally, which allows to components re-render only when it actually needs.

What is Redux

- A container where you can store your whole application data.
- so, we can say state management for whole application.
- We can store our states in redux for access anywhere in our application.
- Component state and redux both are not same


Store
Dispatch
Event Handler
Reducer
R
R
R
State
UI
Deposit $10
Withdraw $10
$0

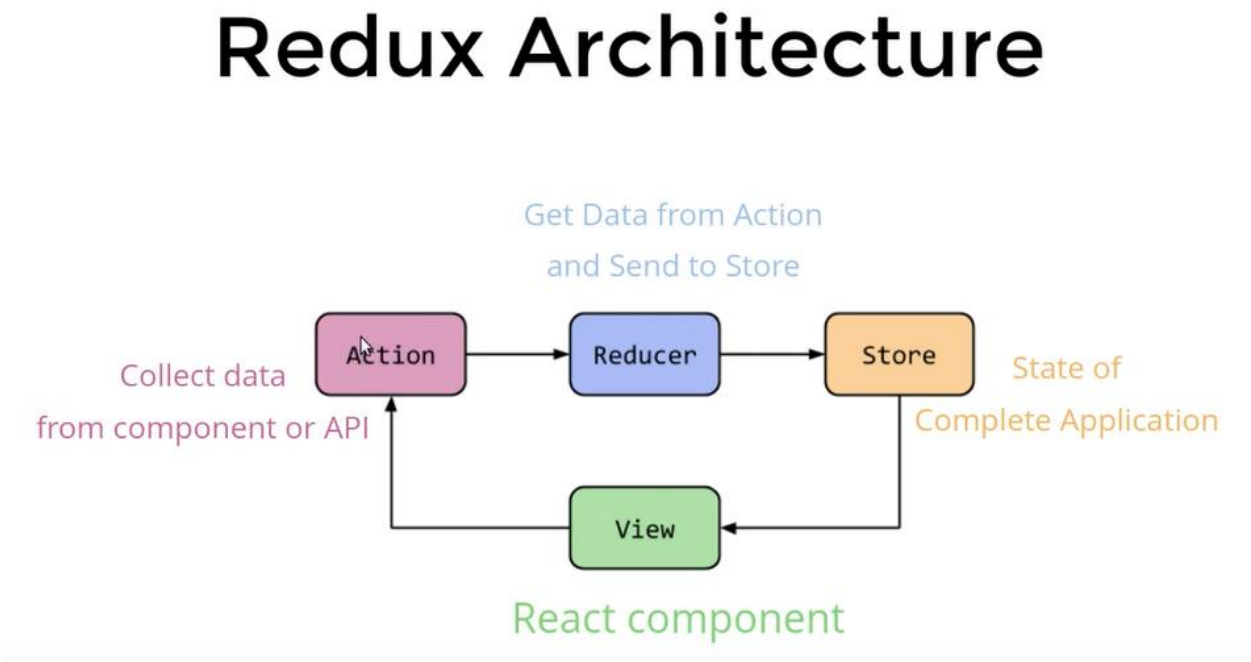
Redux Architecture

The components of Redux architecture are explained below.

STORE: A Store is a place where the entire state of your application lists. It manages the status of the application and has a dispatch(action) function. It is like a brain responsible for all moving parts in Redux.

ACTION: Action is sent or dispatched from the view which are payloads that can be read by Reducers. It is a pure object created to store the information of the user's event. It includes information such as type of action, time of occurrence, location of occurrence, its coordinates, and which state it aims to change.

REDUCER: Reducer read the payloads from the actions and then updates the store via the state accordingly. It is a pure function to return a new state from the initial state.



View / UI: User interface.

Action: used to collect data or event from user.

Reducer: forward data to store in predefined format.

Store: Used to save Redux Data.

Note: Application contains only 1 store for redux, and data flow in unidirectional format only.

Install Redux in React JS.

Create new project with **npx create-react-app reduxdemo**

Install redux in newly created project with **npm install redux, npm install react-redux**

Now you can verify redux in packages.json file

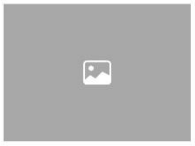
NOW, DELIVERY IS FREE FROM 100$ OF PUCHASE, IN EUROPE.

NEW ARRIVAL MEN WOMAN KIDS SHOP **FOOTWEAR**

GET READY FOR SUMMER VIVES

SHOP COLLECTION →

Find Categories You Looking For



NEW ARRIVAL

- Footwear
- Trending
- Converse
- Accessories
- Polos & T-shirt

On Service

- Top Printing
- Folding
- Poly Bagging
- Hong Tags
- Fail Printing
- Hem Tags

Buy Best Quality Shoes

Inspired by the mans footwear, Brings all-new mega special collection
Footwear charming lucid palette! Available Now.

Collection Type

- Trending
- Footwear
- Polos & T-shirt
- Accessories
- Converse
- Others+

Women's Top Sneikers
$85.00

Men's Converse Shoe
$85.00

Kids Summer Converse
$85.00

Children Su
$85.00

Customer's Kinds Word

"I really like them. I;m going to get more of them. I really like the Converse and they are comfortable"

Osman Gazi
UI/UX Designer

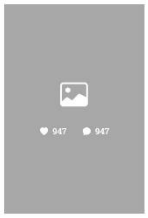
"Amazing and stylish footw to size classy. Have received Complinebt. Love it!"

