Otimização de Renderizações em React

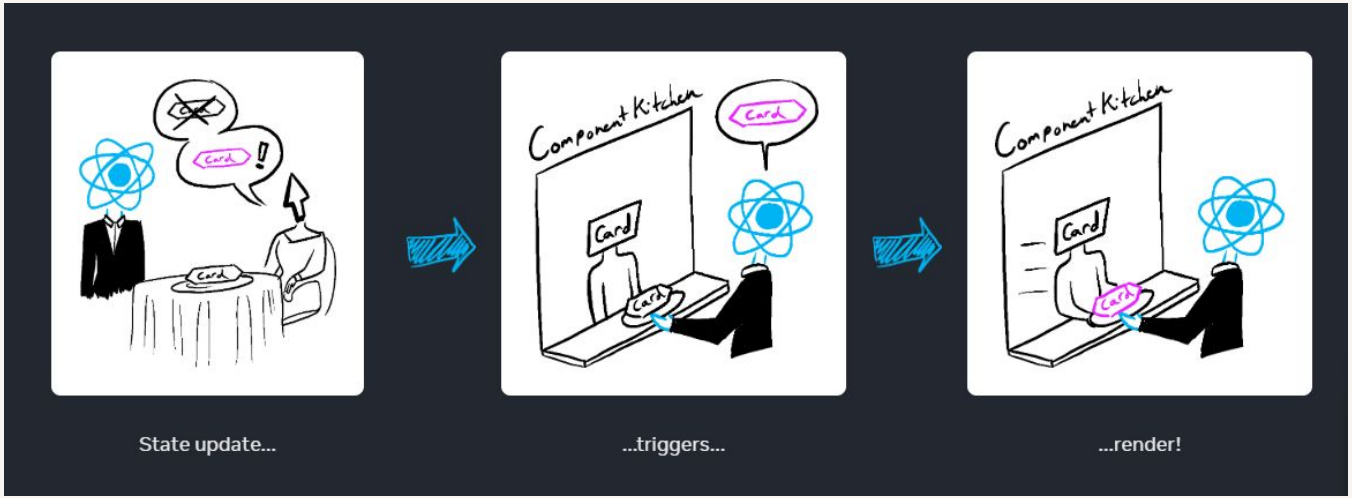
No navegador

RenderTree	Árvore com estilos aplicados
Reflow	Cálculo do Layout
Pintura	Exibição de tudo na tela

https://web.dev/articles/critical-rendering-path/render-tree-construction?hl=pt-br

O React

O React abstrai essa manipulação da DOM com uma abordagem reativa e declarativa



https://react.dev/learn/render-and-commit
https://github.com/acdlite/react-fiber-architecture
https://www.youtube.com/watch?v=i793Qm6kv3U

Quando um componente renderiza?

01 – Quando um estado muda

02 – Quando uma prop muda

03 – Quando o componente pai renderiza

Quando um componente renderiza?

01 – Quando um estado muda

02 – Quando uma prop muda

03 – Quando o componente pai renderiza

Se nada mudar, renderiza mesmo assim

Armazenar os valores anteriores de todos os componentes para comparar é muito custoso

Quando um componente renderiza?

01 – Quando um estado muda

02 – Quando uma prop muda

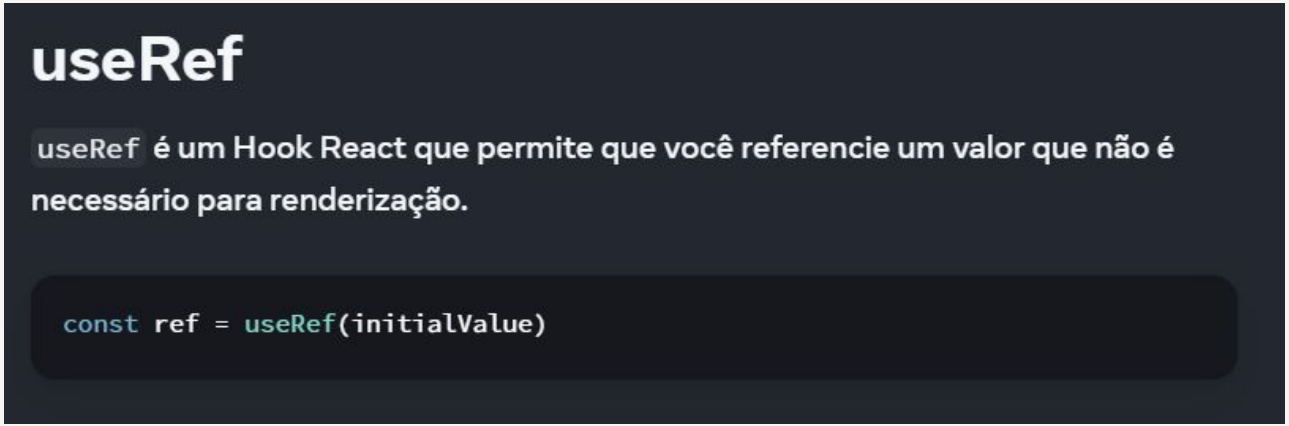
03 – Quando o componente pai renderiza

Se nada mudar, renderiza mesmo assim

Armazenar os valores anteriores de todos os componentes para comparar é muito custoso

Memo (spoiler)

