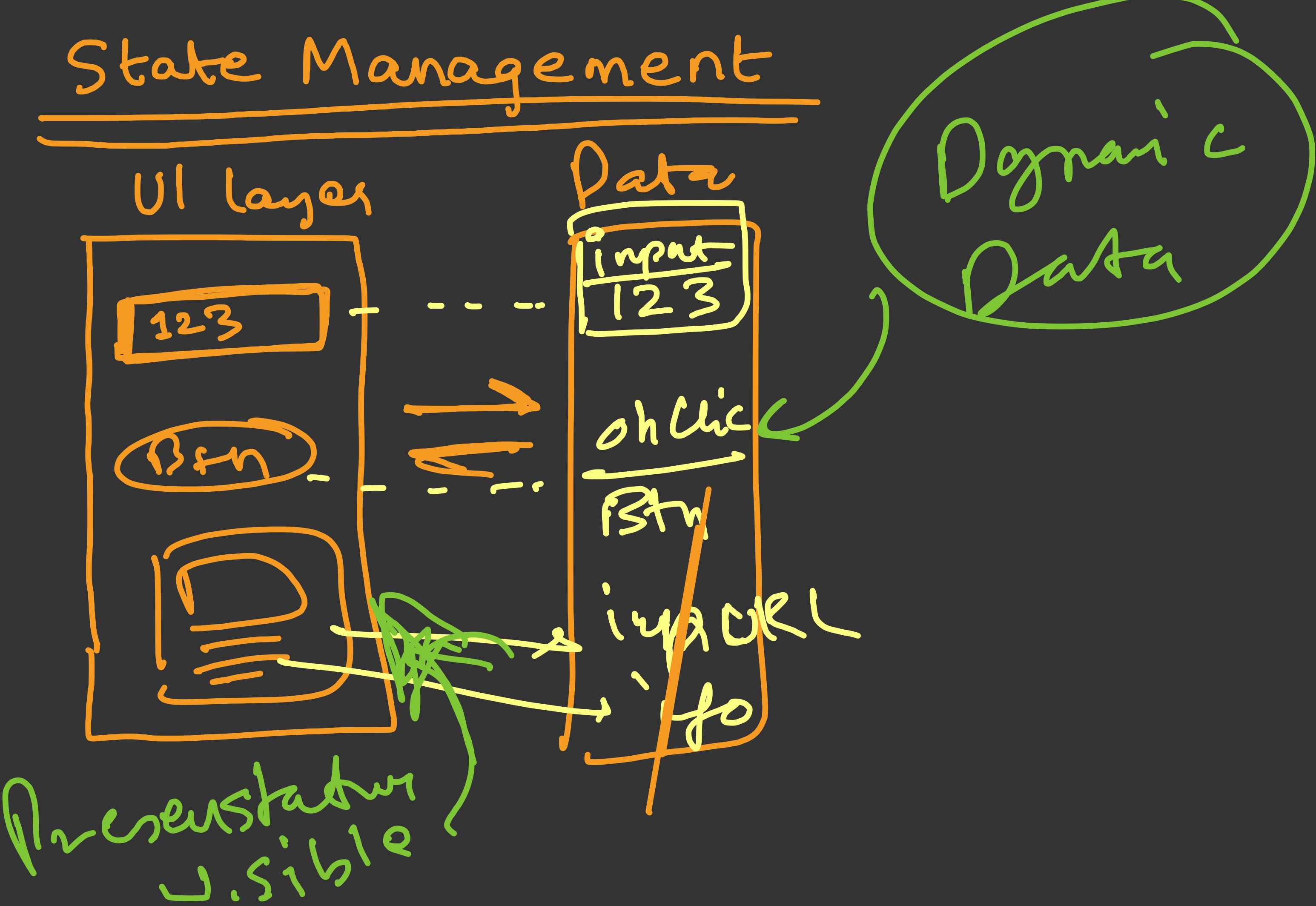
State Management

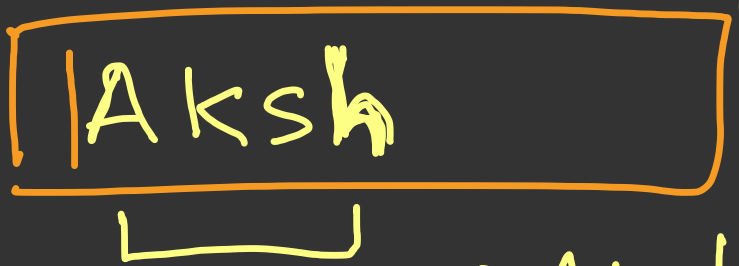


Why do you need State??

