The logo features the word "React" in a large, bold, black sans-serif font. A light blue infinity symbol is overlaid on the letters, with the text "WHAT'S NEW IN" positioned above the top curve and "(v16.3 & v16.4)" positioned below the bottom curve.

WHAT'S NEW IN

React 16

(v16.3 & v16.4)



/BASEMENT-LAB/MEETUP-REACT-2018-05-30

BASEMENT LAB

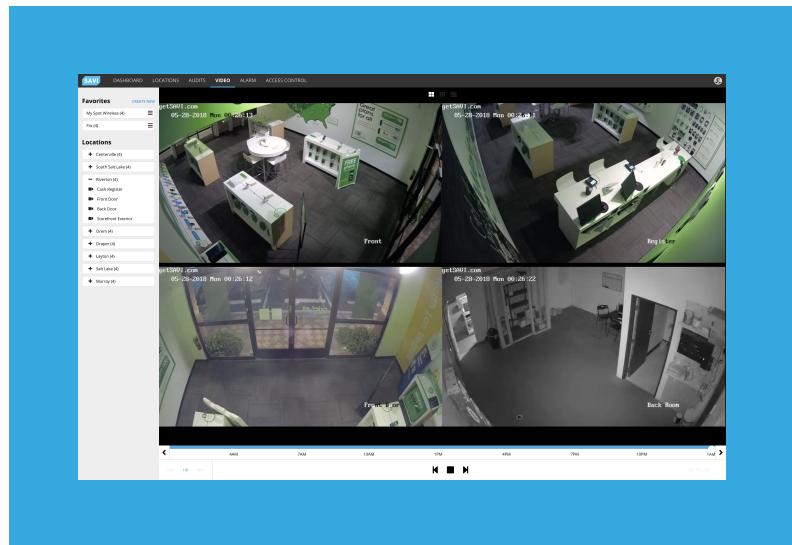
DEREK CLAIR

github.com/derekclair

@derekclair



getSavi.com



- React & Typescript - frontend
- GraphQL - client API
- Elixir - microservices backend
- Protobuf & gRPC - service-to-service network
- PostgreSQL & *undetermined*
- WE'RE HIRING!

React v16.0

SEPTEMBER 26, 2017

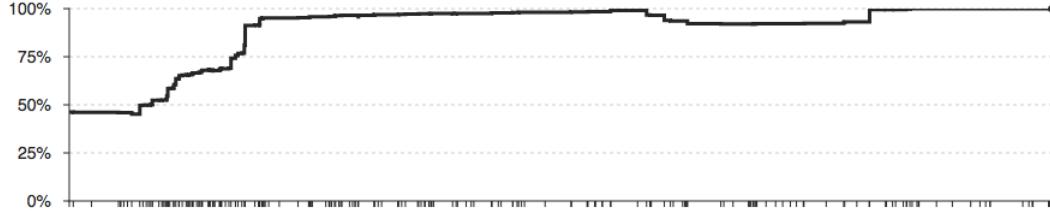
FIRST *MAJOR* VERSION UPDATE EVER!

REACT FIBER

2164 of 2164 unit tests passing

YES 

[Learn more about the rewrite...](#)