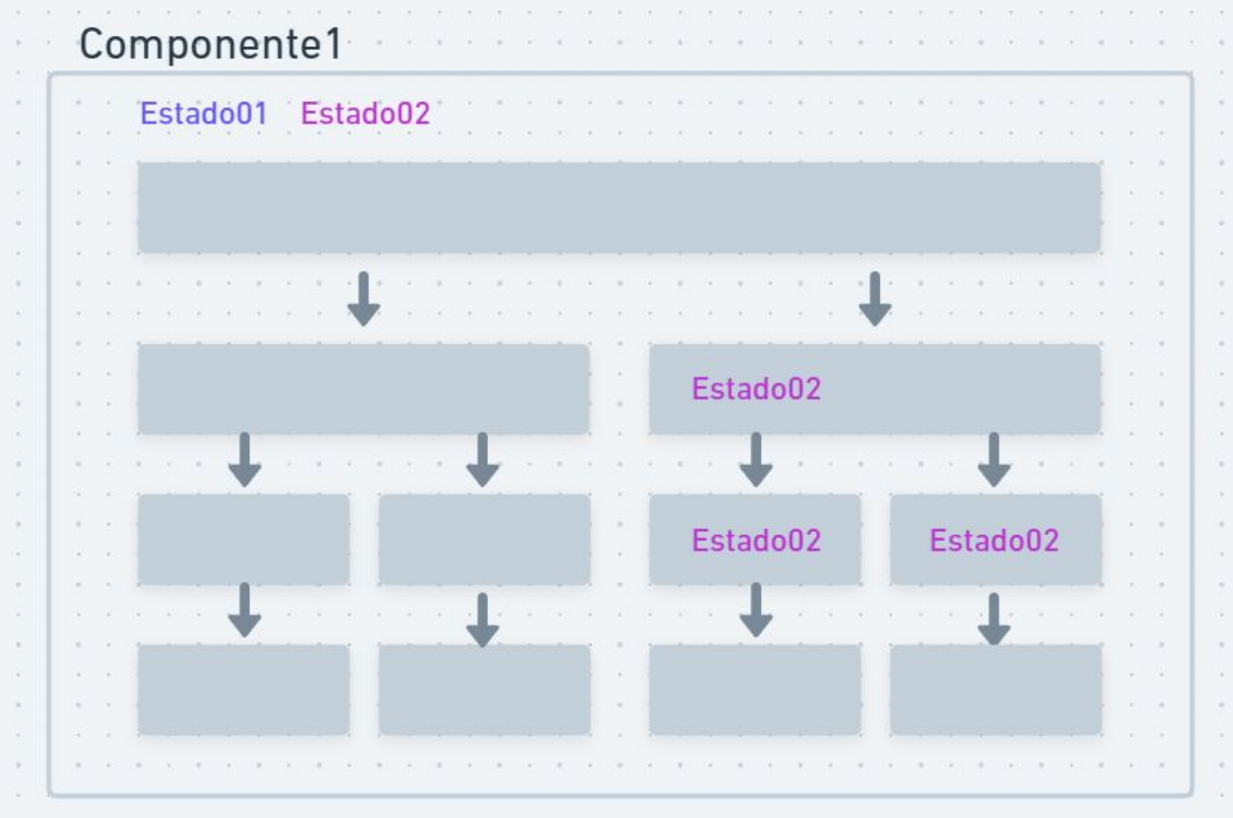
Eu preciso de um estado?



https://react.dev/reference/react/useRef

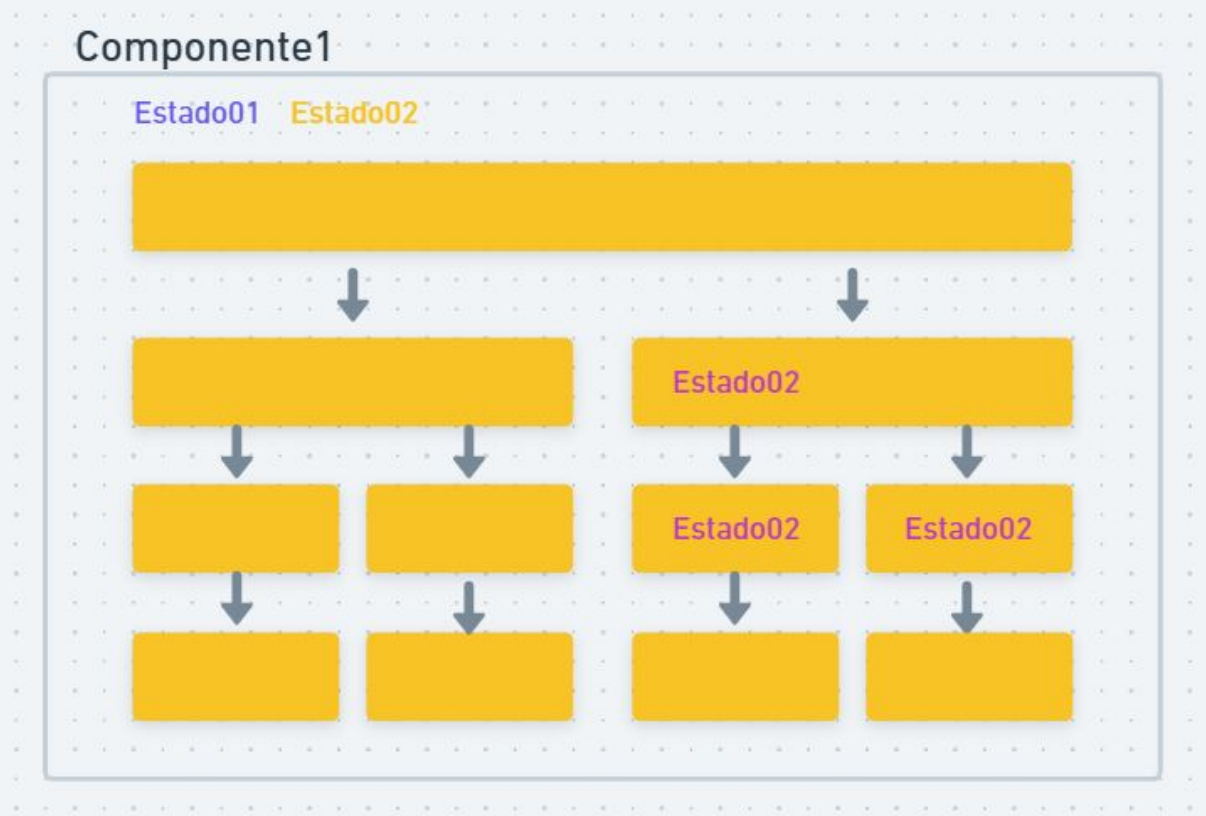
01. Descer o estado (Problema)

