

# Redux

## Управляем глобальным состоянием приложения

Кирилл Талецкий

TeachMeSkills  
12 Октября 2023

# Что такое Redux

- Это подход и библиотека для организации централизованного хранения данных на фронтенде
- Реализует простую и понятную архитектуру хранения и управления данными, которую удобно масштабировать
- Удобство достигается за счёт предсказуемости и прозрачности изменений данных

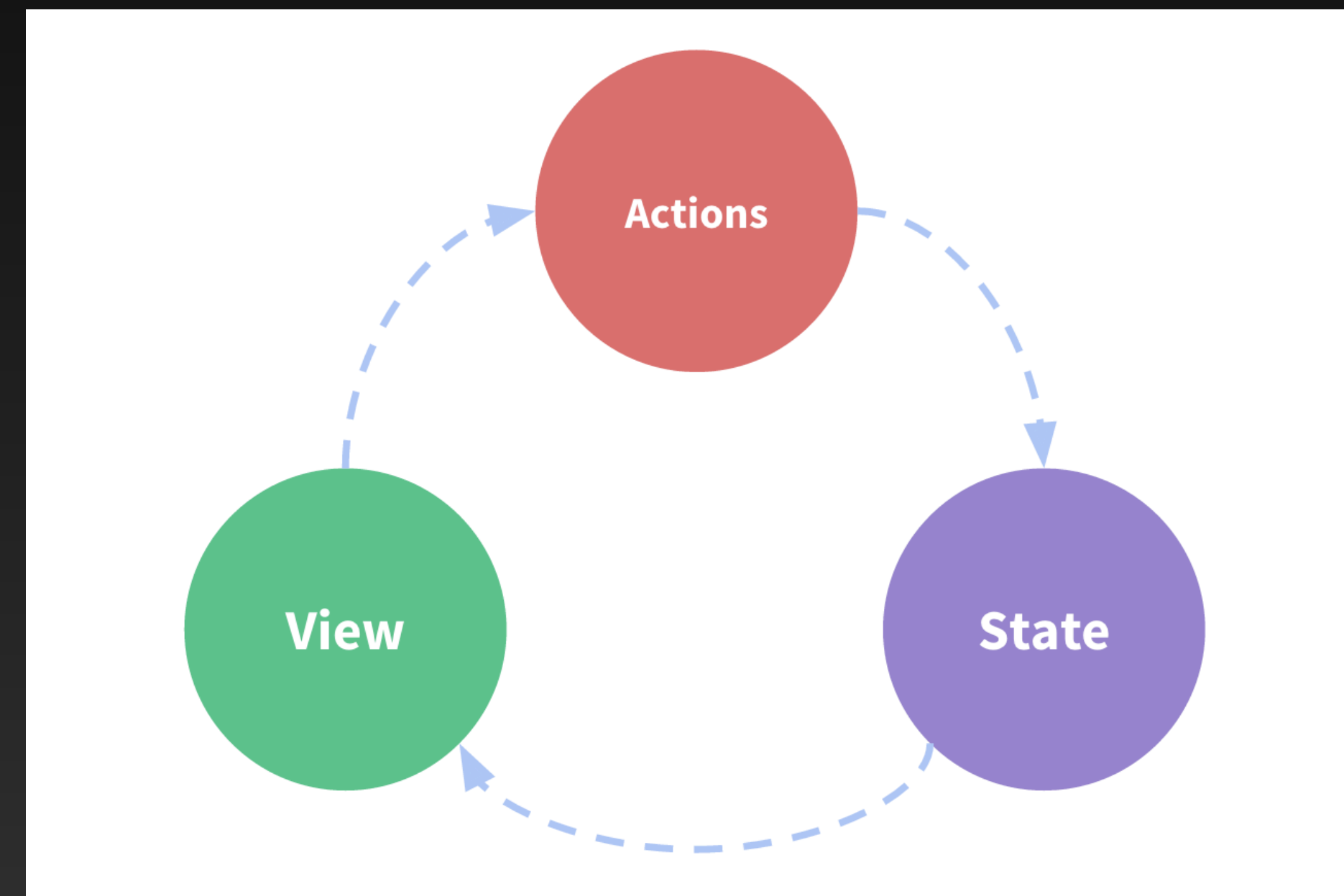
# Почему Redux?

