

🚩 Current Skill Redux Overview

## Introduction

One of the features provided by React is one-way data flow. Although this feature can help developers debug and track data, when we get to state management (especially when the application grows over time), it can open the door to hell.

Redux is one of the most known and used state management libraries. Managing states with Redux will be our main interest in this Super Skill. Here is what we're going to see:

- What is Redux?
- The mechanism that Redux uses to manage states.
- Redux middleware.
- Redux with hooks.

## Managing states with React

By definition, a state is an object that defines the current condition of our application.

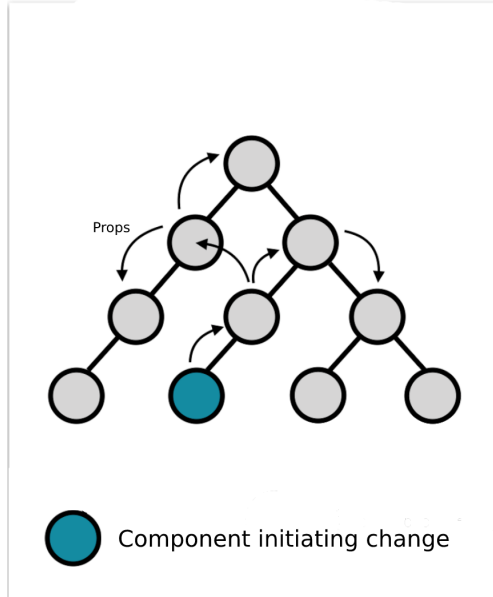
Since React is a component-based UI library, which uses the state to track the moving parts of the user interface, state management in applications is very important to understand.

React allows developers to use state at the component level by using class-based components or functional-based components.

## The Problem With State Management

Since component state can be passed around in props, it becomes confusing quickly. For example, if you need to pass the state deep inside a component tree to a child, you have to pass it through every component in between. This is called **prop-drilling**, and it makes it increasingly difficult to keep track of your state or modify any code in the future as it turns into a tedious process.

To solve the problem of prop-drilling, developers working with React often use some sort of state management tool. This is where many people switch to Redux to manage state.



## Getting started with Redux

To solve the problem of prop-drilling, developers working with React often use some sort of state management tool. This is where many people switch to Redux to manage state.

At first look, Redux seems complex and absolutely unnecessary. Some of the questions many new developers have on their minds when learning Redux are, “Why do I need to separate my state?” And, “What are all these new terms and how did my folder structure become so messy?”

But once you understand the core concept behind Redux, it'll be the solution for all state management problems.

[< Previous](#)

[next >](#)