REACT JS

Content

- Introduction React JS and JSX
- Getting at the core - Virtual DOM (What is virtual DOM? How it works? Real DOM vs React Virtual DOM)
- Concepts - Components and its lifecycle, State, props, event handling, routing
- Architecturing our App
- Demonstration - how to create youtube like video streaming app
- Bonus - Few debugging tricks

WHAT IS REACT?

- A JavaScript Library For Building User Interfaces
- Renders your UI and responds to events.
- It also uses the concept called Virtual DOM, creates an in-memory data structure cache, enumerates the resulting differences, and then updates the browser’s displayed DOM efficiently.
- One of the unique features of React.js is not only it can perform on the client side, but it can also be rendered on the server side, and they can work together interoperably.

WHAT IS REACT?

Angular has

❏ modules
❏ controllers
❏ directives
❏ scopes
❏ templating
❏ linking functions
❏ filters
❏ dependencyinjection

WHAT IS REACT?

~~Angular~~ **React** has **JUST COMPONENT**

❑ modules

❑ controllers

❑ directives

❑ scopes

❑ templating

❑ linking functions

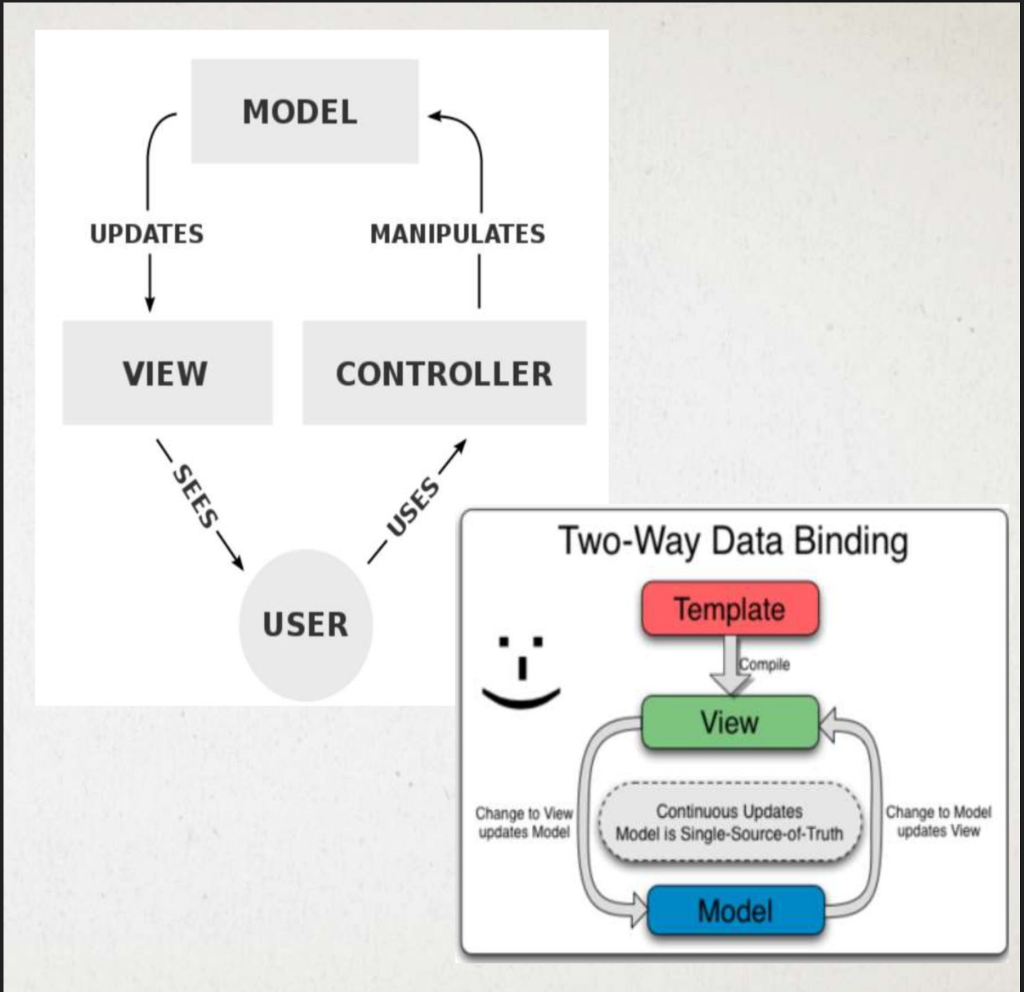
❑ filters

❑ dependencyinjection

What is REACT

#2 Single Source of Truth

MVC proposes that your **Model is the single source of truth**— all state lives there. Views are derived from the Model, and must be kept in sync. When the Model changes, so does the View.