https://overreacted.io/before-you-memo/



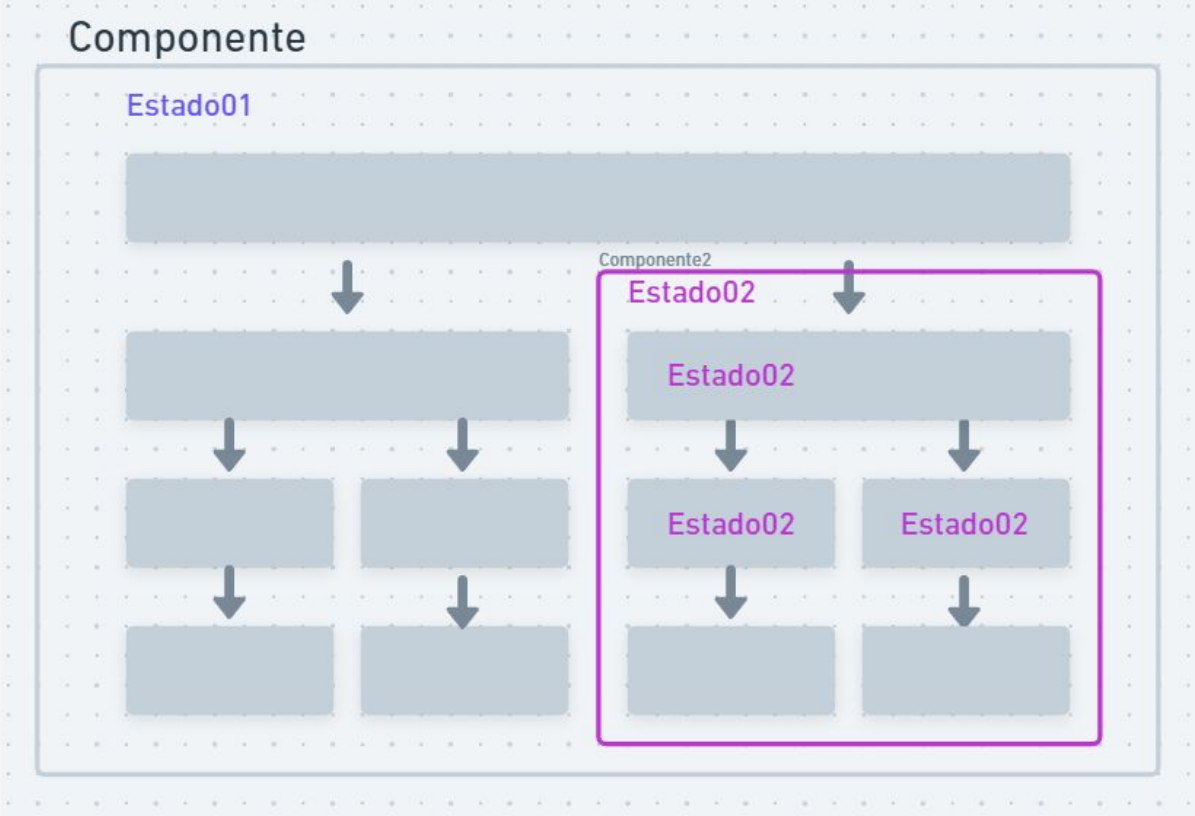
01. Descer o estado (Problema)

https://overreacted.io/before-you-memo/



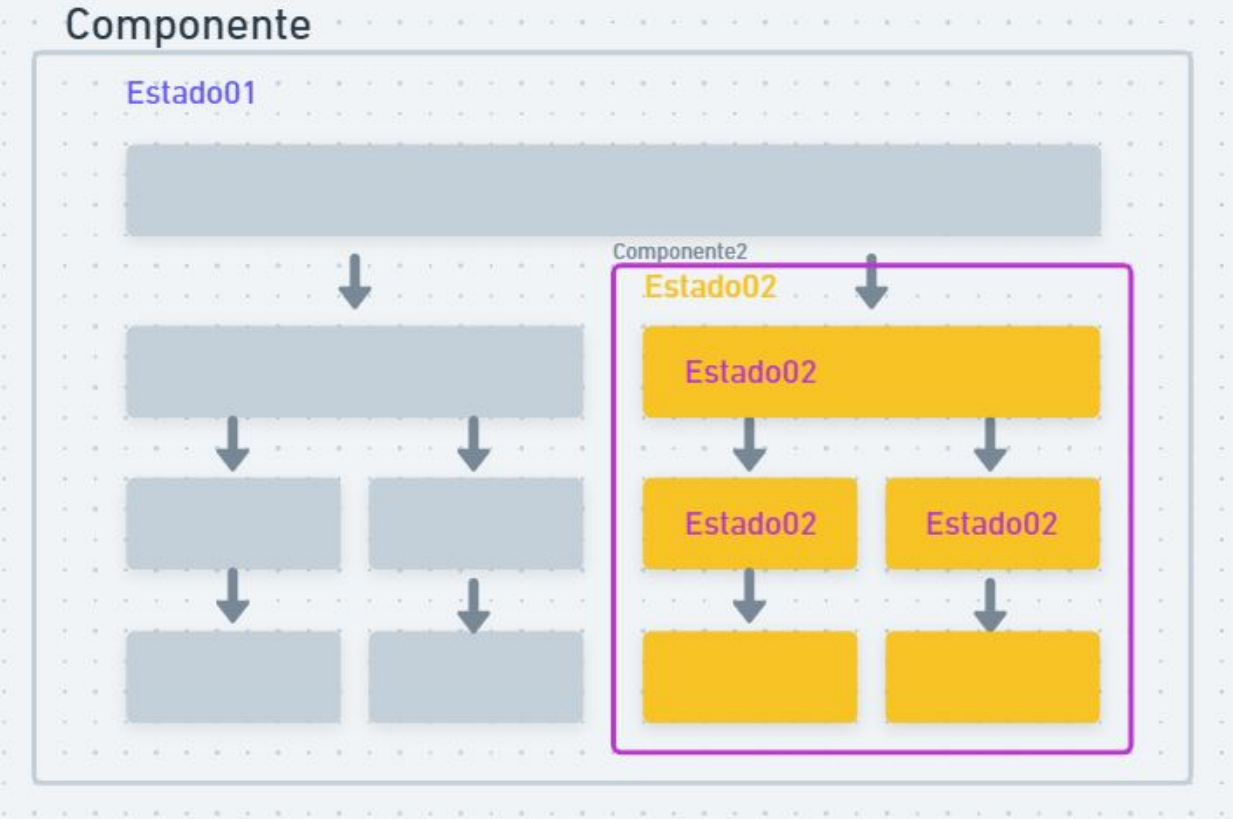
01. Descer o estado (Solução)