- Redux гарантировано даёт ответы на вопросы:
  - Когда произошло изменение данных?
  - Почему изменения произошли?
  - Кто вызвал изменения?
  - Как поменялись данные?
- Это очень удобно для средних и больших проектов, оперирующих большим количеством данных на фронтенде

# Концепция

## Односторонний поток данных/управления

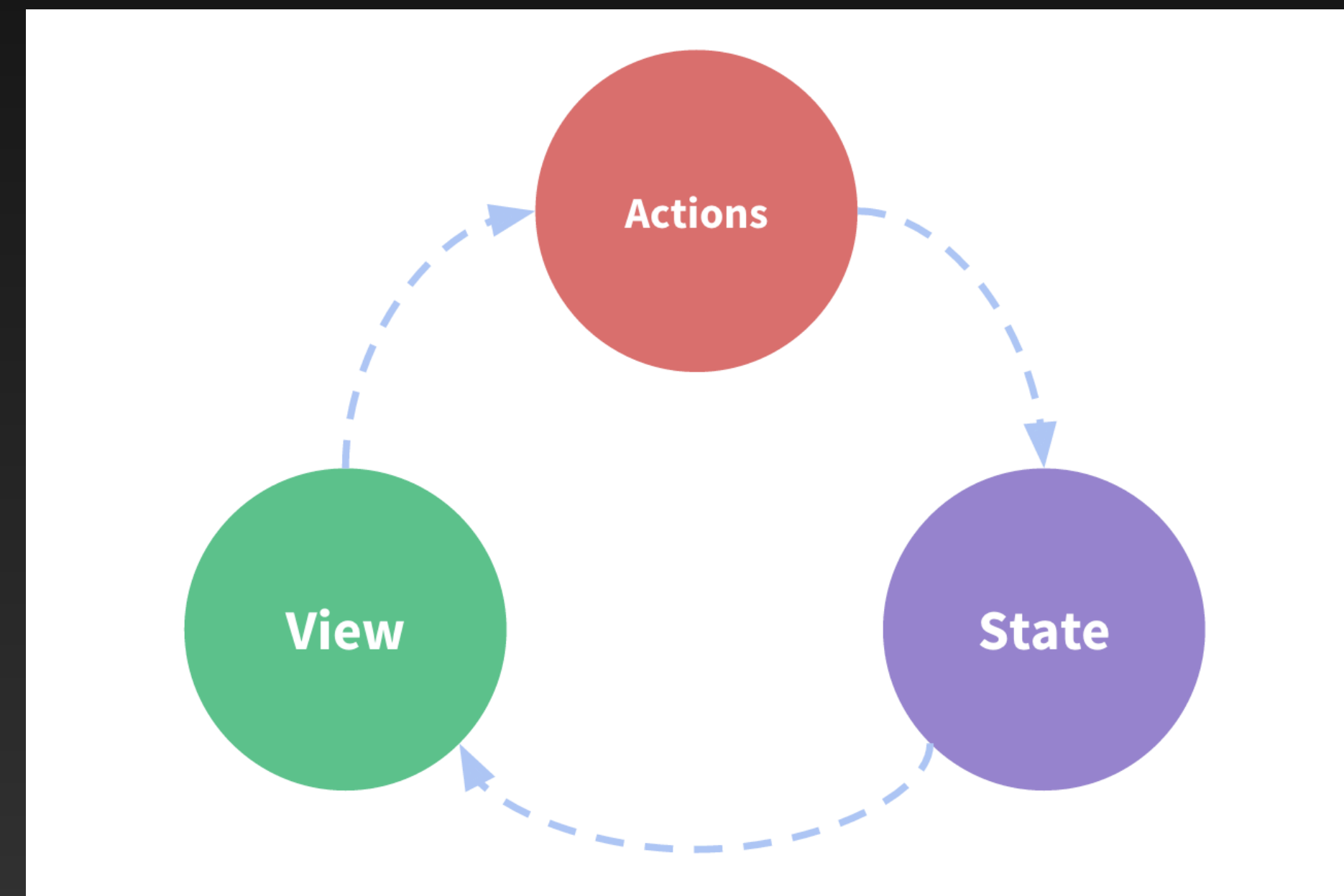
- **State** — состояние — источник истины, данные, управляющие нашим приложением
- **View** — отображение — код, описывающий как будет выглядеть приложение, в зависимости от состояния
- **Actions** — действия — события, которые происходят в приложении и вызывают изменение