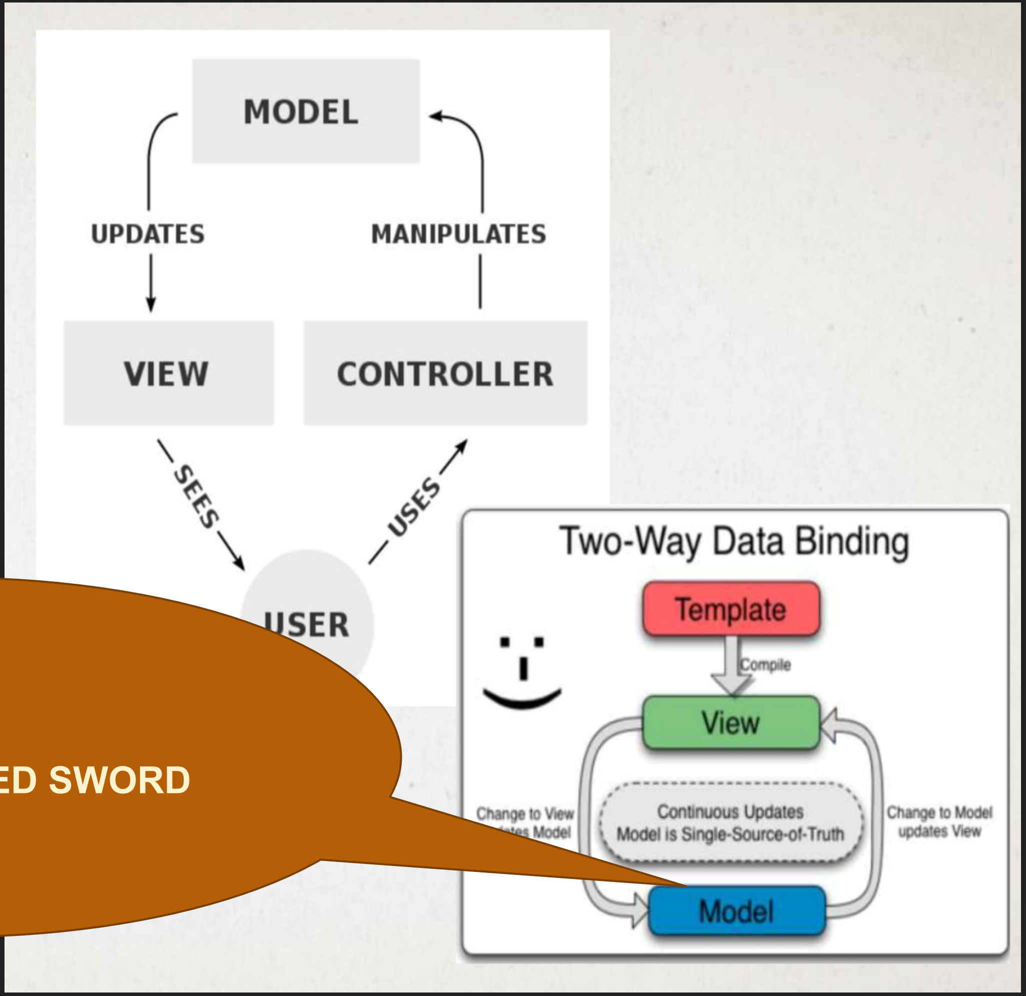
What is REACT

#2 Single Source of Truth

MVC proposes that your **Model is the single source of truth**— all state lives there. Views are derived from the Model, and must be kept in sync. Wh

