

# REACT PROPS



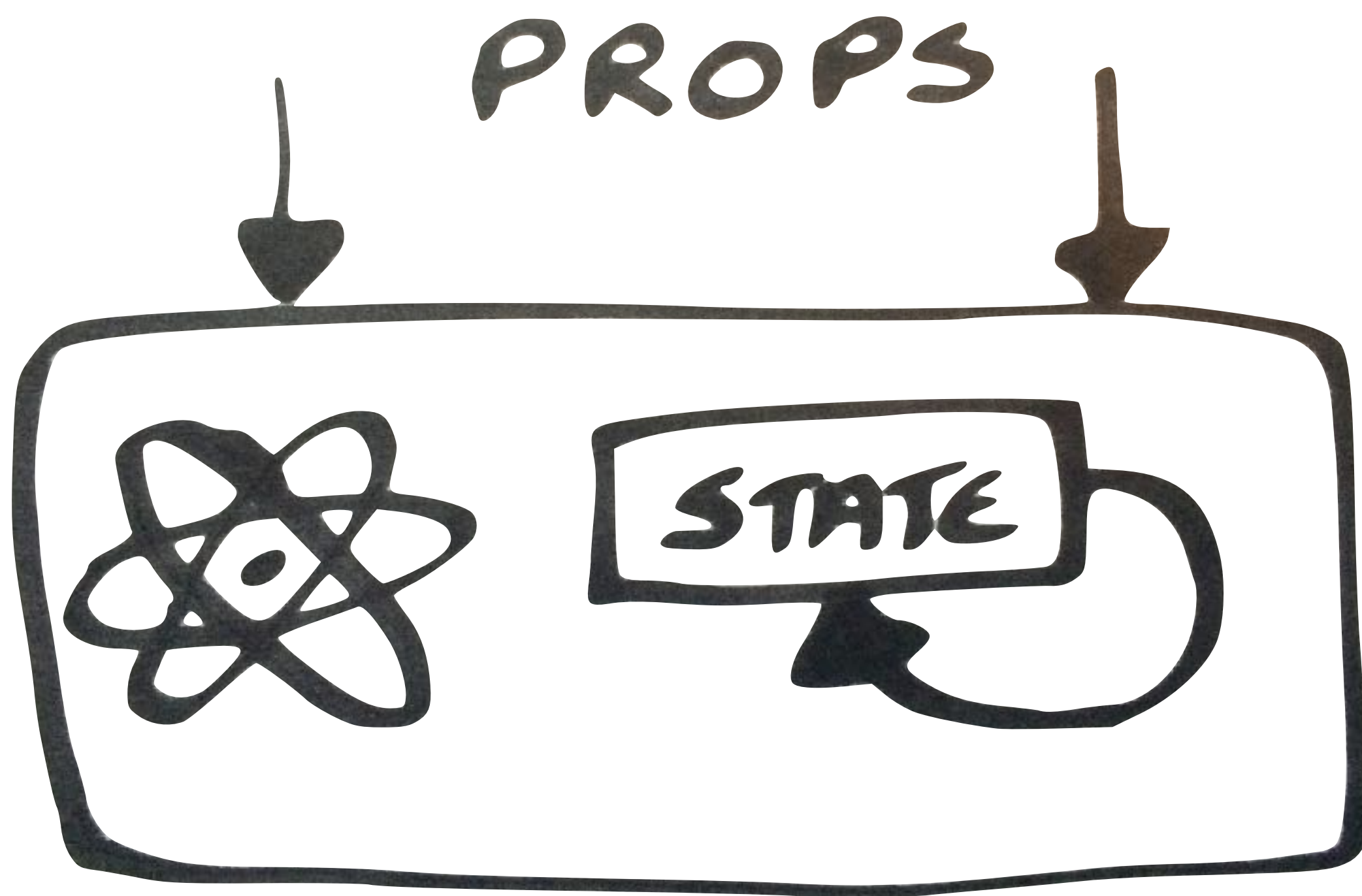
# REACT PROP PATTERNS



# COMPONENTS

- **Self-contained, concern-specific building blocks**
- **Components:**
  - are reusable
  - are composable
  - lead to separation of concerns
- **Allows a “Divide and conquer” approach to development**

# PROPS VS STATE



- Think of Props as components attributes, or “configuration”. They are received from parent components.
- A component manages its own state internally. State usually starts with a default value defined in the component’s constructor and then suffers from mutations in time (in. eg. generated from user events).

# CONTENTS

- **Props and Component Composition**
- **Default Props**
- **Children prop**
- **React.Children API**

# PROPS AND COMPONENT COMPOSITION

# PROPS & COMPONENT COMPOSITION

- **Composing components is an essential pattern in React for building complex UIs and applications.**
- **Props are of key importance in component composition. They are the mechanism used in React for passing data from parent to child components.**
- **Props are Read-Only: They can't be changed from inside. they are passed and “owned” by the parent component.**

# CODEPEN