ReactJS - Front End migration story @ Realtor.com

Khanh Dao

1/26/2018

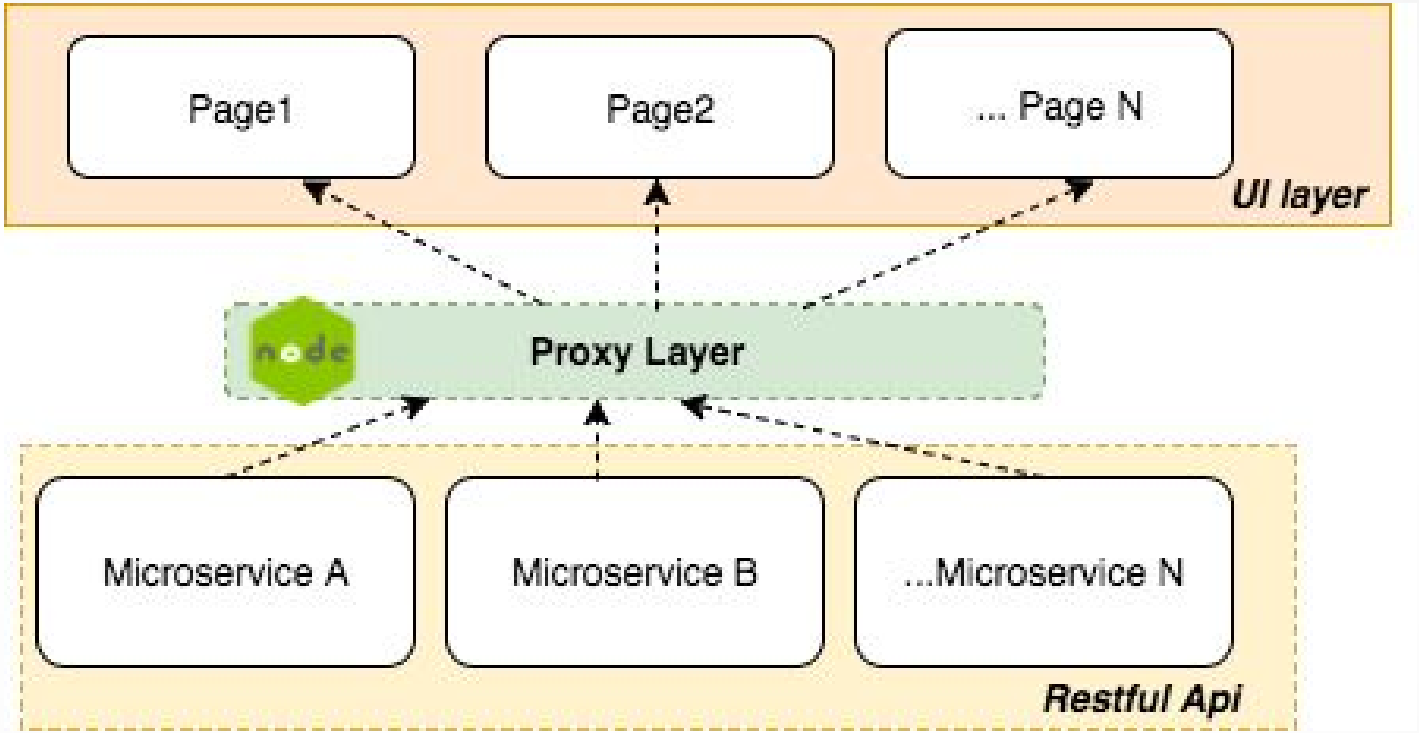
realtor.com®

Agenda

- A walk down memory lane
- Technical Debt
- ReactJS to rescue?
- Deep dive and lesson learned
- Q&A

realtor.com®

Recalling the past...

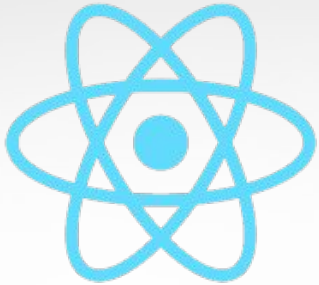


Recalling the past…(cont.)

- NodeJS/NPM system
- Custom Bootstrap
- jQuery and plugins
- DustJS
- Bimo
- PubSub pattern

Why didn’t it work for us?

- Large scale UI component and modular support
- When application grow larger, heavy DOM manipulation slow down client rendering and performance.
- Lacking of data management system that should only change the data needed to be change on DOM rendering, state management of data is confusing.