changes, so d

Angular -
I AM DOUBLE EDGED SWORD



What is REACT

#2 Single Sourc Truth

MVC proposes t Model is

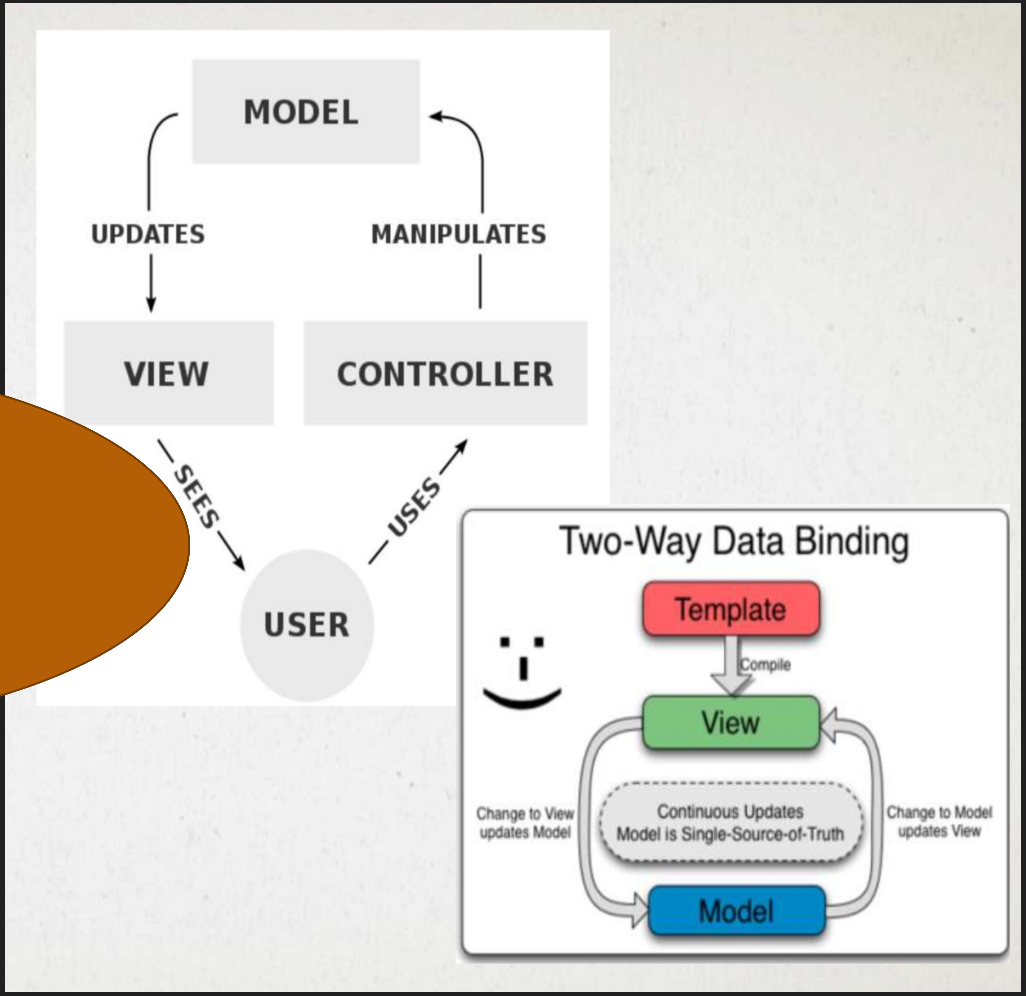
the si

s

be

changes,

Only render when state changes

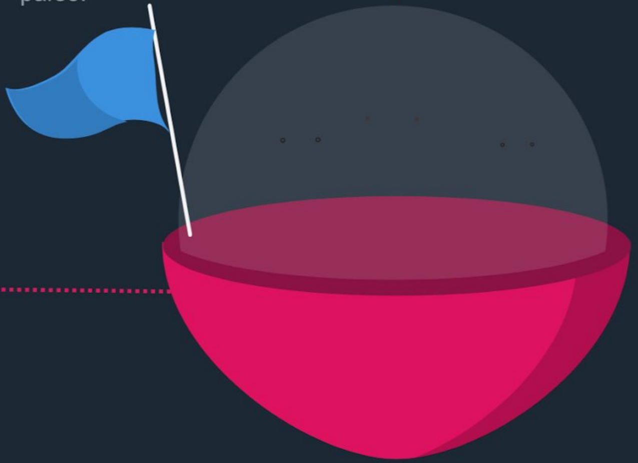


JSX (JavaScript XML)

Allows us to write HTML like syntax which gets transformed to lightweight JavaScript objects. The syntax is intended to be used by pre-processors (i.e., transpilers like Babel) to transform HTML-like text found in JavaScript files into standard JavaScript objects that a JavaScript engine will parse.