↳ Dynamic Behavior

Interactivity

Name



value ⇒ Aksh

let name = "Aksh"

① Local State ✓

(Component)

React

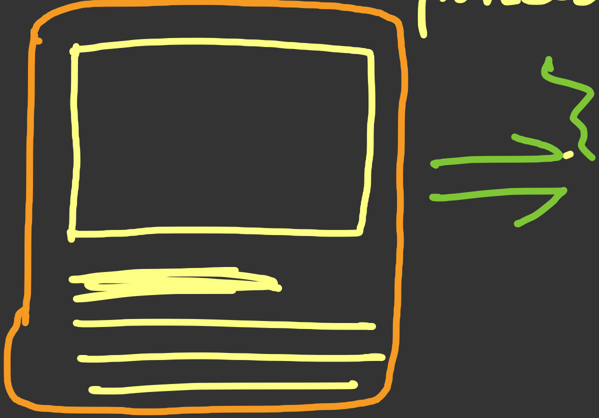
Context API

② Props Drilling ✓

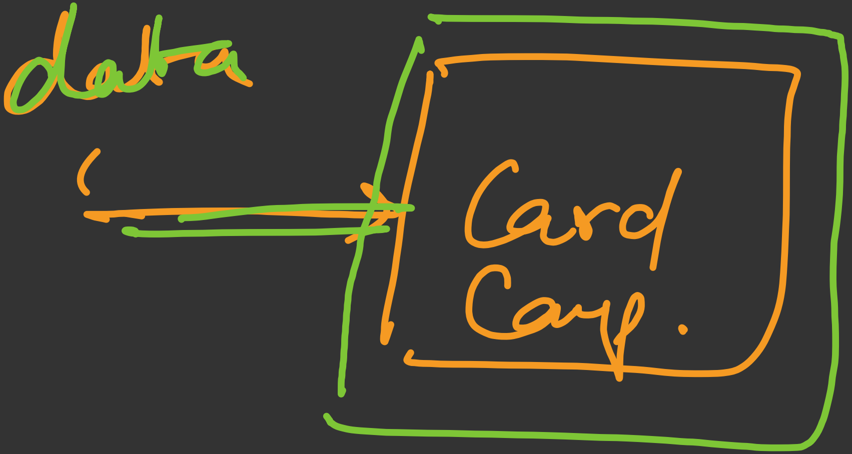
(Passing data b/w Components)

