

Working with Data



Samer Buna

@samerbuna | ReactjsCamp.com

MongoDB



Local or Hosted

MongoLab

Collections and documents

MongoDB Node.js driver

API endpoint

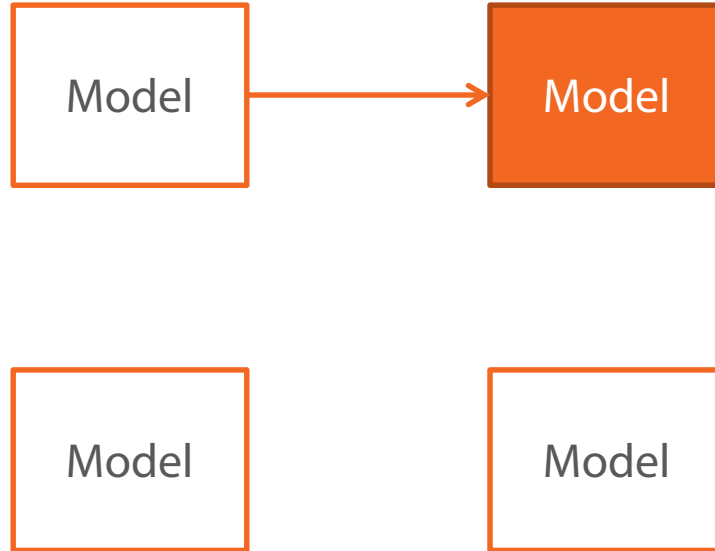


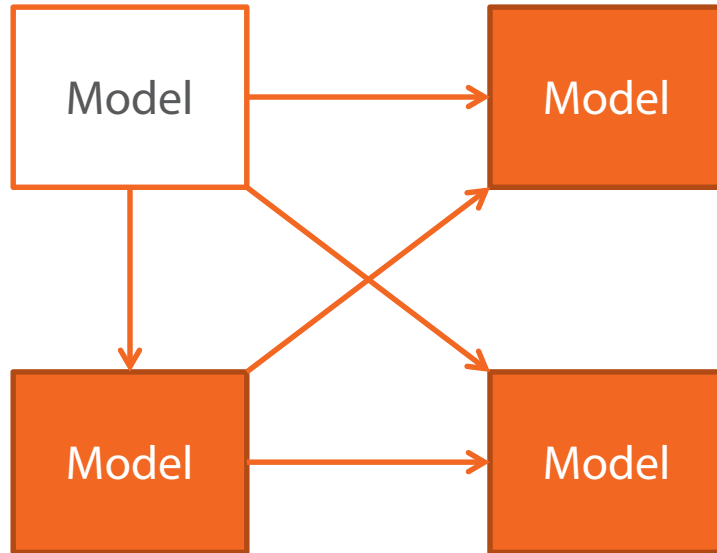
Model

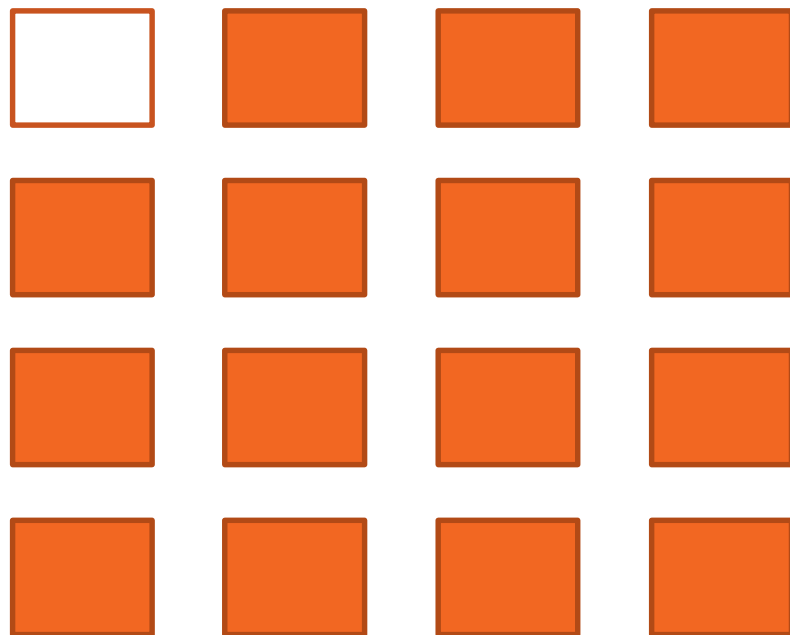
Model

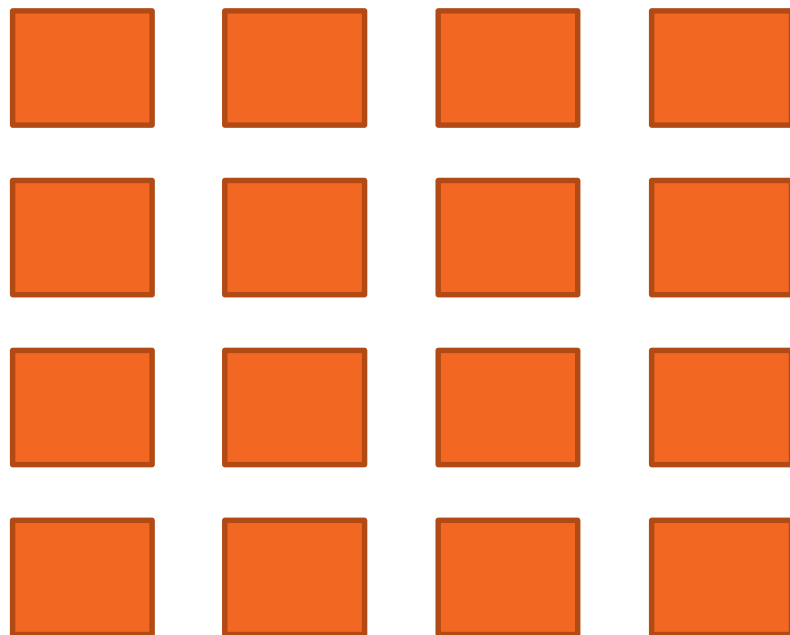
Model

Model



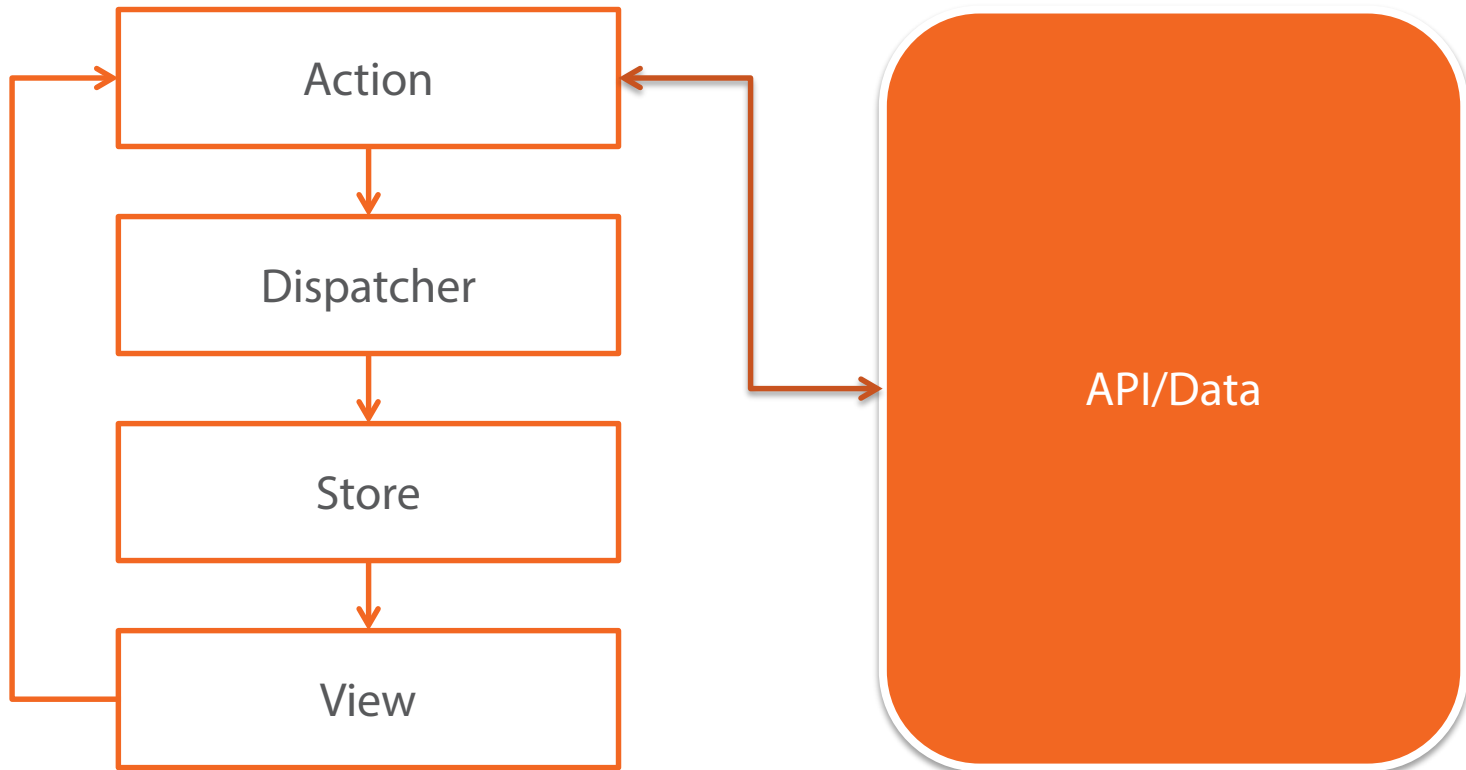






“Tie your hands to free your mind”

– Jessica Kerr



Actions

Objects/Data

“type” property: How to use the action

Action Creators: Methods in a module

“API” of the application

```
import AppDispatcher from "..."  
import ActionTypes from "... "  
  
export default {  
  receiveData(data) {  
    AppDispatcher.dispatch({  
      actionType: ActionTypes.RECEIVE_DATA,  
