Flux introduction slide

This slide give you introduction to Flux,Its design pattern,its uses and difference between flux and redux.

By Sumit Kohli
FullStack Consultant
https://in.linkedin.com/in/sumit-kohli-234a962
PH: 8218935010

What is Flux

Flux a **unidirectional data flow pattern** to solve state management.
Flux is a design pattern intially used by facebook to solve the issue to state management and UI Updates.

Why we need Flux

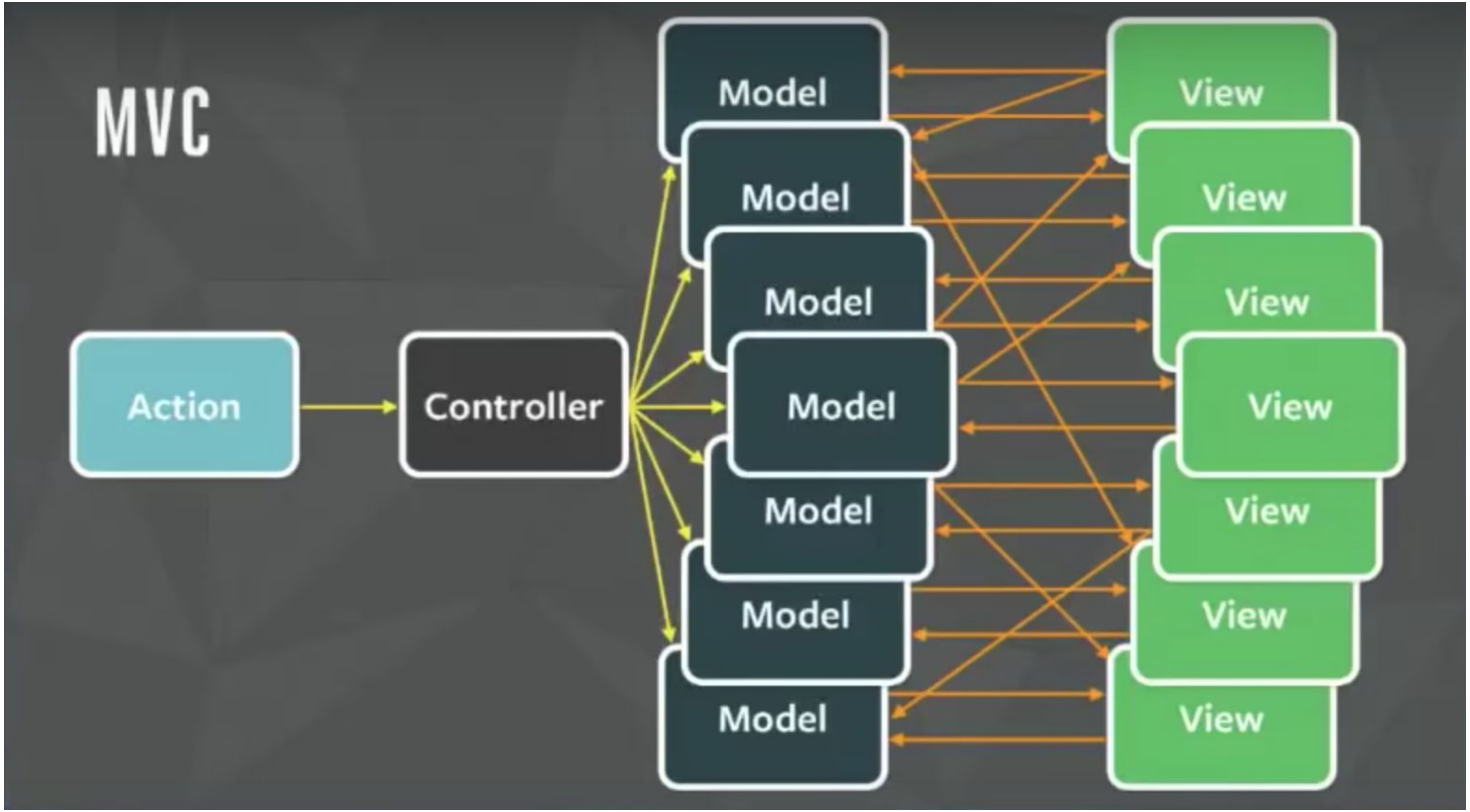
We have seen the major issue with state of components that they are local to the component.

In a large application we need to pass date information and UI updates across various components . For that we need to use a design pattern,which is Flux.

Flux Uses

Flux has been used excessively by SPA frameworks like backeboneJs, Angular and React. It can also be used by plain javascript frameworks.

Flux Design pattern



Components of Flux

View: It renders the UI. Whenever any user interaction occurs on itit fires off action. For example, if a user clicks the **Delete, VIew** button.

Action: Action handles all the events. These events are passed by the view to action. It generally used to make API calls. Once the action is done it is dispatched using the Dispatcher. The action can be something like add a Todo, Delete a Todo.

Ex:

```
{
        actionType: “Add_TODO,
  data: {
    title: "Make Report",
  }
}
```

Component of Flux

Dispatcher: this is the central hub . It dispatches the payload from Actions to Store.

Store: Store holds the app state. An application can have one or many app stores. Stores get updated based on action.

Flux Benfits

1. The code becomes quite clear and easy to understand.
2. Scalable apps can be built.
3. Clear Data Flow
4. Unit Testing can be easily done due to clear data flow across components

Difference between flux and redux

Redux is advanced version of flux and has widely replaced flux.
In Flux we can have multiple Stores.
In Redux we have single store.

Flux stores handle the business logic.
Redux Reducers handle the business Logic

Thanks You

Thanks

Sumit Kohli