③ State Management Lib. ✓

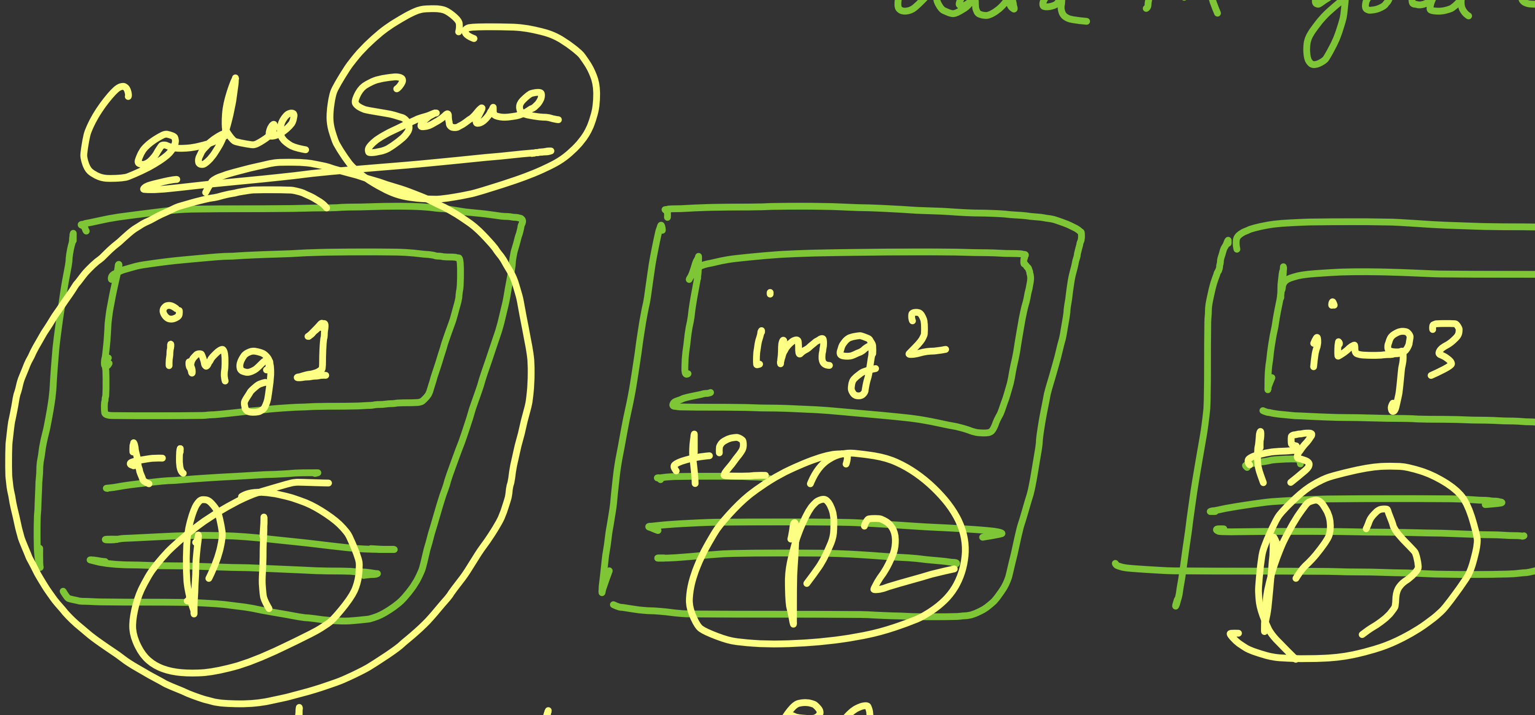
<Card Component />



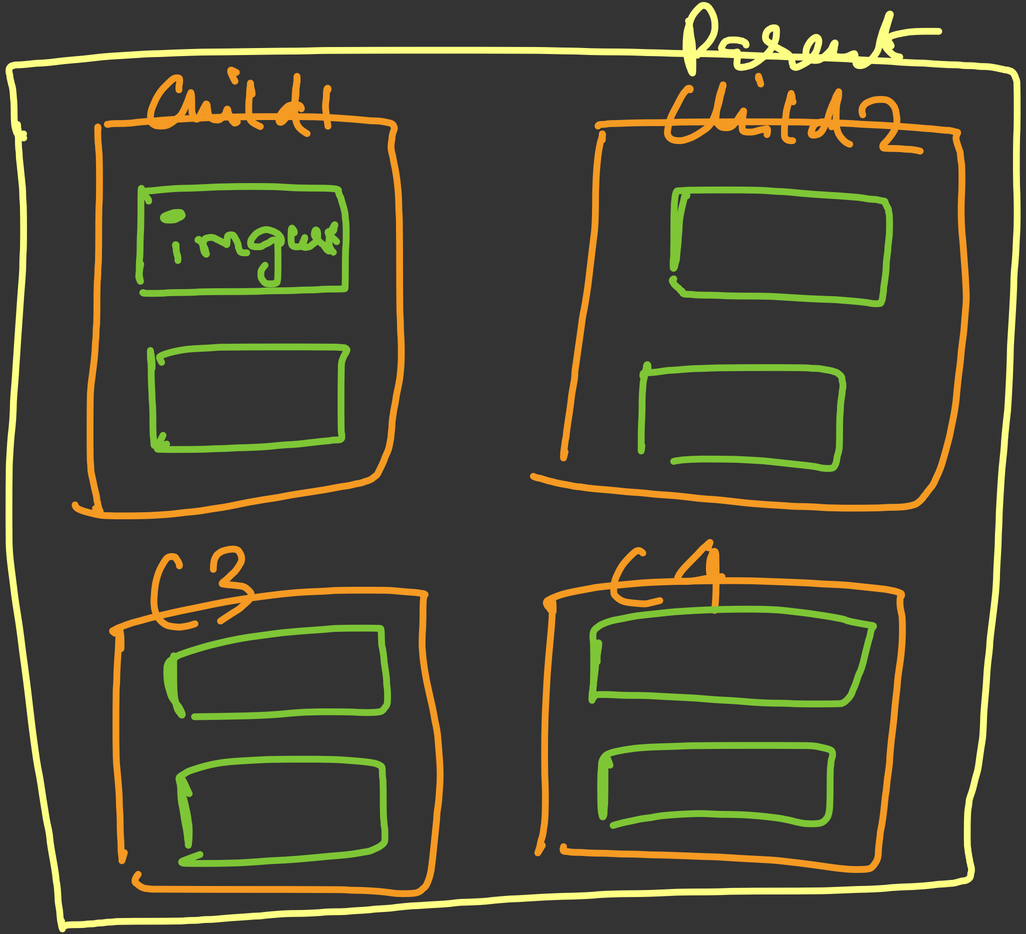
{img_url, title, description,} {local State}