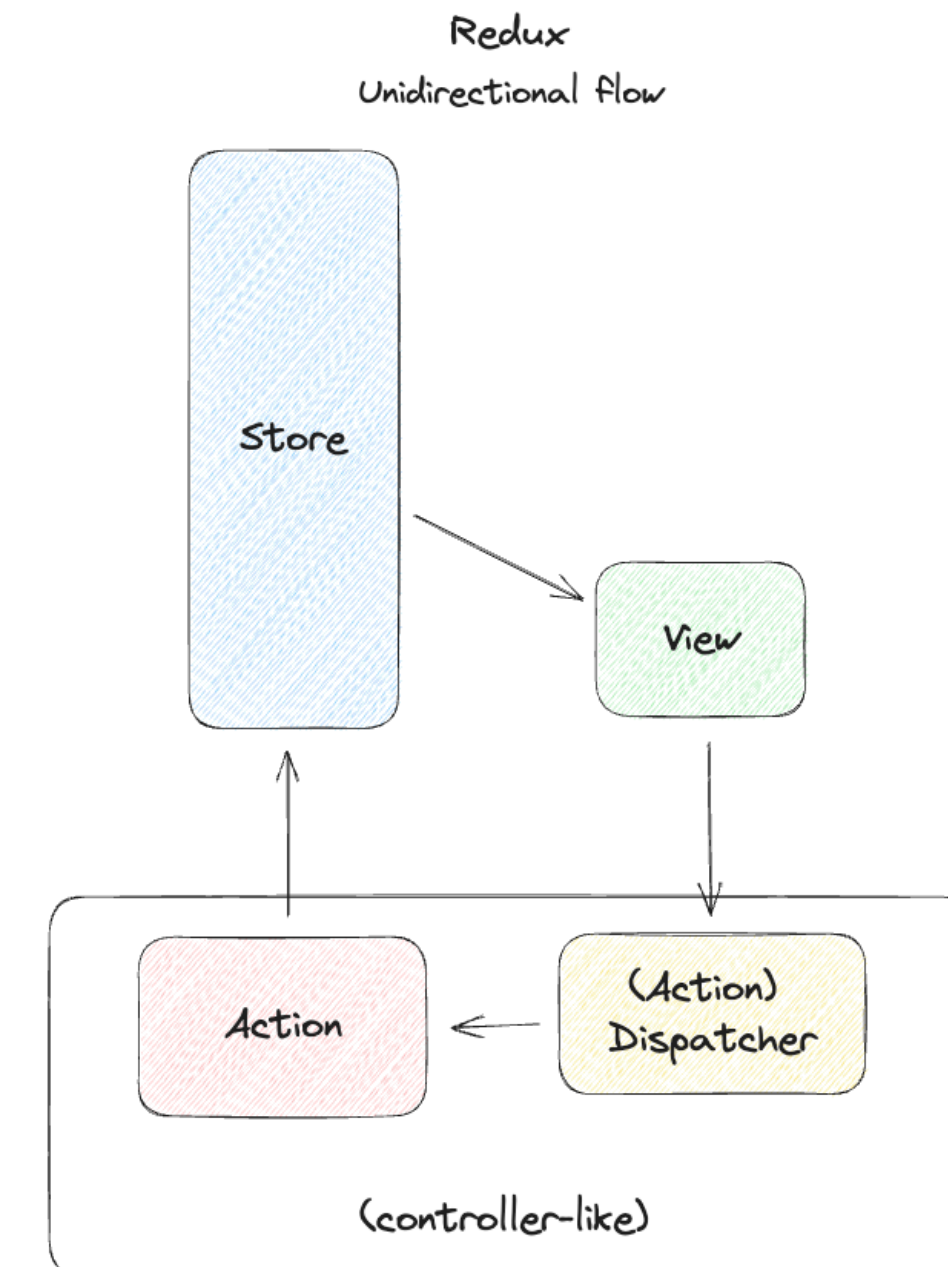
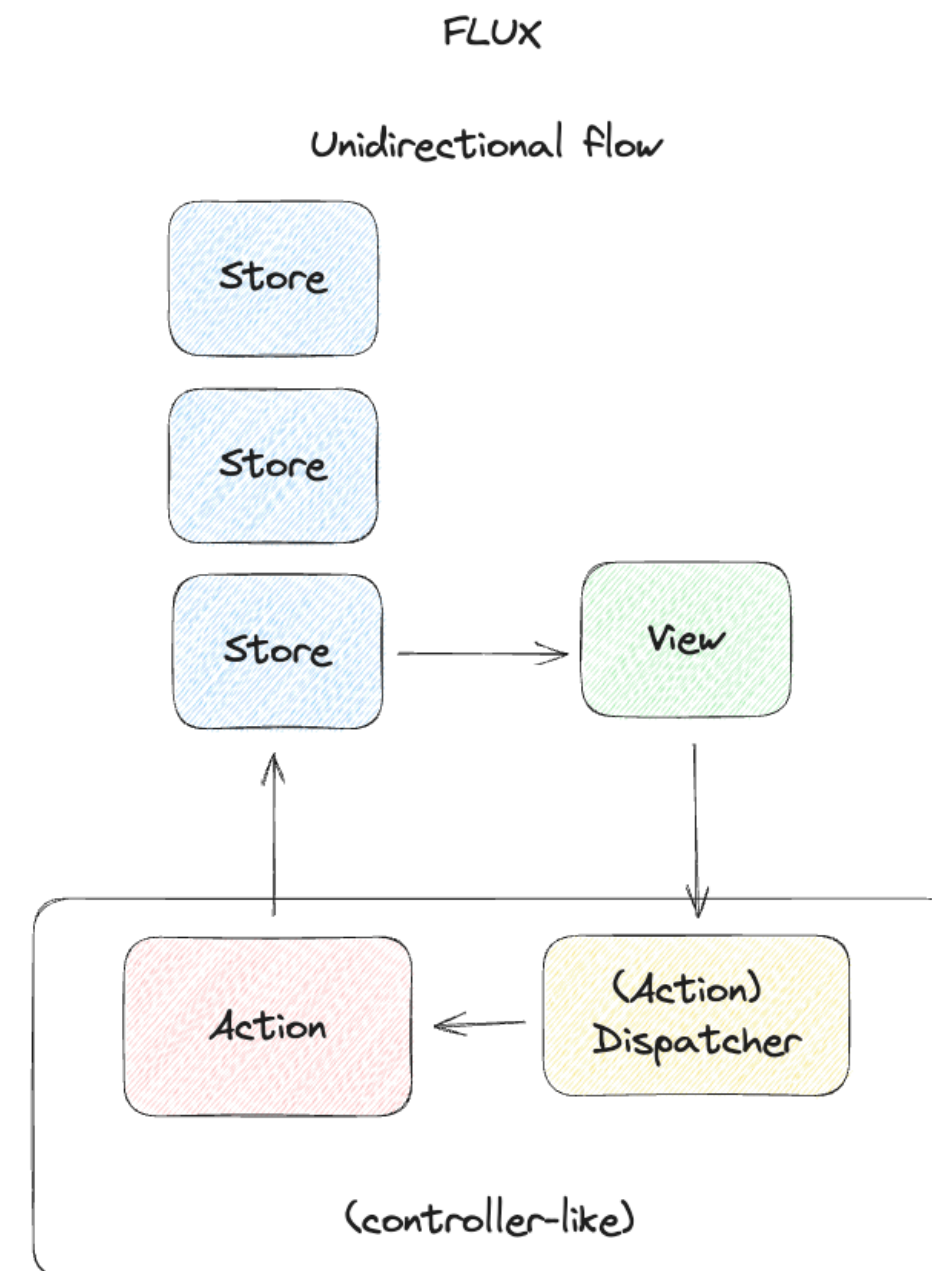