Virtual DOM

A JavaScript representation of the actual DOM. It's better to think of the virtual DOM as React's local and simplified copy of the HTML DOM. It allows React to do its computations within this abstract world and skip the "real" DOM operations, often slow and browser-specific.



Components

React is Component Based. You can build encapsulated components that manage their own state, then compose them to make complex user interfaces.

```
var React = require("react");
var ReactDOM = require("react-dom");

require("./index.css");

class App extends React.Component {
   render: function() {
         return (
                  <div>
                           Hello World!
                  </div>
         )
   }
}

ReactDOM.render(
<App />,
document.getElementById('app'));
```

DOM vs Virtual DOM

DOM (Document Object Model)

The WHAT Part

Document Object Model is a cross-platform and language-independent application programming interface that treats an HTML, XHTML, or XML document as a tree structure wherein each node is an object representing a part of the document

DOM is an abstraction of a structured text. If the structured text is an HTML code, then the DOM will be simply called HTML DOM. Elements of that HTML become nodes in the DOM.

Is the DOM really SLOW?

People often throw around the statement “The DOM is slow”. This isn’t exactly true. The DOM is fast. Adding and removing DOM nodes doesn’t take much more than setting a property on a JavaScript object. It’s a simple operation.

Let’s test it out.
Ten THOUSAND divs in about 200 milliseconds.

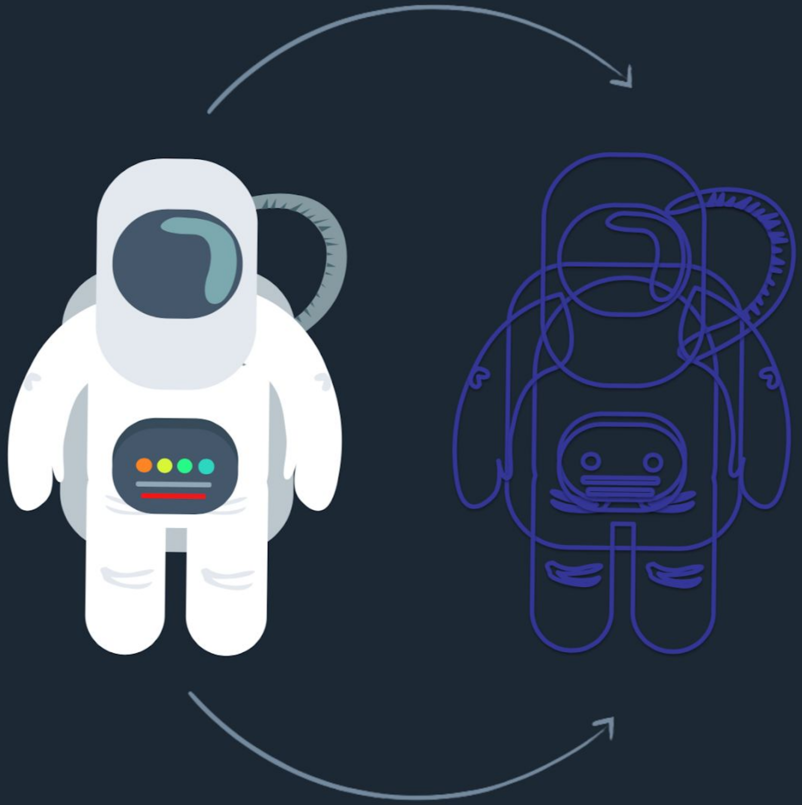
What is slow with DOM, however, is the layout that browsers have to do whenever the DOM changes. Every time the DOM changes, browser need to recalculate the CSS, do layout, and repaint the web page. This is what takes time.

The Blueprint

Virtual DOM

Virtual DOM is in-memory representation of Real DOM. It is lightweight JavaScript object which is copy of Real DOM.

The Virtual DOM is an abstraction of the HTML DOM. It is lightweight and detached from the browser-specific implementation details. Since the DOM itself was already an abstraction, the virtual DOM is, in fact, an abstraction of an abstraction.



What Makes VDOM Faster?