Local State → any dynamic data in your app

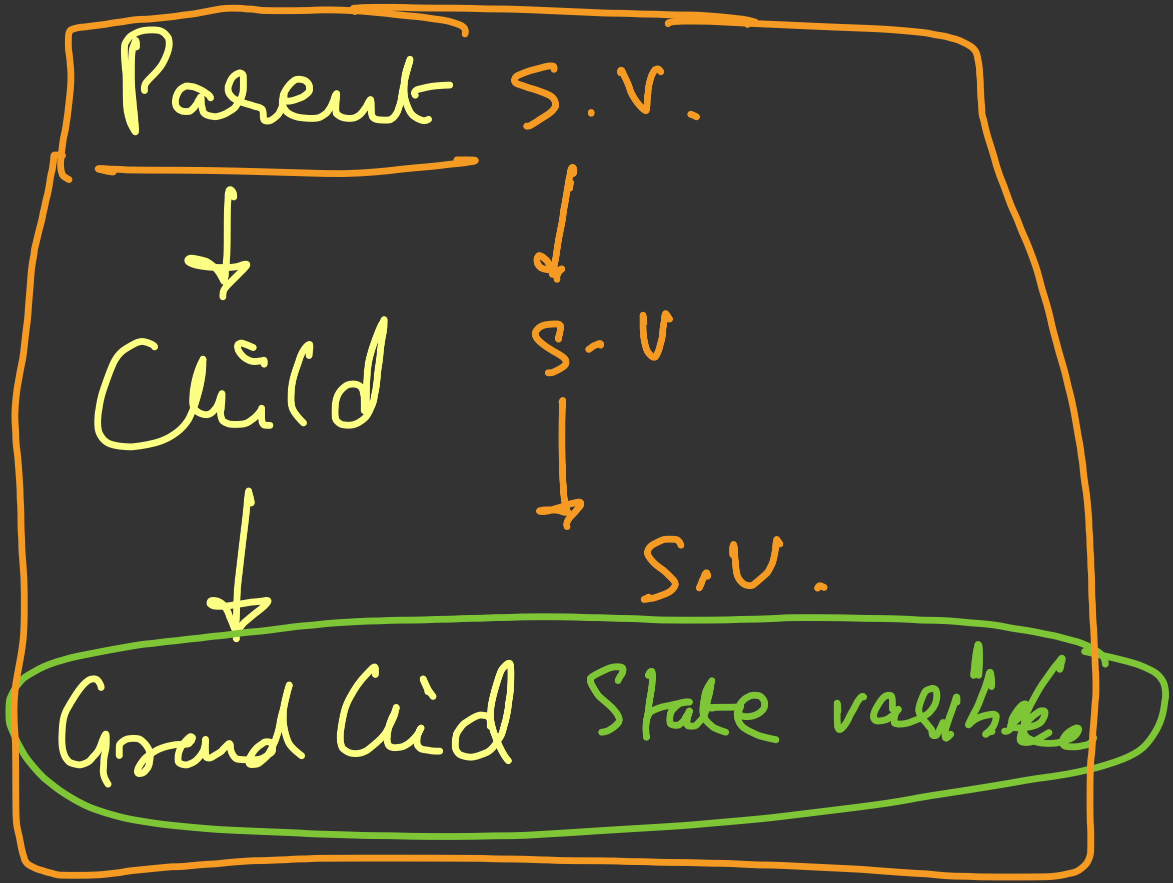


data changes

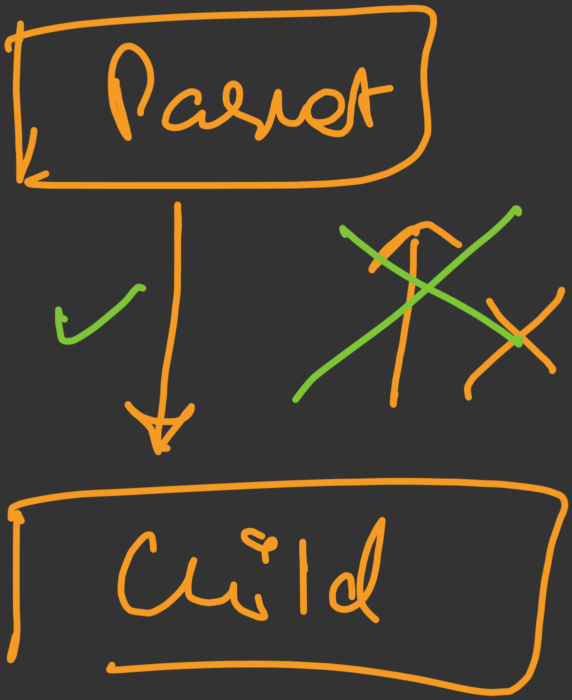


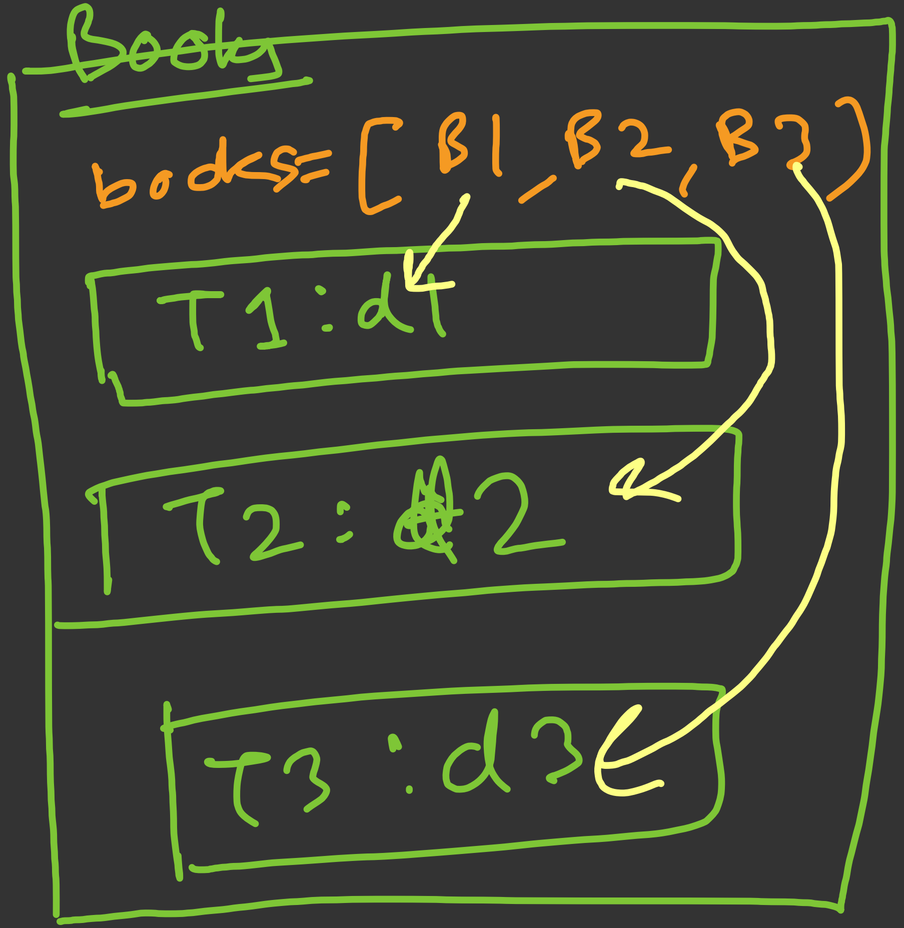
