Intro to Redux

Cory House

@housecor   reactjsconsulting.com

Here's The Plan

Do I need Redux?

3 Principles

Flux vs Redux

Redux Flow

Do I need Redux?

Vanilla JS   **jQuery**   **React**   **React + Redux**

Simple   Complex

No setup   Significant setup

When Do I Need Redux?

Complex data flows

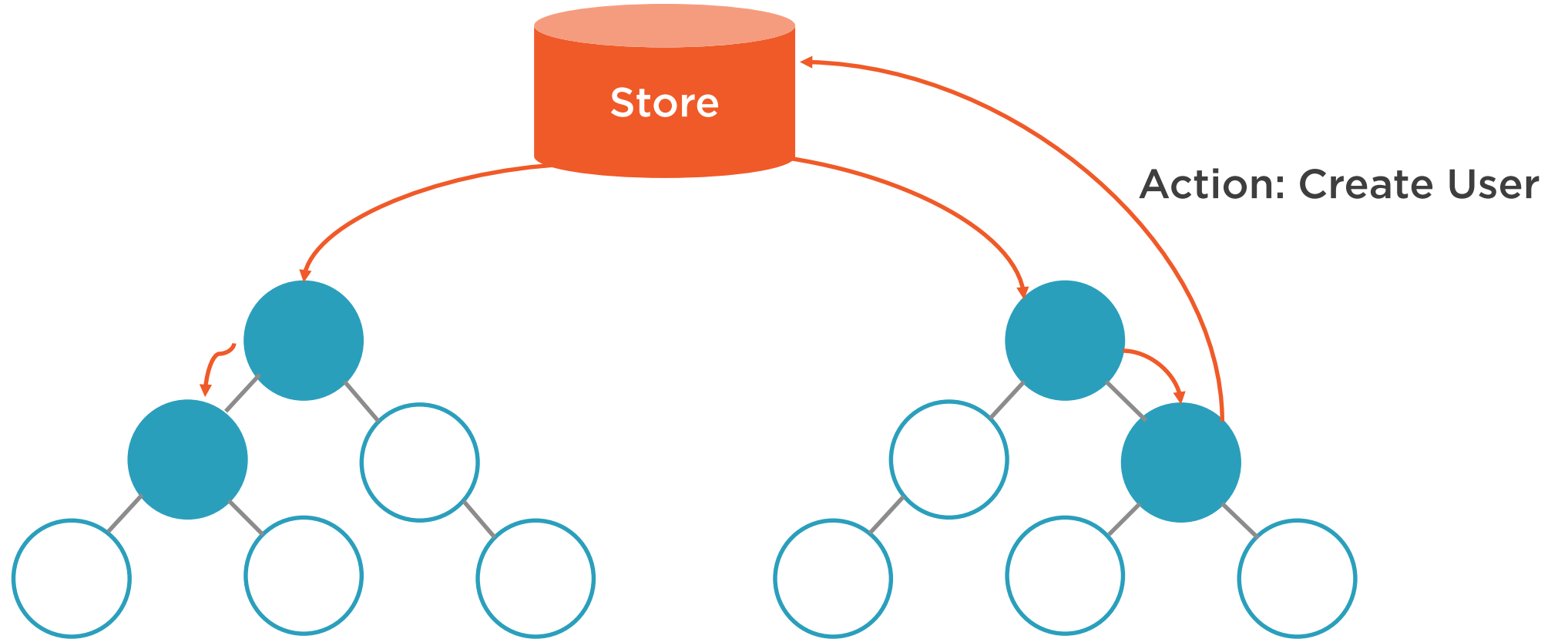Inter-component communication

Non-heirarchical data

Many actions

Same data used in multiple places

"…If you aren't sure if you need it, you don't need it."
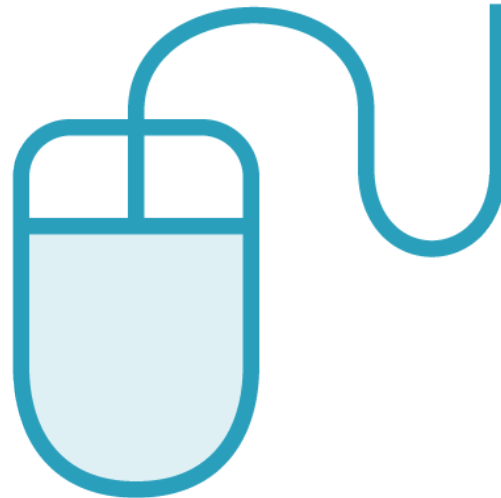
Pete Hunt on Flux and Redux

Store

Action: Create User

Redux: 3 Principles

One immutable store   **Actions trigger changes**   **Reducers update state**

Flux vs Redux

Data down.
Actions up.