https://overreacted.io/before-you-memo/

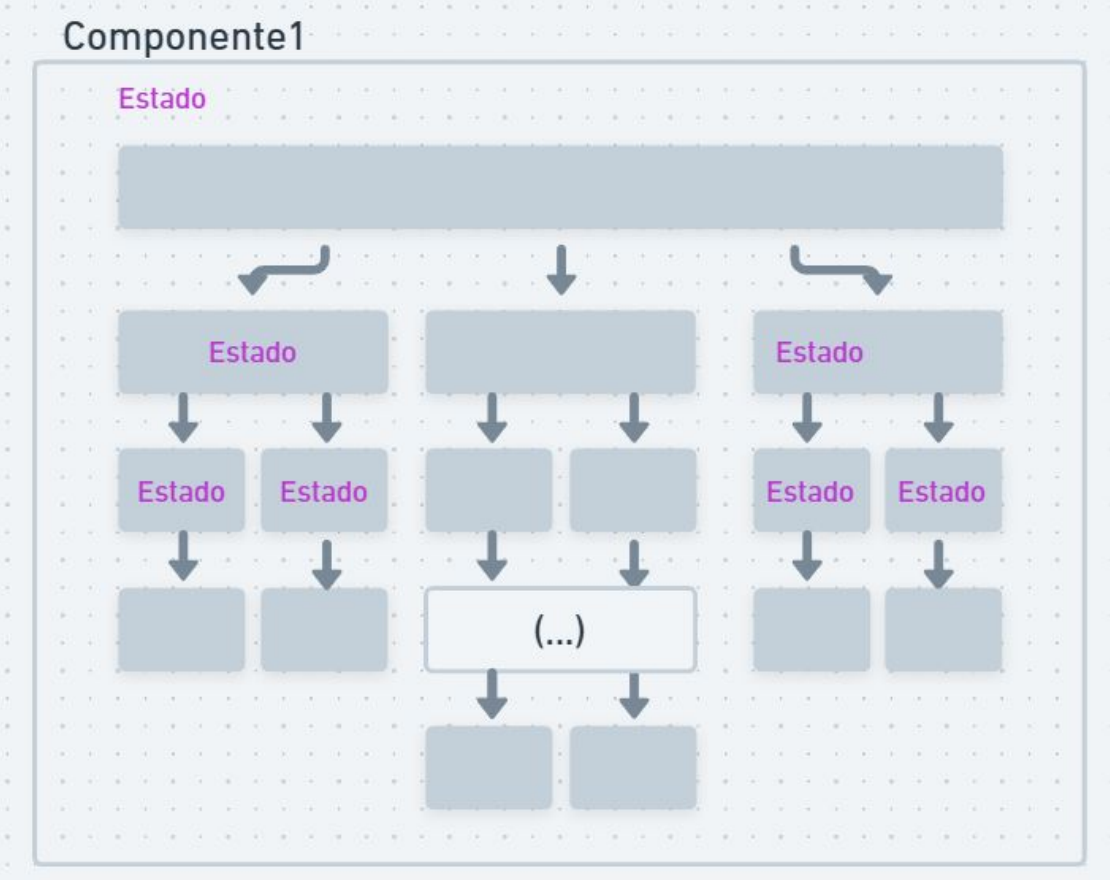


01. Descer o estado (Solução)

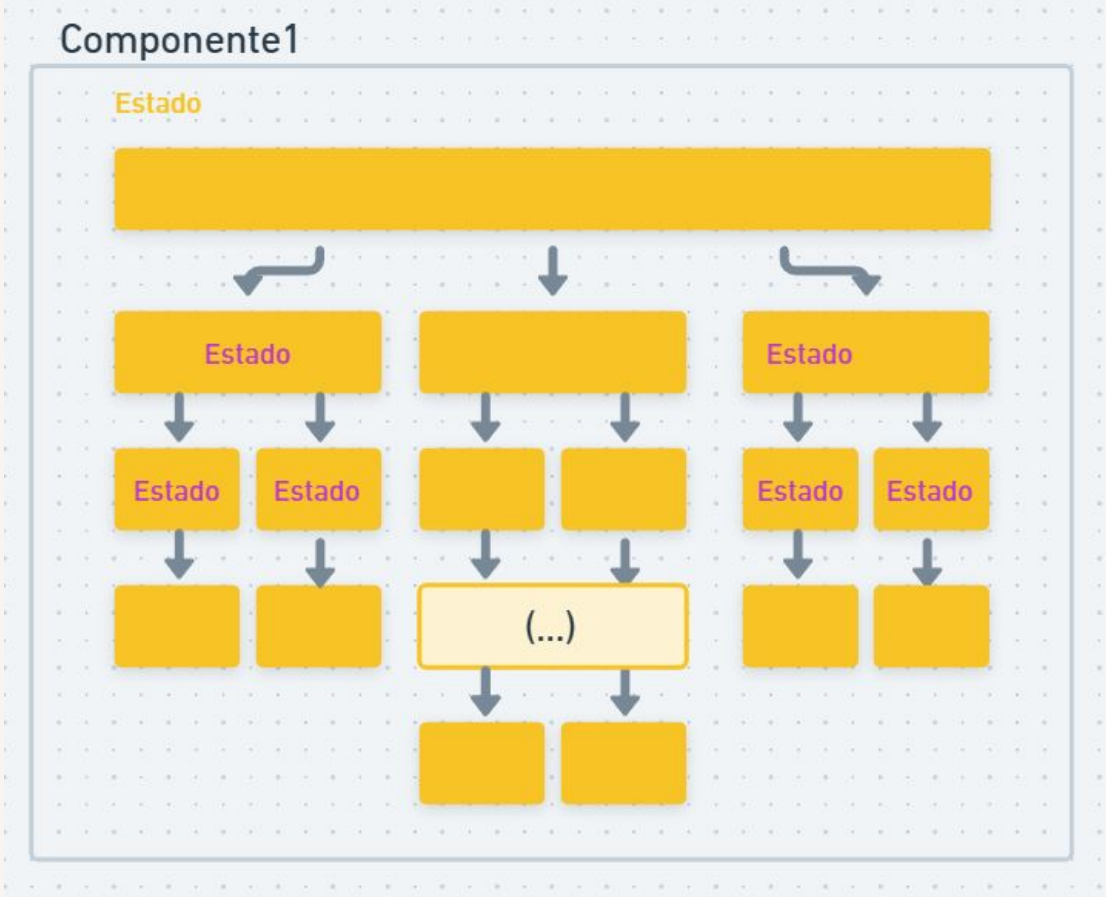