Local State

x → Component level

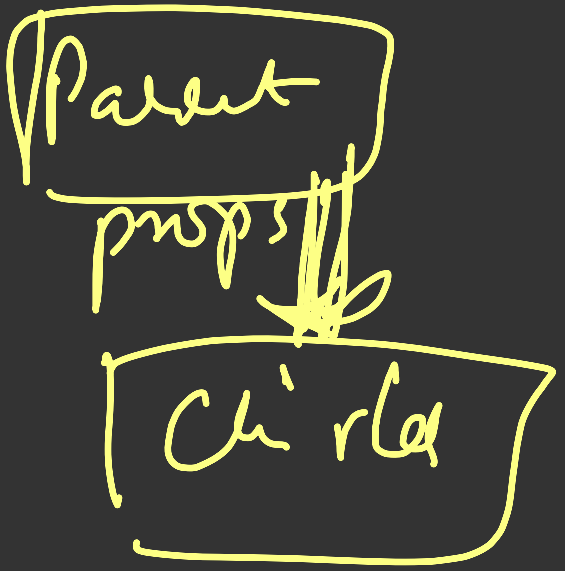

Parent
s.v.
Child
s.v
Grand Child
s.v.
State variable

Data flow - Uni directional

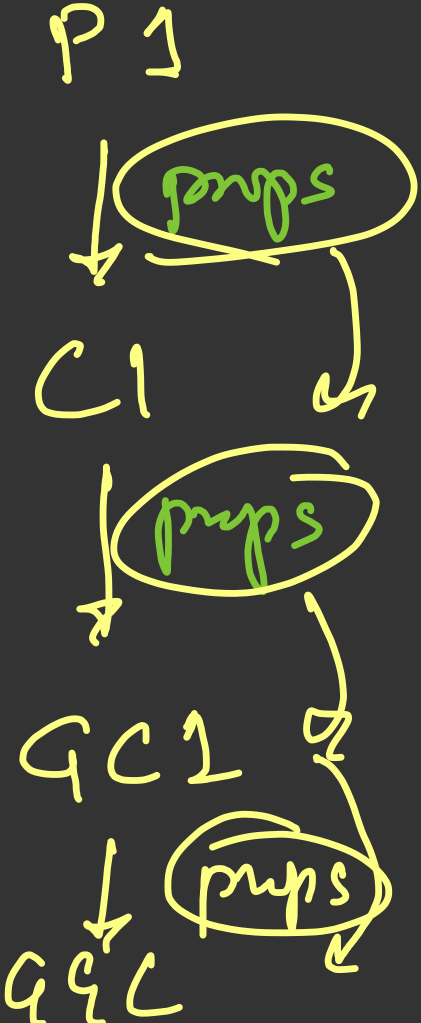