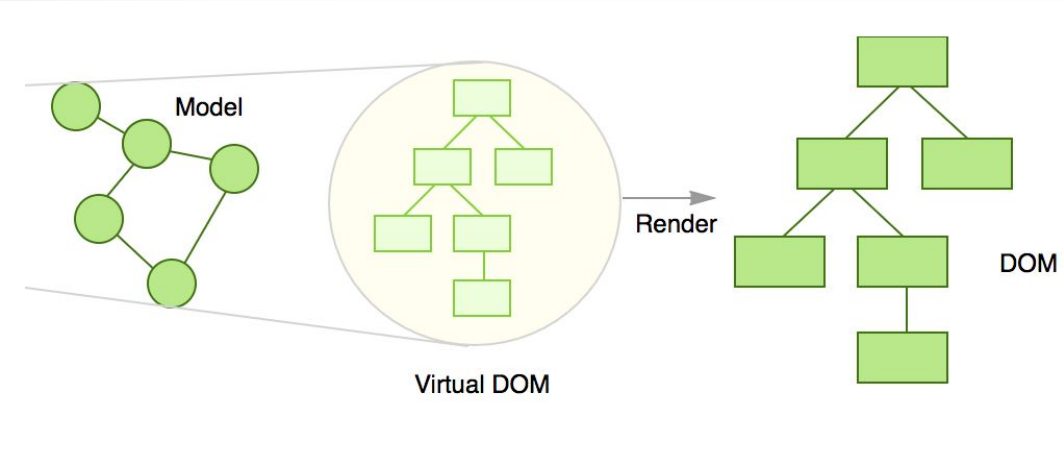


React

“React makes it painless to create interactive UIs. Design simple views for each state in your application, and React will efficiently update and render just the right components when your data changes.”

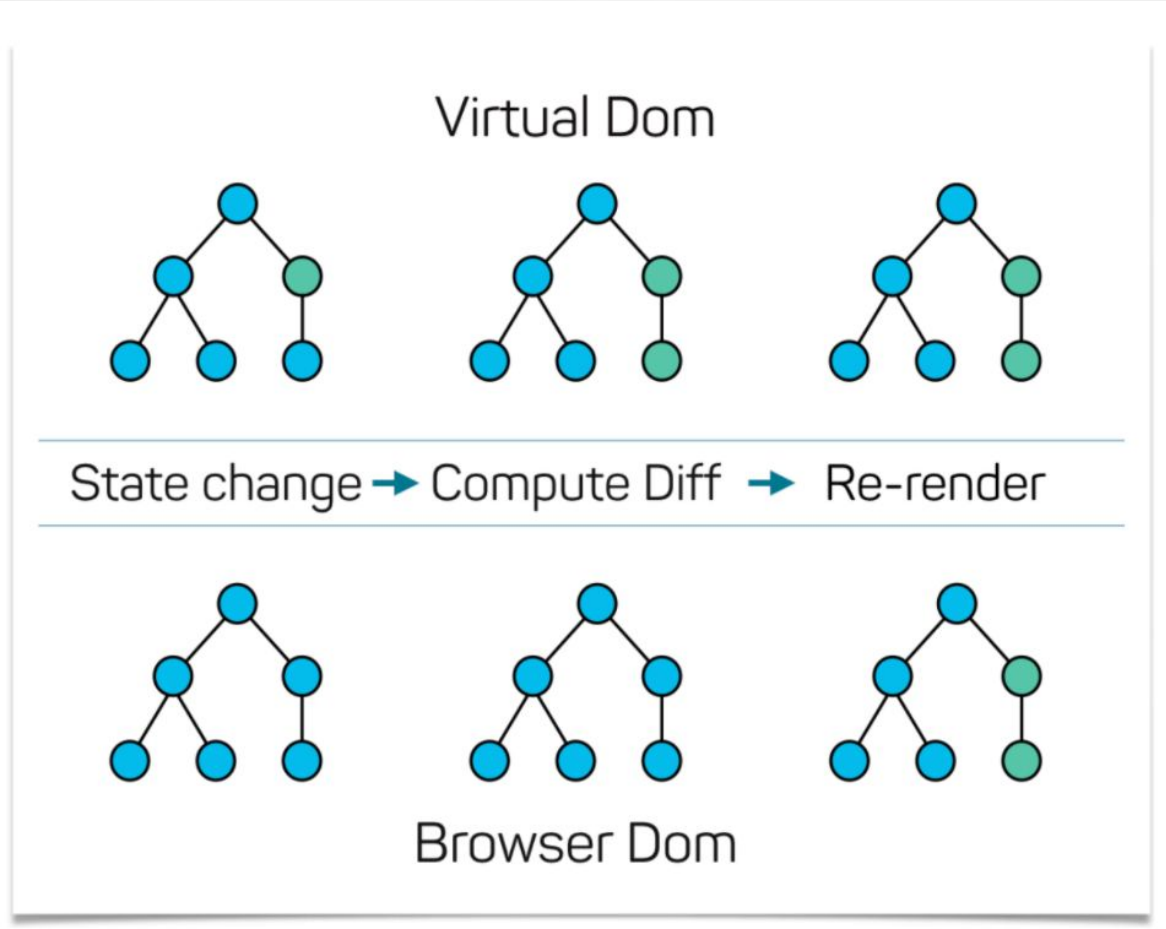
Reference: https://reactjs.org/

What is a Virtual DOM?