Efficient **diff** Algorithm

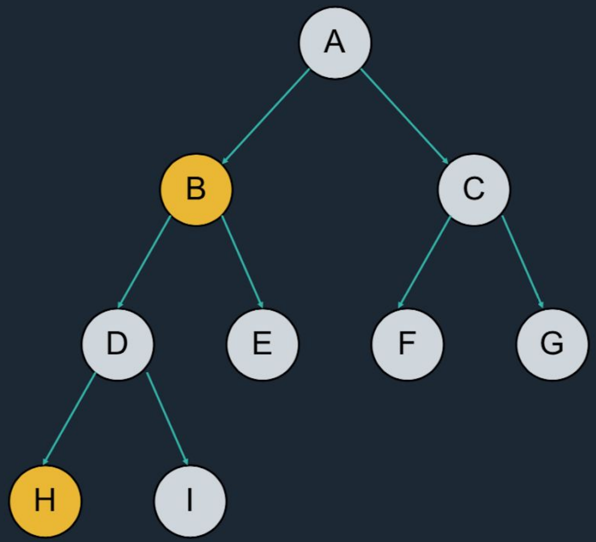
Batched Update Operations

Efficient Update of Sub-tree

Change Detection is done using Observables

Instead of Dirty Checking

Diff Algorithm – Traversing VDOM



Breadth First Search

ReactJS traverses the tree using Breadth First Search algorithm

States of element B and H have changed. So when using BFS, ReactJS reached element B it will by default re-render the element H. This is the reason to use BFS for tree traversal.

Batch Update

ReactJS using the diff algorithm to find the minimum number of steps to update the Real DOM.

Once it has these steps, **it executes all the steps in one event loop without involving the steps to repaint the Real DOM**. Thus, if there are more element which gets updated ReactJS will wait for the event loop to finish then, in bulk will updated the real DOM with all the updated elements.

Once all the steps are executed, **React will repaint the Real DOM**. This means during the event loop, there is exactly one time when the Real DOM is being painted. Thus all the layout process will run only on time for updating the real DOM.

Quick view on Virtual DOM Process

1

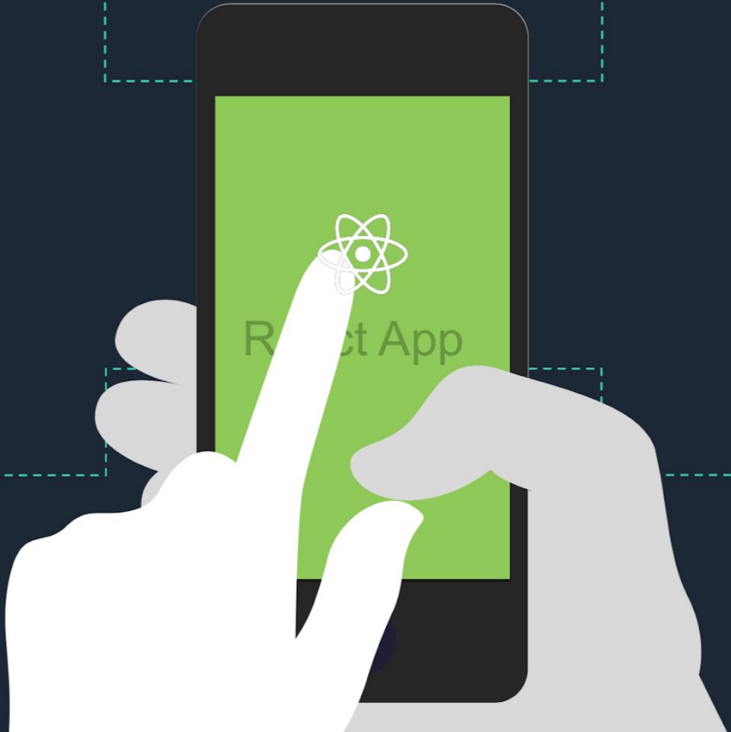
Detect Changes

Using **observables**, signal to notify our app some data has changed

2

Re-render Virtual DOM

Whenever setState() method is called on any component, ReactJS makes that component dirty and re-renders it



3

Apply Diff Algorithm

Compare previous virtual DOM with new virtual DOM to evaluate the changes

4

Update Real DOM

Only update real DOM with necessary changes evaluated by diff algorithm

QUESTIONS?