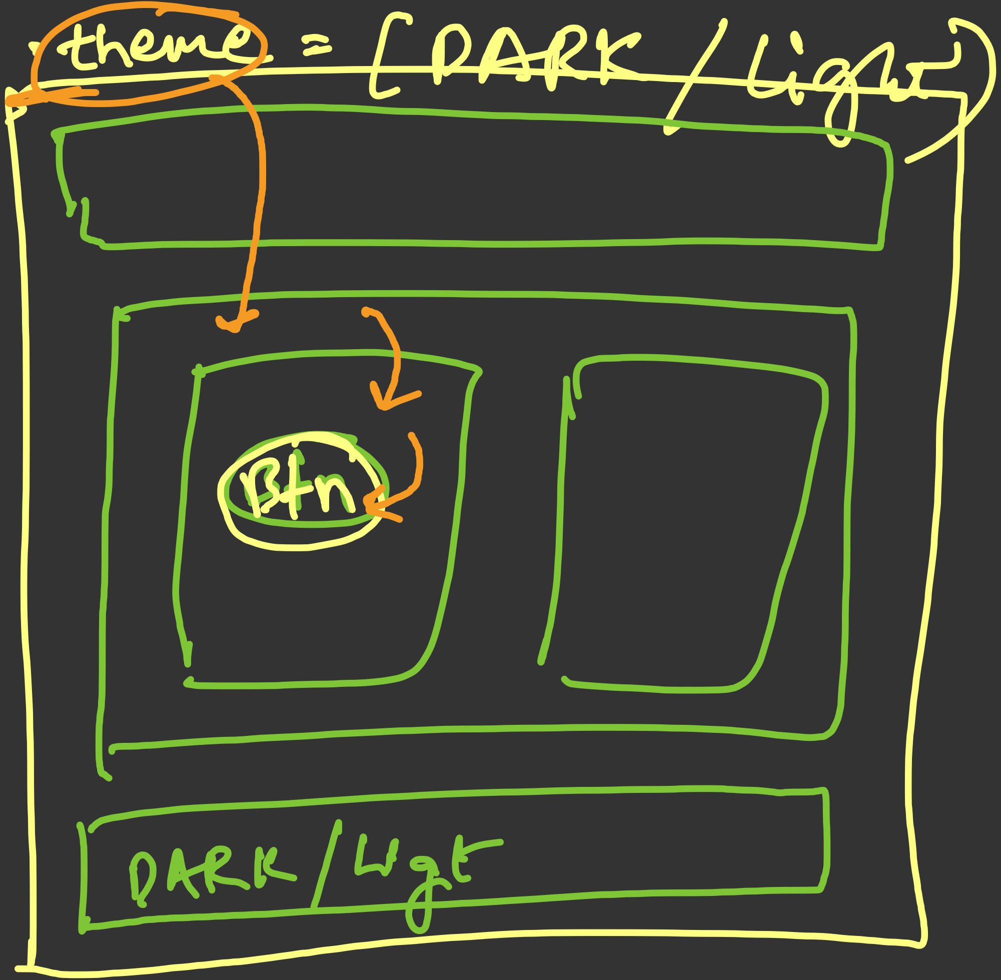


Passing props

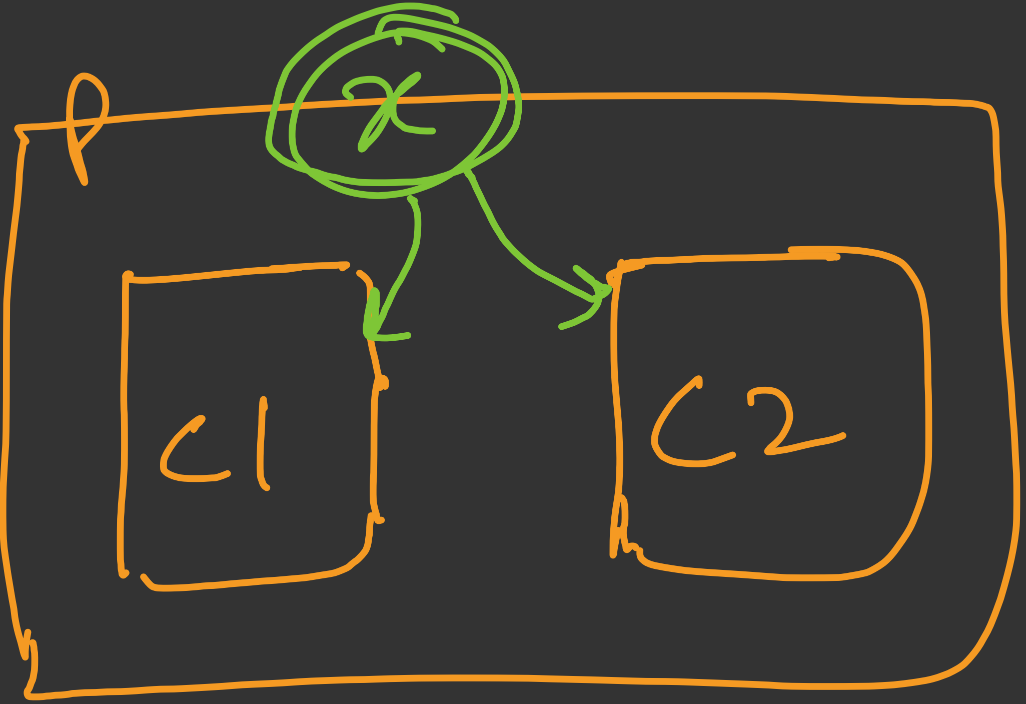


Props drilling





Props drilling

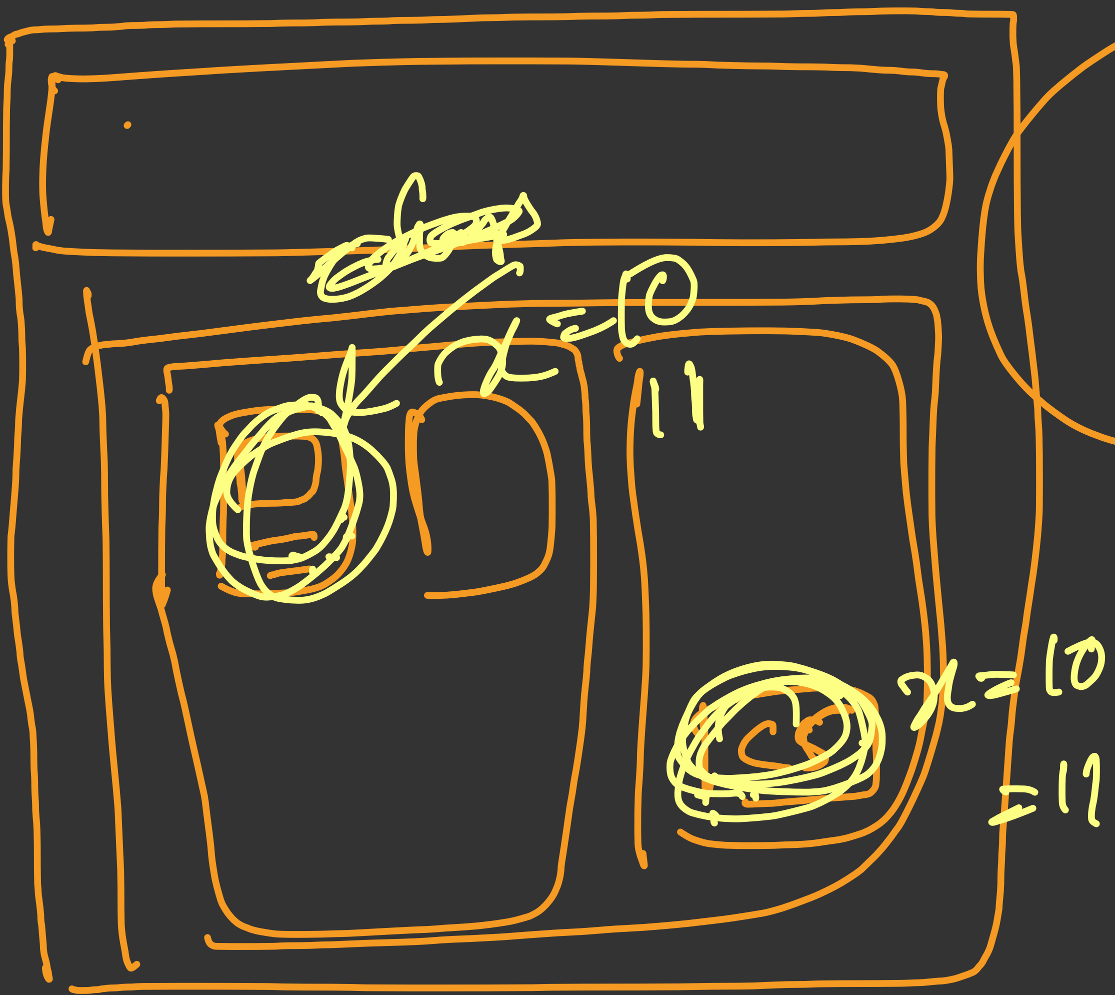