- Abstraction of actual HTML DOM
- Store state independent of the DOM and allow compare to the DOM

realtor.com®



realtor.com®

```
class Button extends React.Component {
  constructor() {
    super();
    this.state = {
      count: 0,
    };
  }
  updateCount() {
    this.setState((prevState, props) => {
      return { count: prevState.count + 1 }
    });
  }
  render() {
    return (
      <button onClick={() => this.updateCount()}>
          {this.props.name}: {this.state.count}
      </button>);
  }
}
ReactDOM.render(<Button name="Click me" />,
mountNode);
```

Prop, State, Ref

Prop:

- Parameter to a function
- React.PropTypes is used for data sanitization

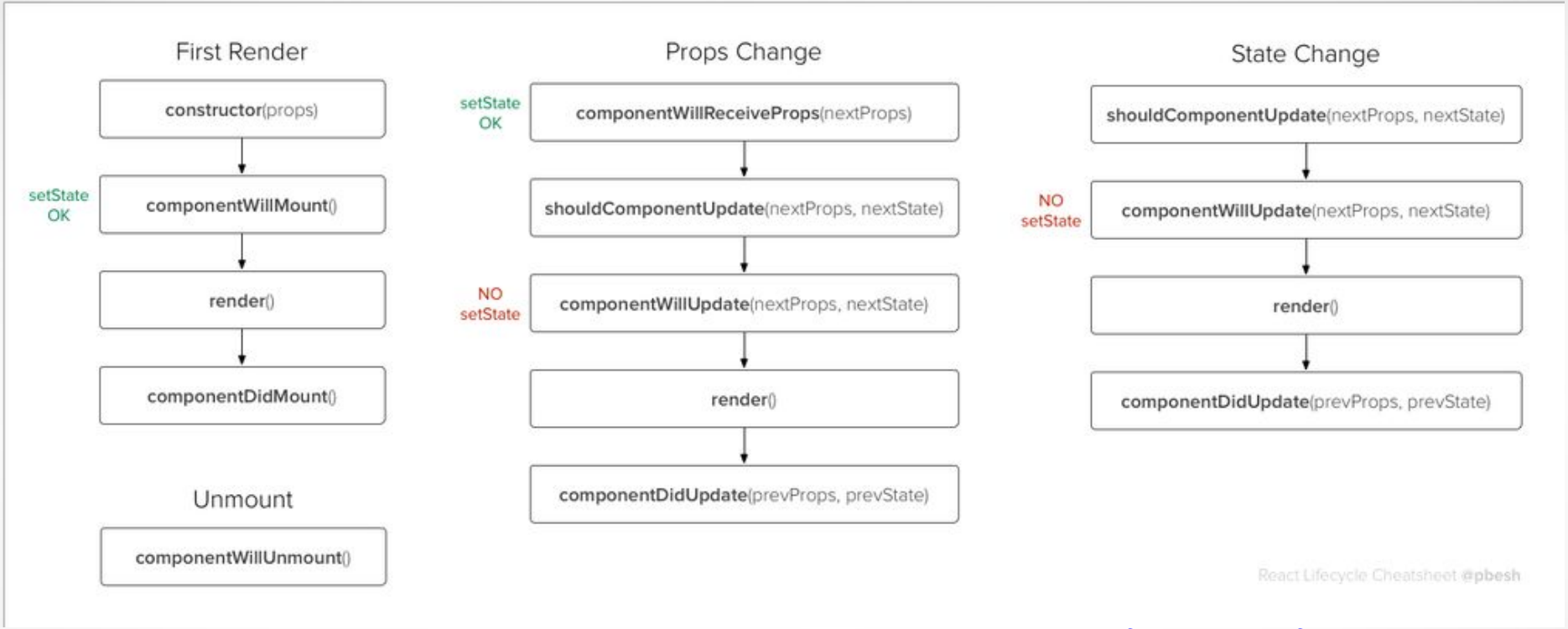
States

- Hold the current state inside component
  - getInitialState
  - setState

Ref

- Keep reference to component

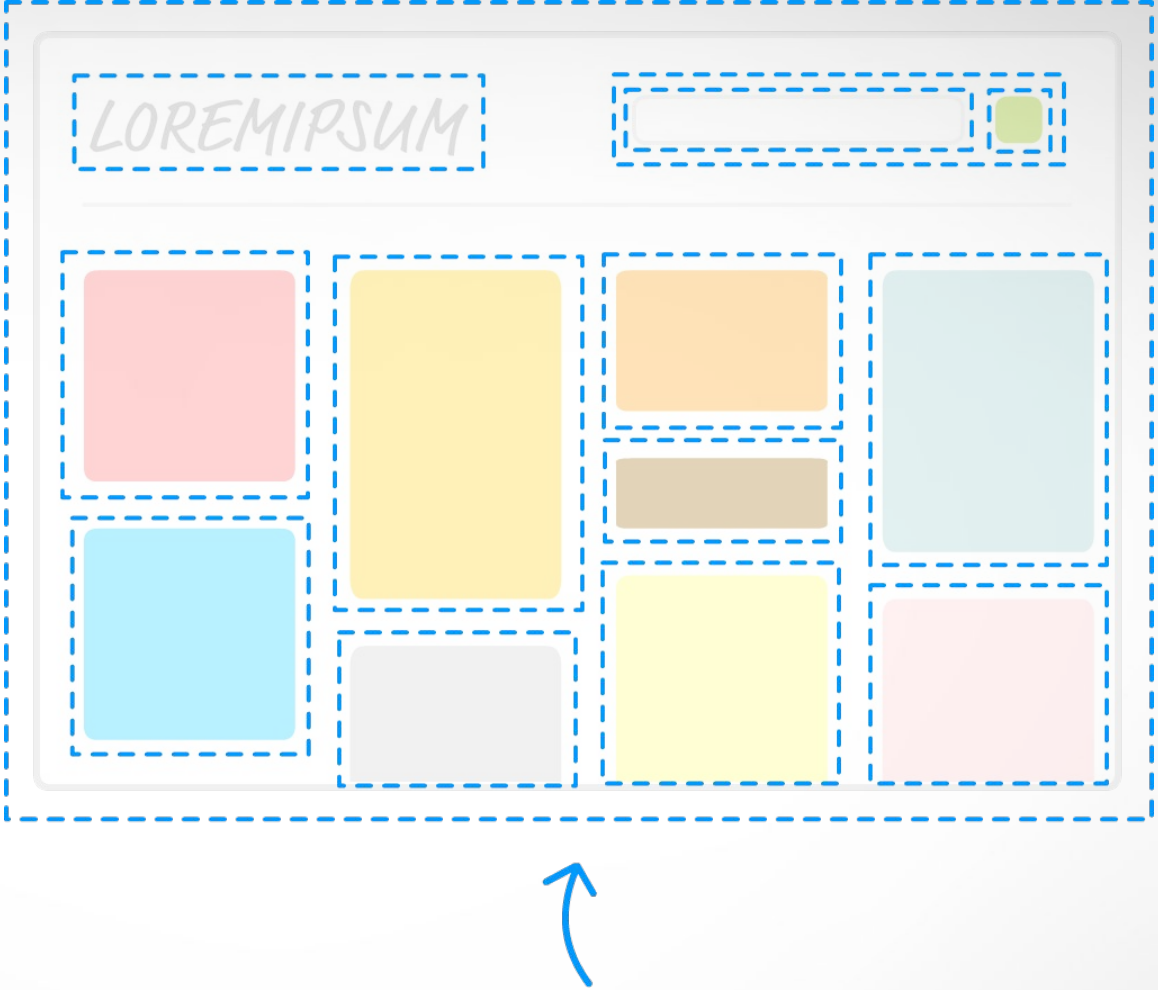
React LifeCycle



https://developmentarc.gitbooks.io/react-indepth/content/life_cycle/the_life_cycle_recap.html

realtor.com®

React Component



realtor.com®

What is a Component…?

- Block of text
- Block of HTML
- Mixture of text and HTML
- A container for other components

realtor.com®

Why Components are important?

- Accelerate development with reusability so you don’t repeat yourself
- Keep a consistent user experience across pages for the same parts of your pages
- Optimize requirement and design process
- Speed up the transition from design to development

realtor.com®

Patterns of developing component

- Presentational components
  - How things look
  - Accept ***props*** from container to render HTML block
  - No dependencies with application code
- Container components
  - How things work
  - Behavioral components

realtor.com®

React Benefits

- One way directional data flow is definitely produce more stable and predictable code
- Stable in-house eco system for share and reuse
  - Unit test to ensure code quality.
  - We've developed over 30 presentational UI components and over 8 behavioral components that share across multiple teams
  - Faster UX/UI cycle as team can adopt a change in component more quickly through seamless API interface, while expecting less bugs

khanh.dao@move.com

https://github.com/lelea2

THANK YOU

realtor.com®