      data  
    })  
  },  
}
```



Dispatcher

Registry of callbacks

Fixed, singleton

Payload is an action

`Flux.Dispatcher()`

`dispatch()`, `register()`, `waitFor()`

```
// AppDispatcher.js  
import Flux from 'flux';  
  
// Export a singleton  
export default new Flux.Dispatcher();
```



Stores

Programming happens here

Each store is a singleton

Totally in control of themselves

Manage application state

Private variable/methods



Stores

Express “interest” in dispatched actions

One way to get data to a store

No setters, only getters

Emit an event when changed


```
let _users = [];  
let _followedIds = [];  
  
class UserEventEmitter extends AppEventEmitter {  
  getAll() {  
    return _users.map(user => {  
      user.following =  
        _followedIds.indexOf(user.id) >= 0;  
      return user;  
    });  
  }  
}  
  
let UserStore = new UserEventEmitter();
```

```
AppDispatcher.register(action => {  
  switch(action.actionType) {  
    case ActionTypes.RECEIVE_USERS:  
      _users = action.rawUsers;  
      UserStore.emit('change');  
      break;  
    case ActionTypes.RECEIVE_FOLLOWER:  
      _followedIds.push(  
        action.rawFollower.user_id);  
      UserStore.emit('change');  
      break;  
    default:  
      // do nothing  
  }  
});
```



Views

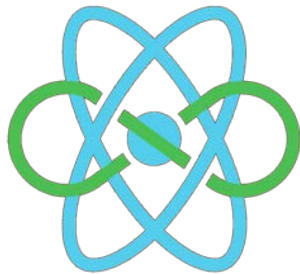
Tree of React components

Controller Views

On change, query stores for new data

React re-renders

Summary



MVC

MongoDB

The Flux pattern

Reactive updates