Lifting State up

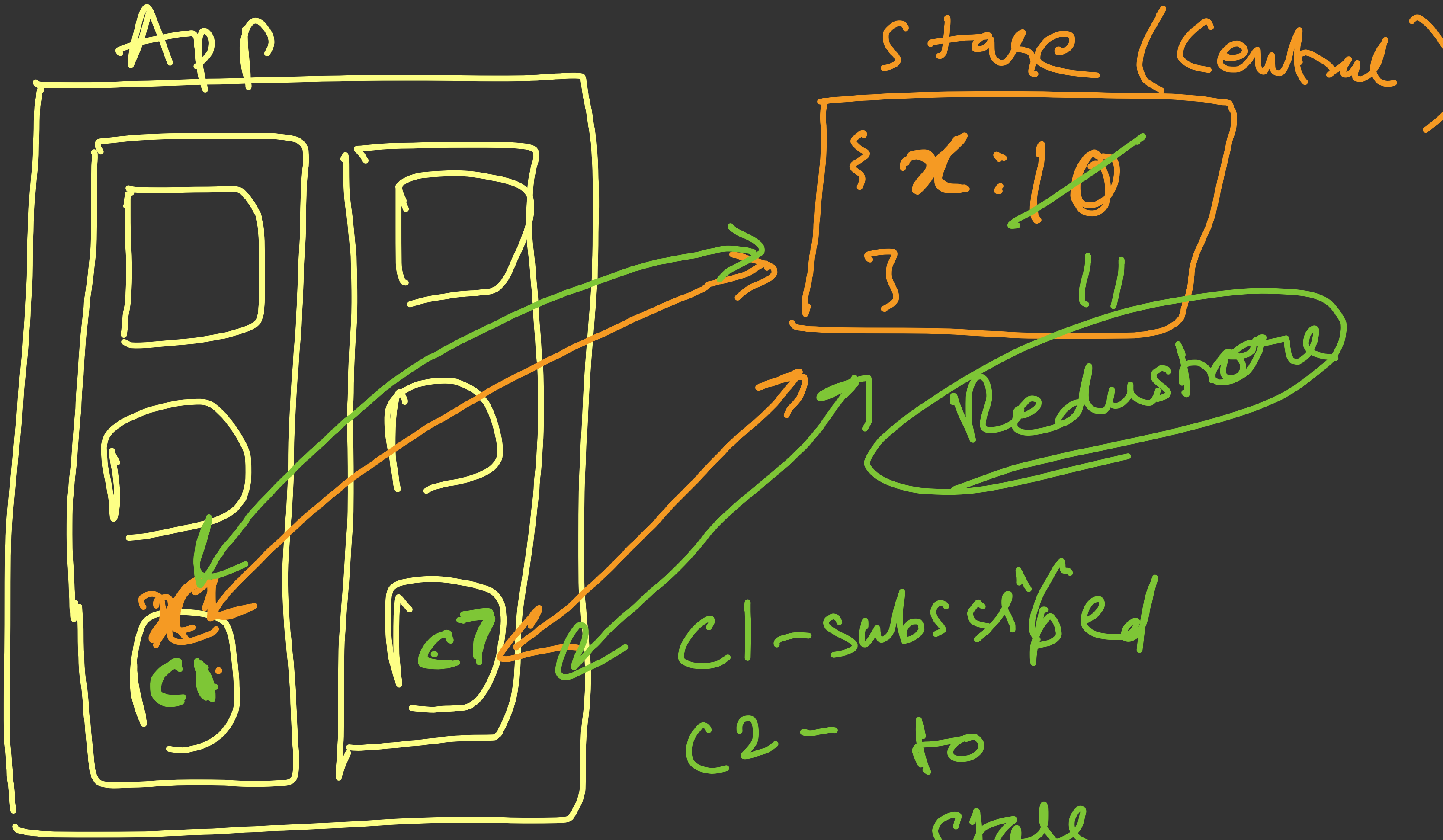
State Manage Libraries

Redux / ngRx

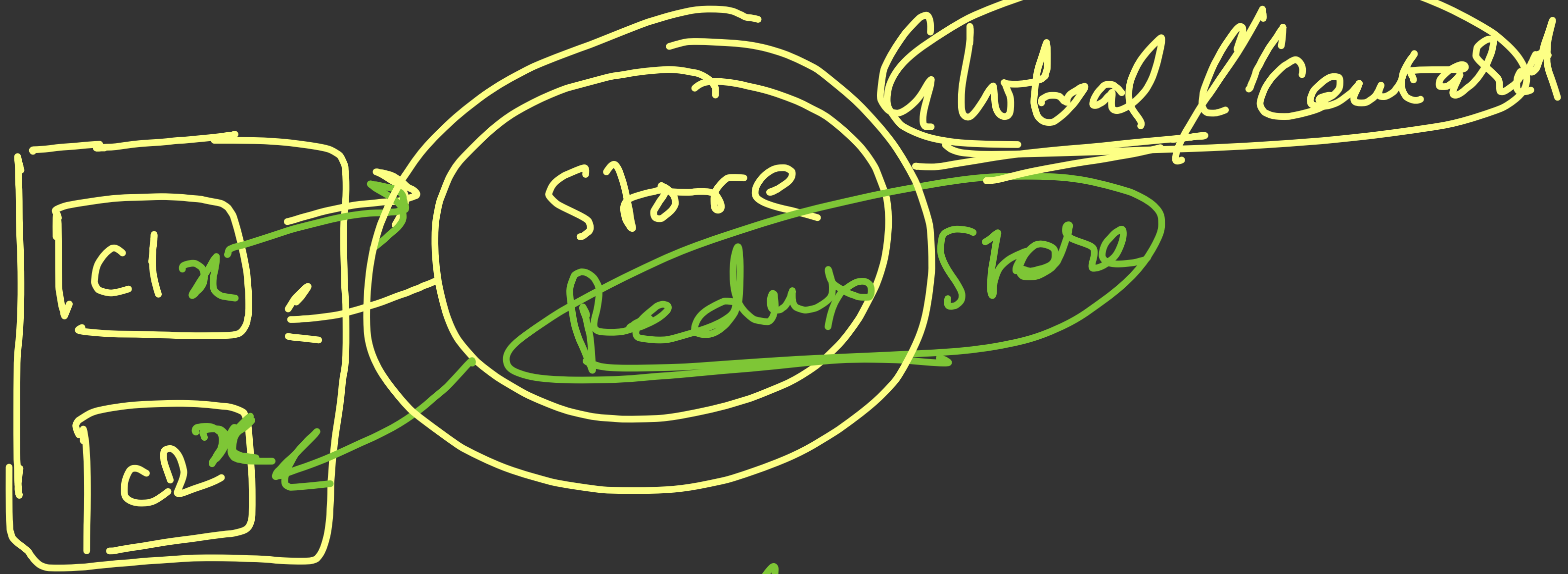
Zustand ↑↑↑



Huge Component Heirarchy



C1 - subscribed
C2 - to
state



#large scale

{ data management
(state) }

DON'T use State Management library for very small projects

Context API

# Debugging is easy

Actions & Reducer

Central Store