









O que é Redux?

- Biblioteca que implementa Arquitetura Flux;
- Controle de estados globais;
- Quando utilizar o Redux?
 - Meu estado tem mais de um "dono"?
 - Meu estado é manipulado por mais componentes?
 - As ações do usuário causam efeitos colaterais nos dados?
- Exemplos: Carrinho de compras, dados do usuário, player de música, etc;

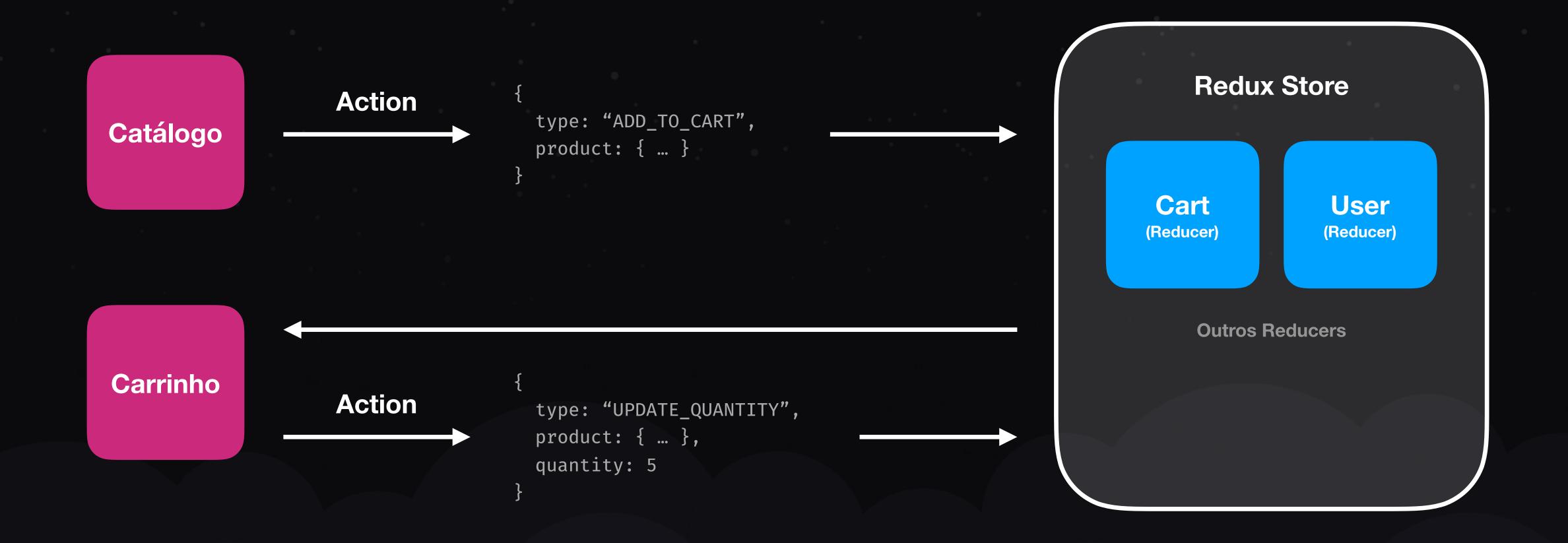








Arquitetura Flux











Princípios

- Toda action deve possuir um "type";
- O estado do Redux é o único ponto de verdade;
- Não podemos mutar o estado do Redux sem uma action;
- As actions e reducers são funções puras, ou seja, não lidam com side-effects assíncronos;
- Qualquer lógica síncrona para regras de negócio deve ficar no reducer e nunca na action;
- Nem toda aplicação precisa Redux, inicie sem ele e sinta a necessidade depois;









Exemplo do carrinho

Cart Reducer

Action

```
type: "ADD_TO_CART",
product: {
  id: 1,
  title: "Novo produto",
  price: 129.9
```

Action

```
type: "UPDATE_AMOUNT",
product: 1,
amount: 5,
```

Cart Reducer

```
id: 1,
title: "Novo produto",
price: 129.9,
amount: 1,
priceFormatted: "R$129,90"
```

Cart Reducer

```
id: 1,
title: "Novo produto",
price: 129.9,
amount: 5,
priceFormatted: "R$129,90"
```





BORA CODAR!







