

Connecting React to Redux



Cory House

Consultant

@housecor | www.reactjsconsulting.com

Here's The Plan



Container vs Presentation Components

React-Redux

- Provider
- Connect

A Chat with Redux



Two Component Types

Container

Focus on how things work

Aware of Redux

Subscribe to Redux State

Dispatch Redux actions

Generated by react-redux

VS

Presentational

Focus on how things look

Unaware of Redux

Read data from props

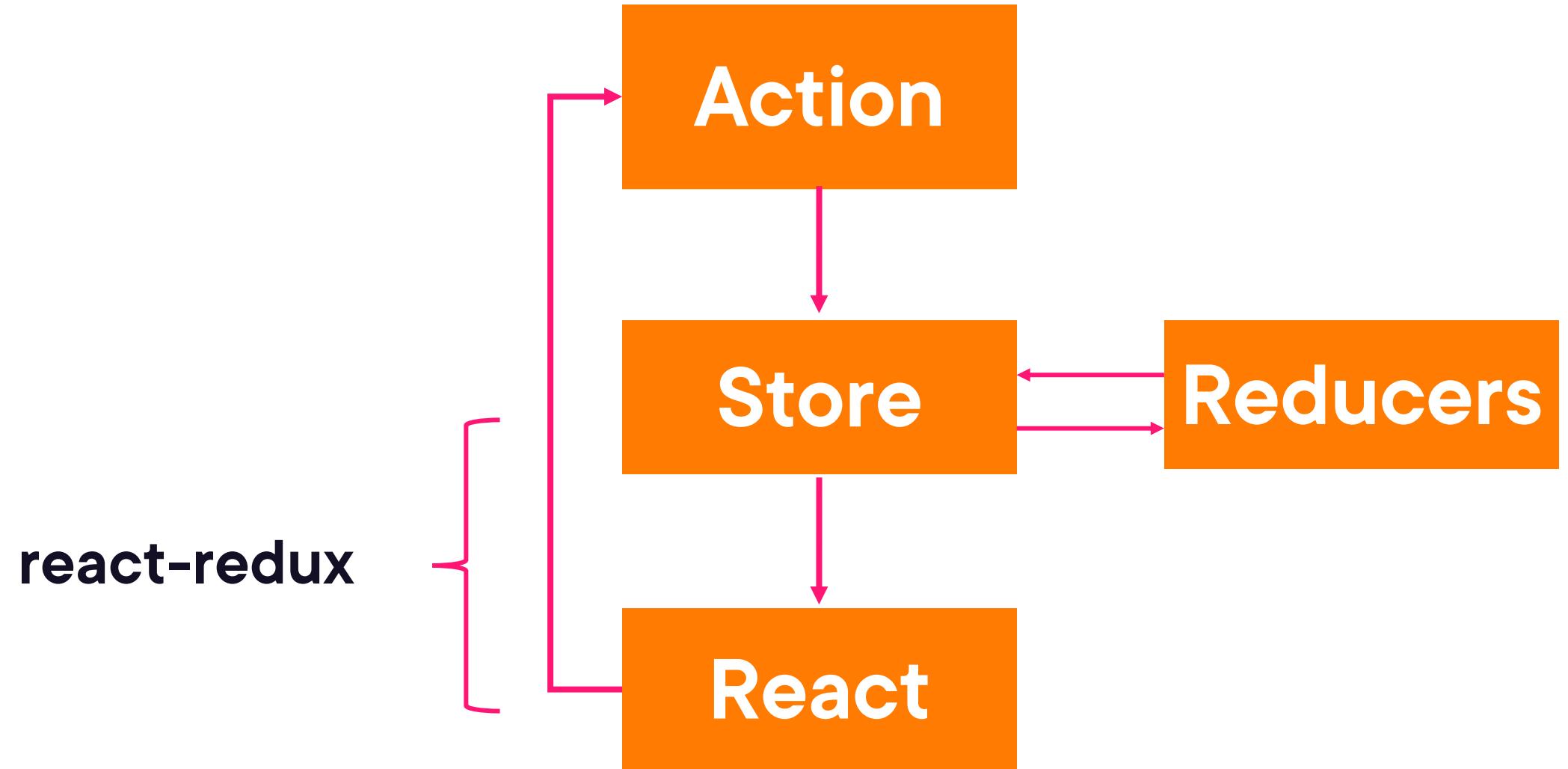
Invoke callbacks on props

Written by hand



Connecting React to Redux





React-Redux Provider

```
<Provider store={store}>
  <App />
</Provider>
```

```
useSelector()
useDispatch()
```

```
connect()
```