https://overreacted.io/before-you-memo/



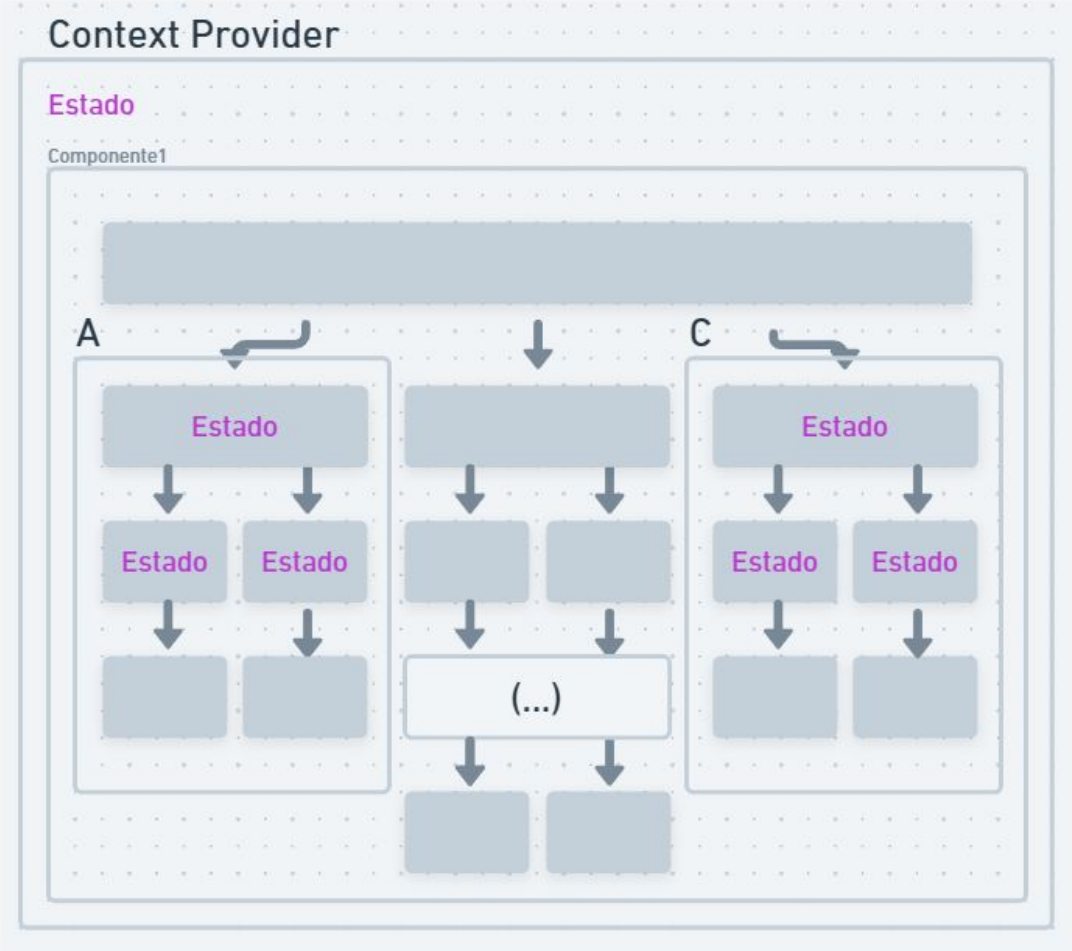
02. API de contexto (Problema)