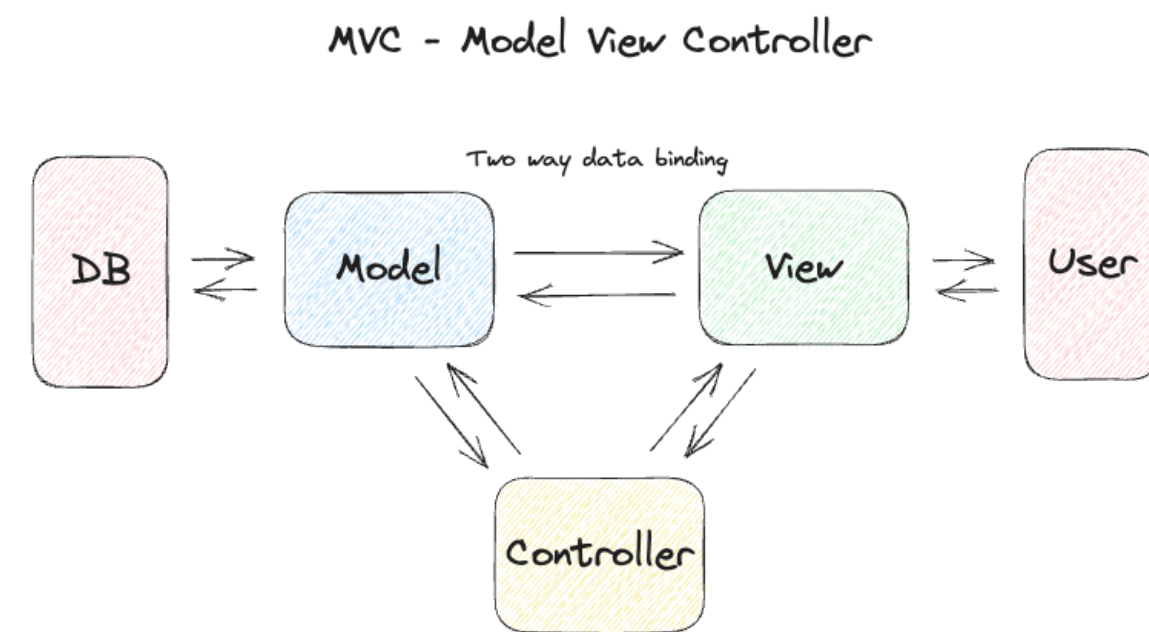
Важно: направление стрелок указывает, кто и на что может влиять.  
Мы получили простой граф зависимостей, который будет легко анализировать в больших проектах