Flux and Redux: Similarities

Unidirectional Flow
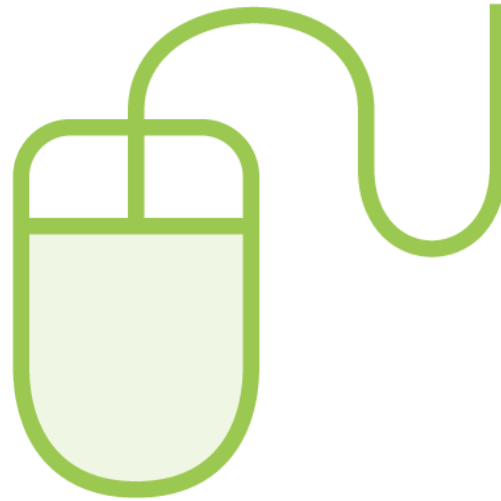
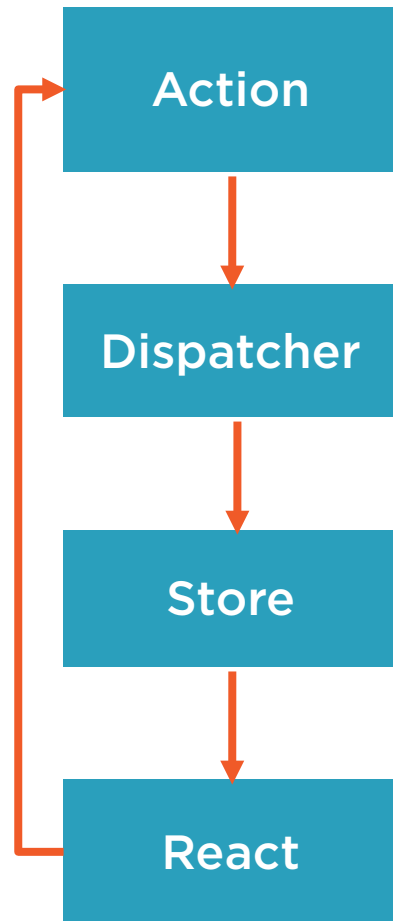**Actions**

Stores

Redux: New Concepts

Reducers

Containers

Immutability

Flux vs Redux

Flux

Redux
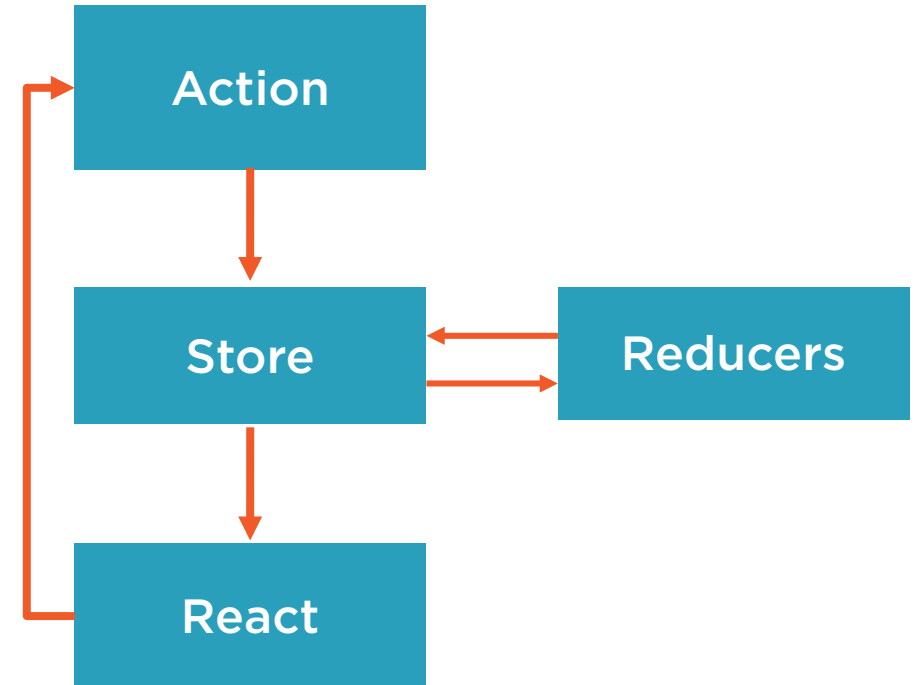
Flux	Redux
Stores contain state and change logic	**Store and change logic are separate**