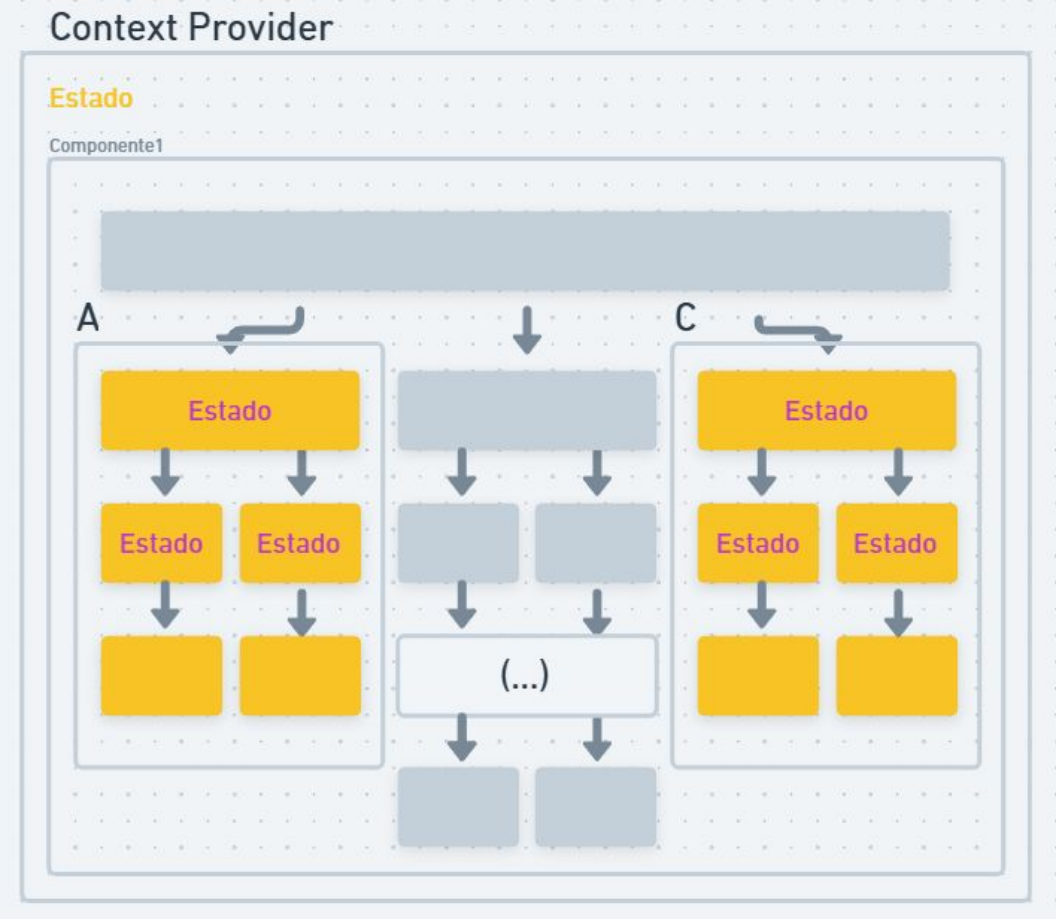
02. API de contexto (Problema)



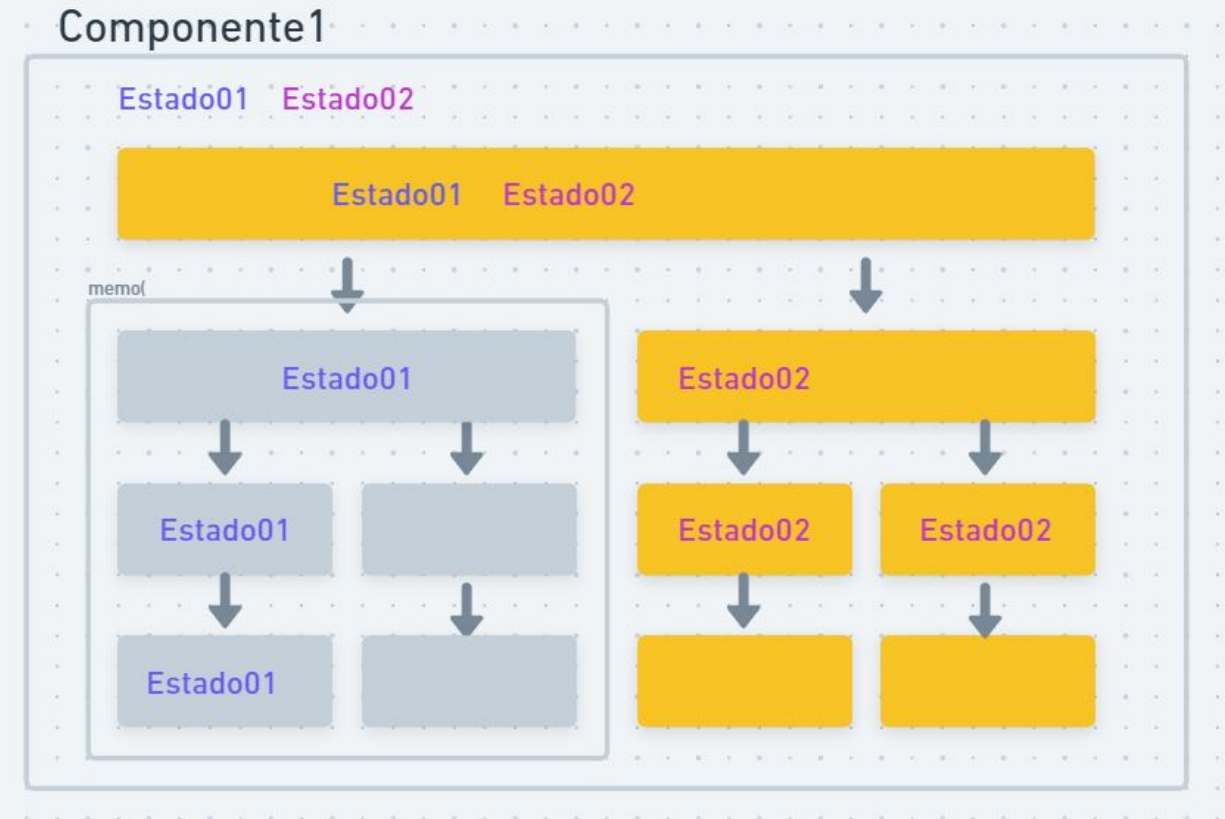
02. API de contexto (Solução)