Osman Gazi
UI/UX Designer

Follow us on our Instagram

Inspired by the mans footwear, Brings all-new mega special collection
Footwear charming lucid palette! Available Now.

#footwear



#footwear

#footwear

#footwear

#footwear

Buy Best Quality Shoes

shamsun..design@gmail.com Subscribe →

Pages

- Shop
- Collection
- New Arrival

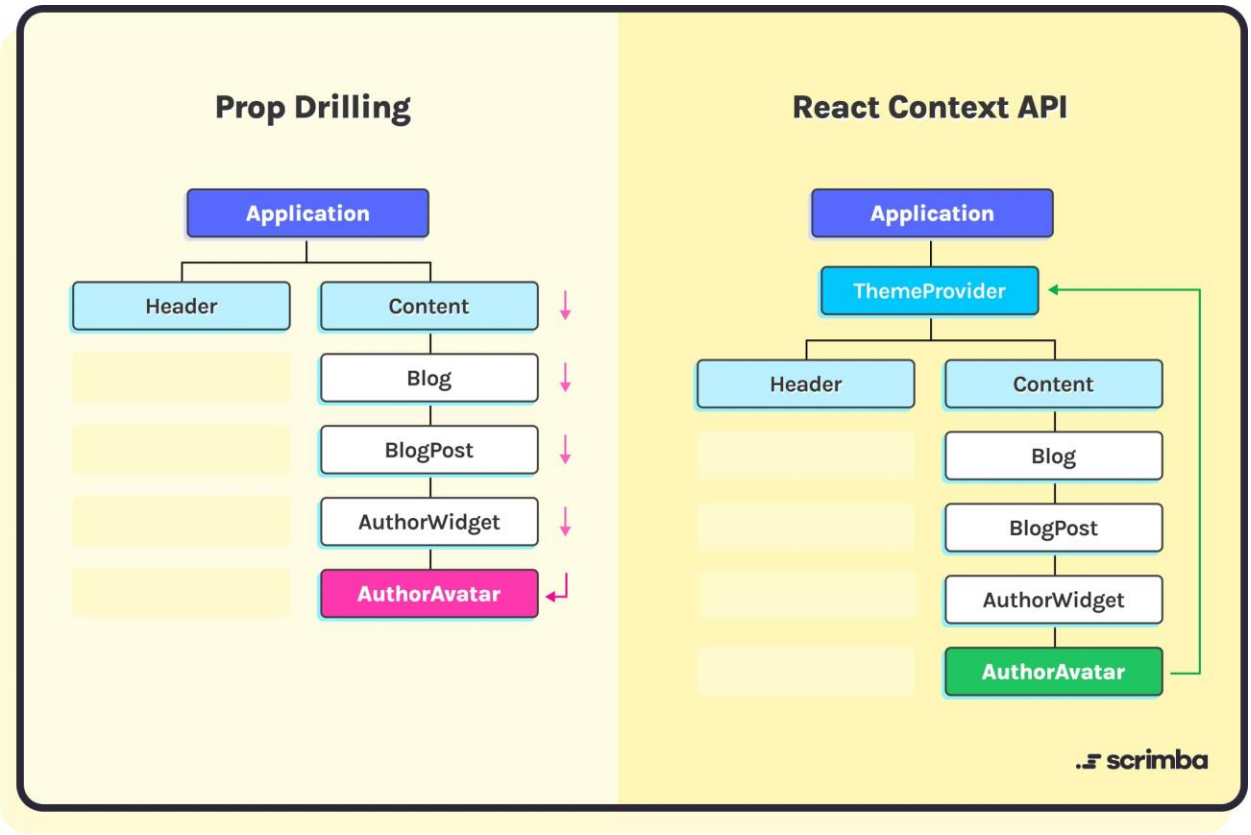
Company

- Contact Us
- Location
- Career
- Privacy Policy
- Collection

Follow

- Facebook
- Instagram
- Youtube
- Pinterest

When we have complex structure like e commerce websites, we need some of data from one element to another element and also reflect data back on component when it will be updated. So, without Redux we need to use concept of props drilling or context API.



Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

import { Provider } from 'react-redux';
import { store } from './Redux/Store';
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Provider store={store}>
    <App />
    </Provider>
  </React.StrictMode>
);
reportWebVitals();
```

App.js

```
import logo from './logo.svg';
import './App.css';
import Counter from './Components/Counter';
import { useDispatch } from 'react-redux';
function App() {
  const dispatch = useDispatch();
  return (
    <div className="App">
      <button onClick={(e) => dispatch({type:"INCREMENT"})}>Increment
</button>
      <Counter></Counter>
      <button onClick={(e) => dispatch({type:"DECREMENT"})}>Decrement
</button>
    </div>
  );
}

export default App;
```

Components/Counter.js

```
import { useSelector } from "react-redux";

function Counter(){
    const count = useSelector(state => state);
    return <>
        <h1>Value of Counter is {count}</h1>
    </>
}

export default Counter;
```

Redux/Store.js

```
import {legacy_createStore as createStore} from "redux";

const reducer = (state = 0, action) => {
    switch(action.type){
        case 'INCREMENT' :{
            return state + 1;
        }
        case 'DECREMENT' : {
            return state - 1;
        }
        default :{
            return state;
        }
    }
}

export const store = createStore(reducer);
```