Flux	Redux
Stores contain state and change logic	**Store and change logic are separate**
Multiple stores	**One store**

	Flux	Redux
Stores contain state and change logic	Store and change logic are separate
Multiple stores	One store
Flat and disconnected stores	Single store with hierarchical reducers

Flux	Redux
Stores contain state and change logic	Store and change logic are separate
Multiple stores	One store
Flat and disconnected stores	Single store with hierarchical reducers
Singleton dispatcher	No dispatcher

	Flux	Redux
	Stores contain state and change logic	Store and change logic are separate
	Multiple stores	One store
	Flat and disconnected stores	Single store with hierarchical reducers
	Singleton dispatcher	No dispatcher
	React components subscribe to stores	Container components utilize connect

	Flux	Redux
	Stores contain state and change logic	Store and change logic are separate
	Multiple stores	One store
	Flat and disconnected stores	Single store with hierarchical reducers
	Singleton dispatcher	No dispatcher
	React components subscribe to stores	Container components utilize connect
	State is mutated	State is immutable

Redux Flow

Redux Flow

Action

{ type: RATE_COURSE, rating: 5 }

Store

Reducers

```
function appReducer(state = defaultState, action) {
    switch(action.type) {
        case RATE_COURSE:
            //return new state
    }
}
```

React

Notified via React-Redux
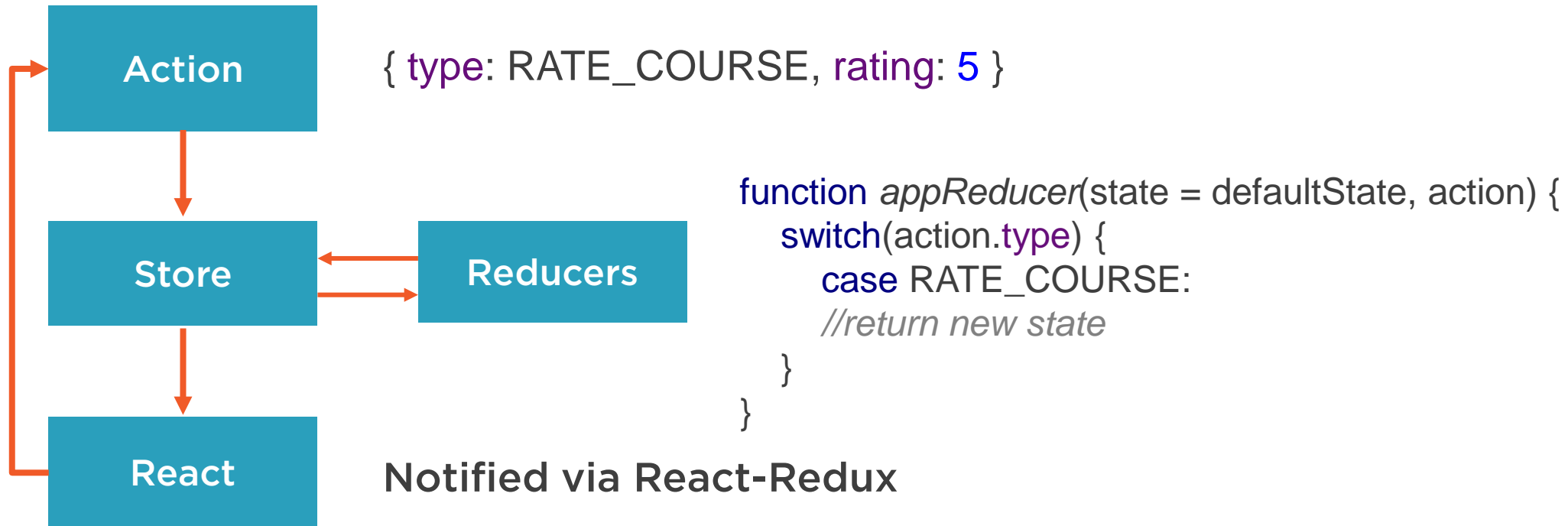
Summary

If you need Redux, you'll know it.

3 Principles
- Immutable Store
- Actions trigger changes
- Reducers return updated state

Flux vs Redux

Unidirectional Redux Flow

Next up: Actions, stores, and reducers