02. API de contexto (Solução)



Memoizando componentes



Memoizando componentes

```
const ComponenteMemoizado = memo(Componente)
```

Cuidado com props que são:

- Objetos
  { a: 1 } === { a: 1 } -> false
- Arrays
  [1, 2, 3] === [1, 2, 3] -> false
- Funções
  () => {} === () => {} -> false

O memo vai comparar a referência e vai considerar que a prop mudou, mesmo sendo a mesma coisa

https://react.dev/reference/react/memo

Memoizando componentes

Prop objeto

```
{/* ao invés de */}
<Componente objeto={objeto} />
{/* prefira */}
<Componente atributo1={objeto.atributo1} atributo2={objeto.atributo2} />
```

Prop array

```
{/* ao invés de */}
<Componente lista={[1, 2, 3]} />
{/* prefira */}
{ [1, 2, 3].map((valor) => <Componente valor={valor} />) }
```

Memoizando componentes

Prop função

```
const ComponentePai = ({ dependencia1, dependencia2 }) => {
  const minhaFuncao = useCallback(() => {
    return dependencia1 + dependencia2;
  }, [dependencia1, dependencia2]);

  return (
    <Componente funcao={minhaFuncao} />
  )
}
```

Priorização

Preciso renderizar, mas quero priorizar a interação do usuário

useTransition/startTransition

Adia a atualização de um estado.

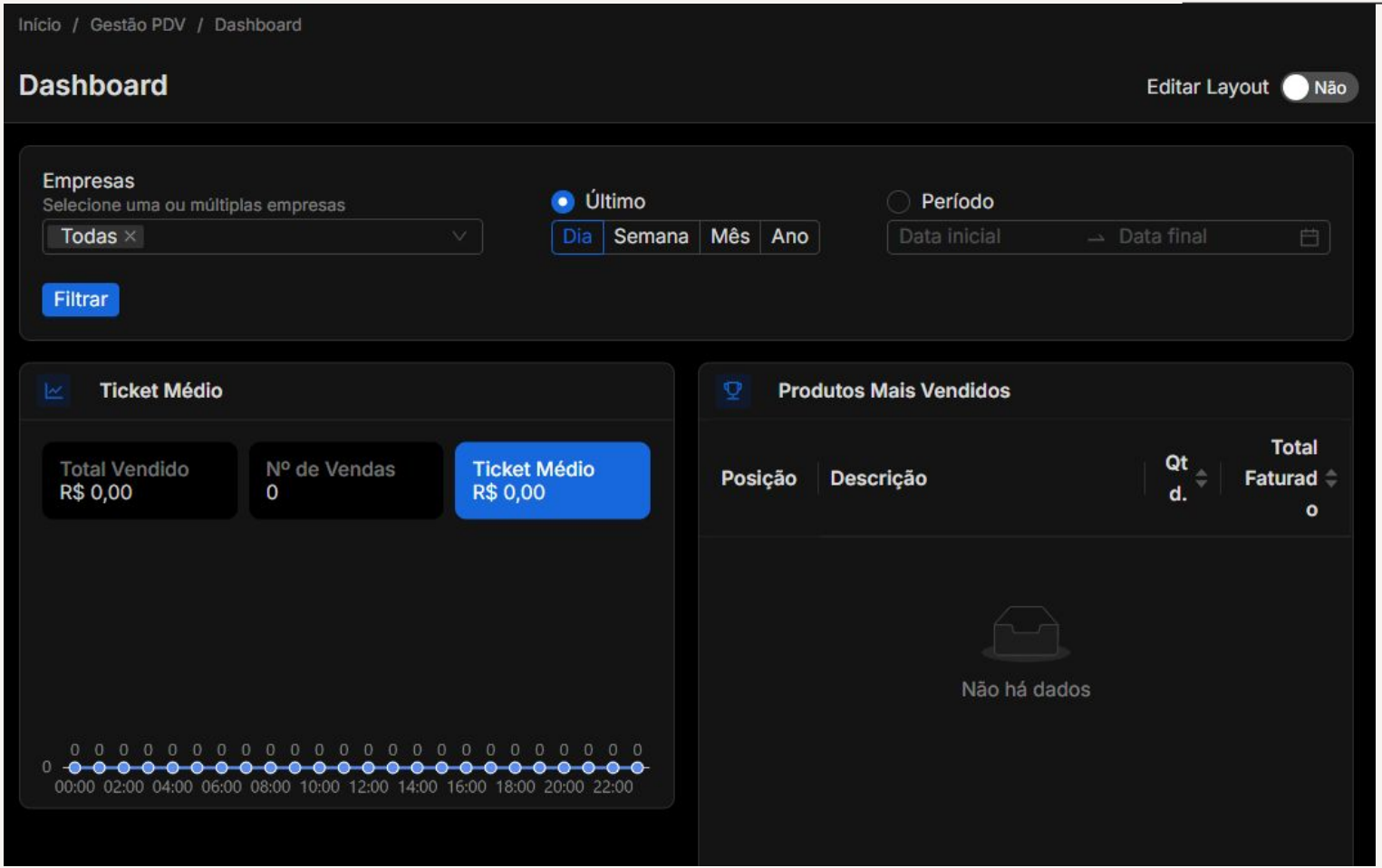
```
startTransition(() => setState(value));
```