React-Redux

```
<Provider store={store}>
  <App />
</Provider>
```



```
import React, { useEffect } from "react";
import { useDispatch, useSelector } from "react-redux";
import { loadCourses } from "../../redux/actions/courseActions";

export default function ManageCoursePage() {
  const courses = useSelector(state => state.courses);
  const dispatch = useDispatch();

  useEffect(() => {
    dispatch(loadCourses());
  }, [courses]);

  return <p>{courses.length} courses</p>;
}
```

Hooks

useSelector: Select Redux data
useDispatch: Dispatch actions



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

connect

Wraps our component so it's connected to the Redux store.



```
function mapStateToProps(state, ownProps) {  
  return { authors: state.authors };  
}  
  
export default connect(mapStateToProps, mapDispatchToProps)(AuthorPage);
```

Redux with Connect



React-Redux Connect

```
connect(mapStateToProps, mapDispatchToProps);
```



What state should I expose as props?

```
function mapStateToProps(state) {  
  return {  
    appState: state,  
  };  
}
```



React-Redux Connect

```
connect(mapStateToProps, mapDispatchToProps);
```



What state should I expose as props?

```
function mapStateToProps(state) {  
  return {  
    users: state.users,  
  };  
}
```



React-Redux Connect

```
connect(mapStateToProps, mapDispatchToProps);
```



What state should I expose as props?

```
function mapStateToProps(state) {  
  return {  
    appState: state,  
  };  
}
```



React-Redux Connect

```
connect(mapStateToProps, mapDispatchToProps);
```



What actions do I want on props?

```
function mapDispatchToProps(dispatch) {
  return {
    actions: bindActionCreators(actions, dispatch),
  };
}
```



4 Ways to Handle `mapDispatchToProps`

1. **Ignore it**
2. **Wrap manually**
3. **`bindActionCreators`**
4. **Return object**



```
// In component...
props.dispatch(loadCourses())
```

Option 1: Use Dispatch Directly

Downsides

1. Boilerplate
2. Redux concerns in child components



Option 2: Wrap Manually

```
function mapDispatchToProps(dispatch) {  
  return {  
    loadCourses: () => {  
      dispatch(loadCourses());  
    },  
    createCourse: (course) => {  
      dispatch(createCourse(course));  
    },  
    updateCourse: (course) => {  
      dispatch(updateCourse(course));  
    },  
  };  
}  
  
// In component...  
props.loadCourses()
```



Option 3: bindActionCreators

```
function mapDispatchToProps(dispatch) {  
  return {  
    actions: bindActionCreators(actions, dispatch),  
  };  
}  
  
Wraps action creators in dispatch call for you
```

```
// In component:  
props.actions.loadCourses();
```



Option 4: mapDispatchToProps as Object

```
const mapDispatchToProps = {  
  loadCourses,  
};  

```

Wrapped in dispatch automatically

```
// In component:  
props.loadCourses();
```



4 Ways to Handle `mapDispatchToProps`

```
props.dispatch(loadCourses());
```

Ignore it

```
function mapDispatchToProps(dispatch) {  
  return { loadCourses: () =>  
    dispatch(loadCourses());  
  };  
}
```

Manually wrap in dispatch

```
function mapDispatchToProps(dispatch) {  
  return {  
    actions: bindActionCreators(actions, dispatch)  
  };  
}
```

Use `bindActionCreators`

```
const mapDispatchToProps = {  
  incrementCounter,  
};
```

Return object



A Chat With Redux

React

Hey CourseAction, someone clicked this “Save Course” button.

Action

Thanks React! I will dispatch an action so reducers that care can update state.

Reducer

Ah, thanks action. I see you passed me the current state and the action to perform. I’ll make a new copy of the state and return it.

Store

Thanks for updating the state reducer. I’ll make sure that all connected components are aware.