# Компонент-счётчик как пример одностороннего потока данных/управления

```
const Counter: FC = () => {  
  // state  
  const [value, setValue] = useState(0);  
  
  // action  
  const increment = () => {  
    setValue((prev) => ++prev);  
  };  
  
  // view  
  return (  
    <h1>  
      Count: {value}  
      <button onClick={increment}>increment</button>  
    </h1>  
  );  
};
```



# MVC vs FLUX vs Redux





# Сущности Redux

State

Состояние, источник истины

Action

Сообщение, как правило, помечено типом и содержит данные которые нужно обновить/положить/удалить

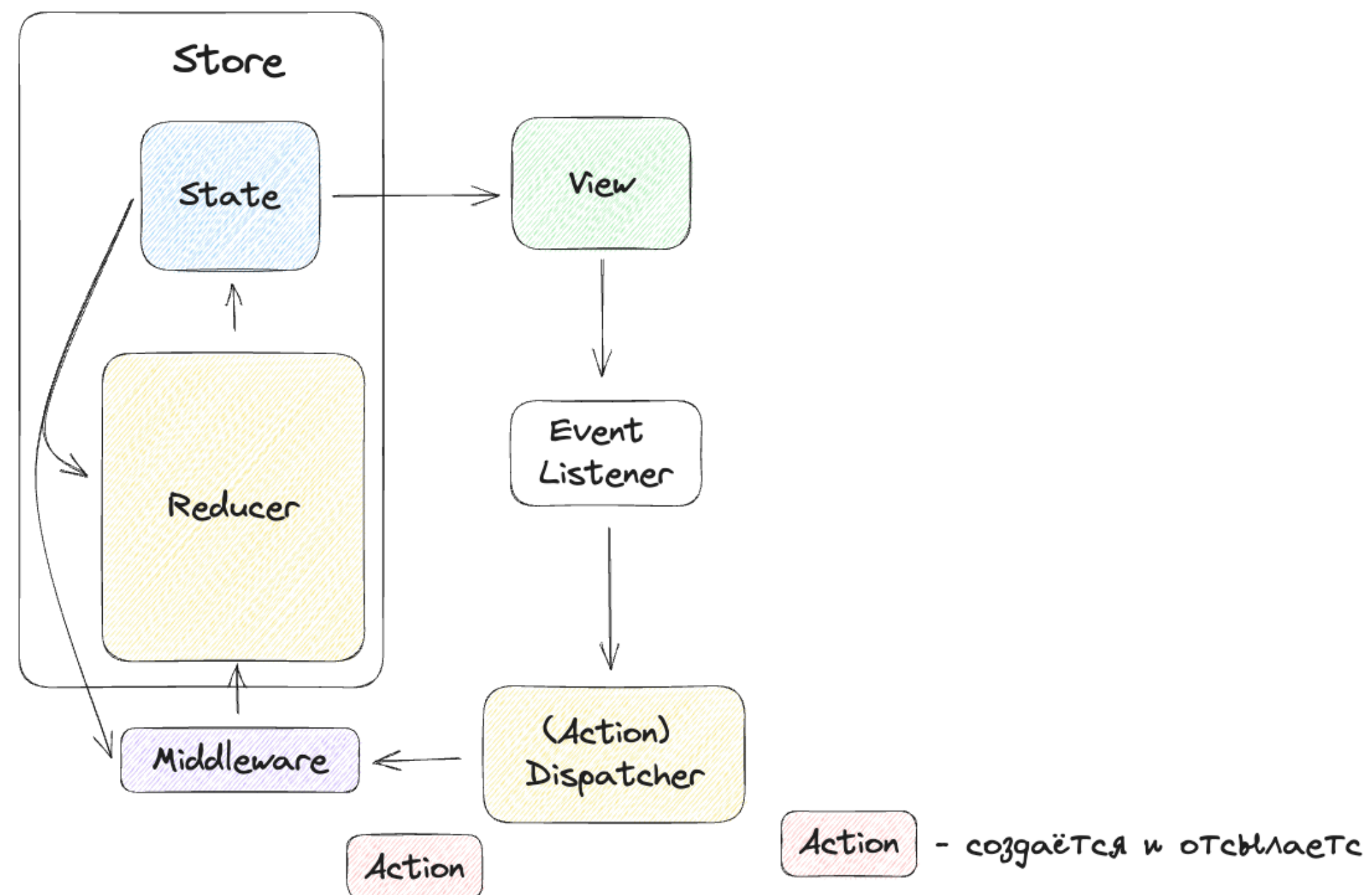
Reducer

Набор инструкций (функций) который отвечает за изменение состояния в зависимости от Action'ов

Middleware

"Посредник", функция или набор функций, которые умеют работать с Action'ами  
Умеет трансформировать/отменять/высылать новые экшены

# Устройство Redux



Action обычно выглядят так:

```
{
  type: "my-action-type",
  payload: {}
}
```

По этому полю Reducer будет различать action'ы между собой  
Опциональная полезная нагрузка, данные, используя которые Reducer обновит состояние



**Практика**