---

useDeferredValue – Para valores

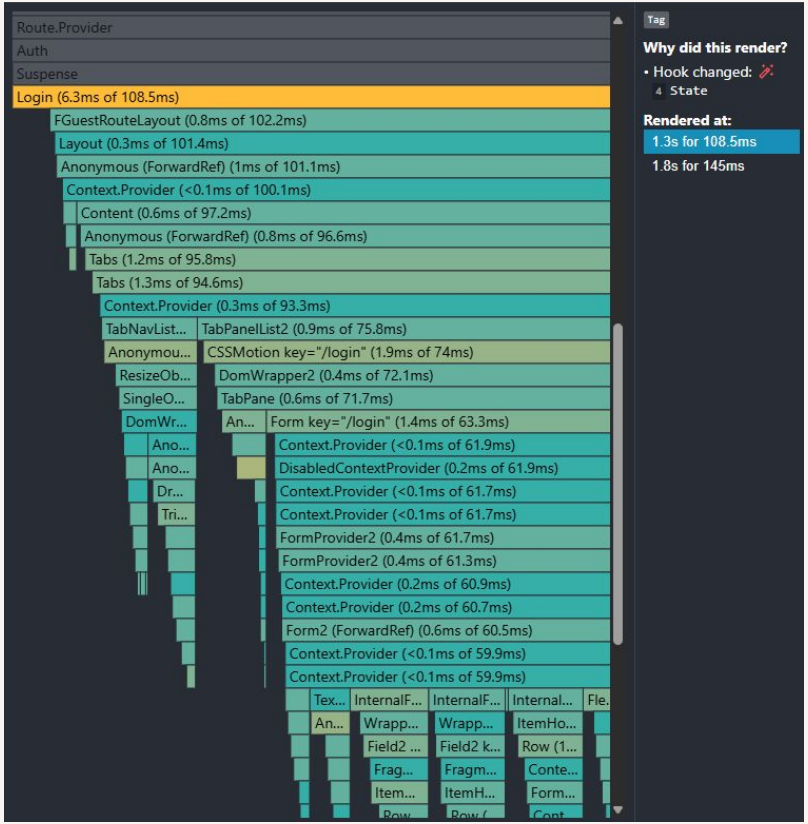
Gera um valor com a sua atualização adiada. Ideal para ser utilizado em arrays de dependências ou passado para componentes memoizados. Utilizado quando eu não tenho o “Setter” do estado (prop)

```
const deferredState = useDeferredValue(state)
```

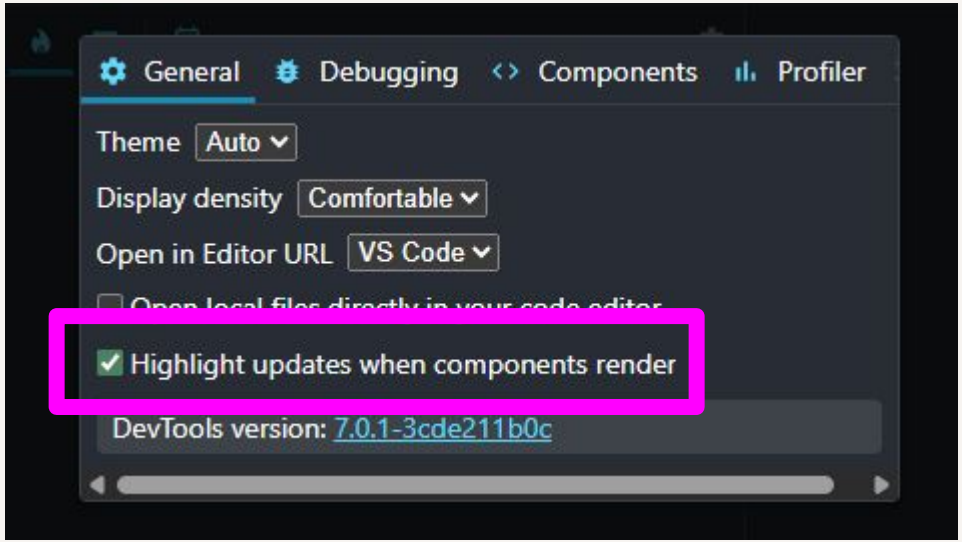

Início / Gestão PDV / Dashboard
Dashboard
Editar Layout
Não
Empresas
Selecione uma ou múltiplas empresas
Todas
Último
Dia
Semana
Mês
Ano
Período
Data inicial
Data final
Filtrar
Ticket Médio
Total Vendido
R$ 0,00
Nº de Vendas
0
Ticket Médio
R$ 0,00
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0
00:00 02:00 04:00 06:00 08:00 10:00 12:00 14:00 16:00 18:00 20:00 22:00
Produtos Mais Vendidos
Posição
Descrição
Qtd.
Total Faturado
Não há dados

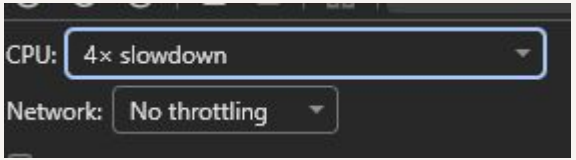
Debugging

https://react.dev/learn/react-developer-tools



Debugging


General
Debugging
Components
Profiler
Theme
Auto
Display density
Comfortable
Open in Editor URL
VS Code
Highlight updates when components render
DevTools version: 7.0.1-3cde211b0c


CPU:
4× slowdown
Network:
No throttling

Dúvidas, contribuições...?