React-Redux

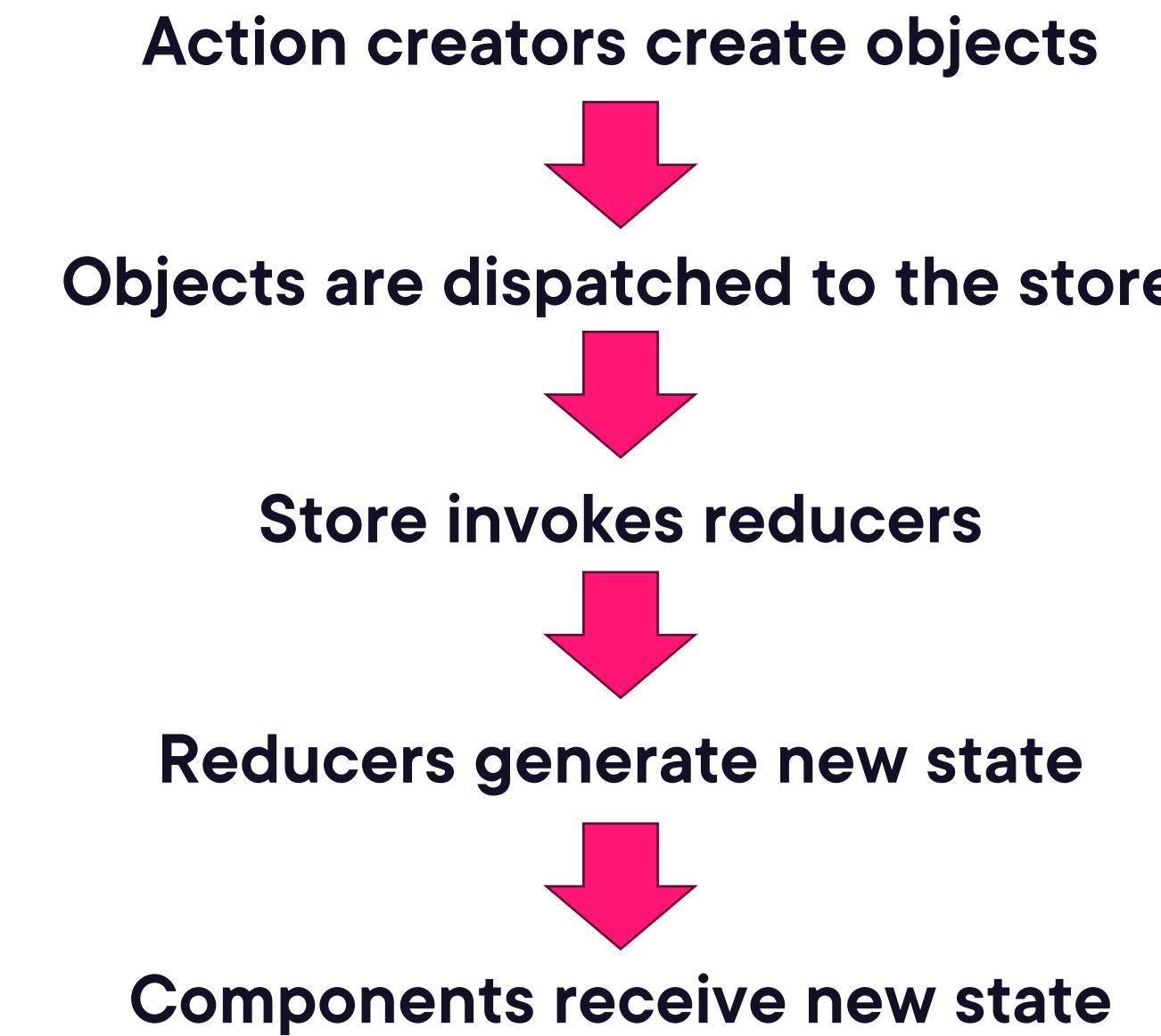
Woah, thanks for the new data Mr. Store. I’ll now intelligently determine if I should tell React about this change so that it only has to bother with updating the UI when necessary.

React

Ooo! Shiny new data has been passed down via props from the store! I’ll update the UI to reflect this!



Redux Flow



Summary



Container vs Presentation Components

React-Redux

- Provider
- Connect
 - mapStateToProps
 - mapDispatchToProps



Up Next:

Let's code Redux