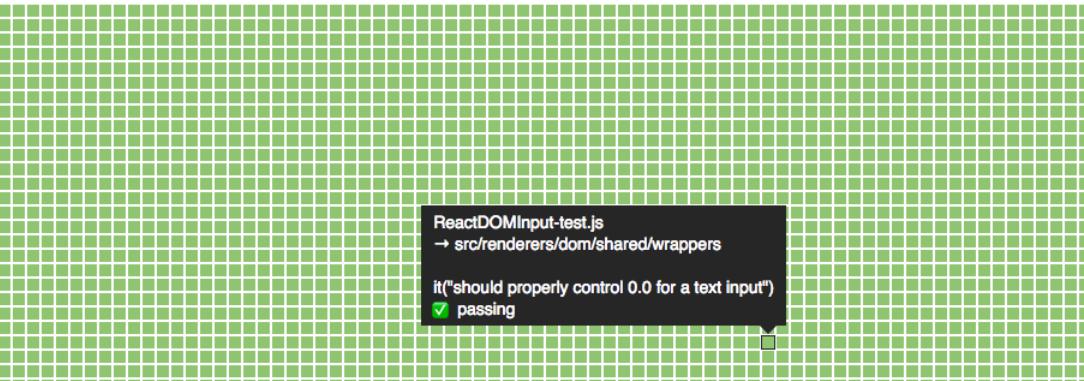
100%

75%

50%

25%

0%



ReactDOMInput-test.js
→ src/renderers/dom/shared/wrappers

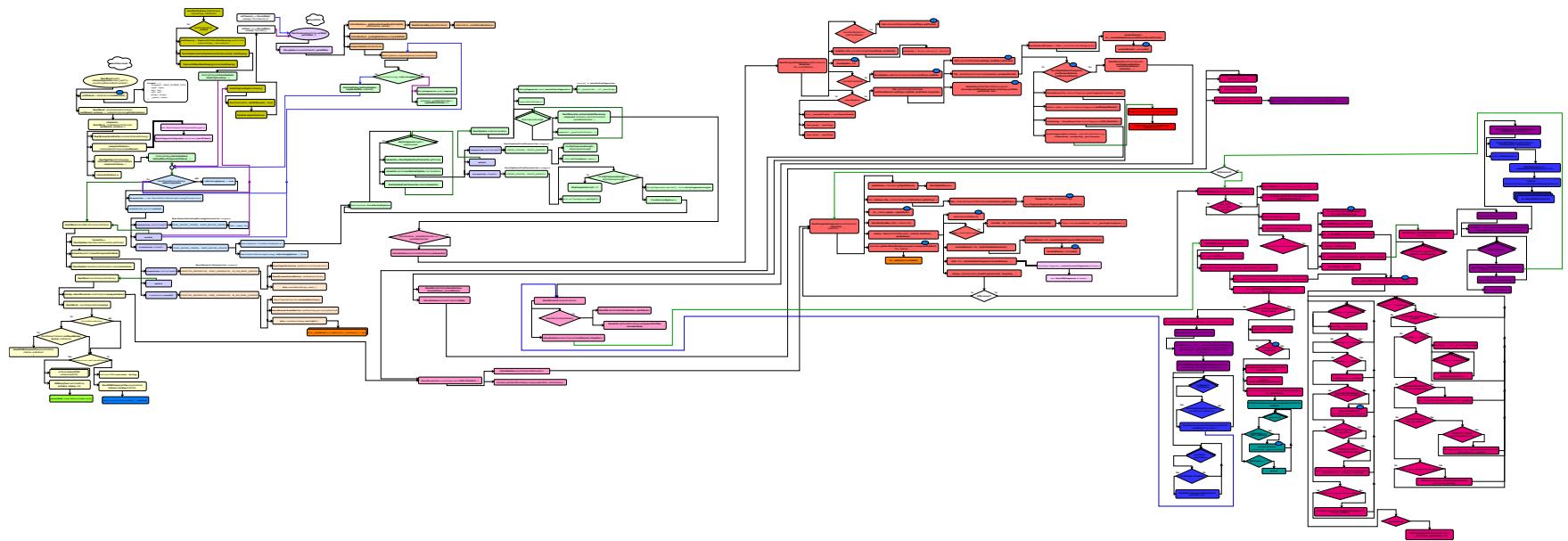
```
it("should properly control 0.0 for a text input")
  ✓ passing
```

What is React Fiber?
an optimization algorithm
React's CORE *reconciliation* algorithm

a

hella' gnarly

reconciliation algorithm



bogdan-lyashenko.github.io

Under the Hood ReactJS



More on React Fiber

- [ReactJS Blog] Update on Async Rendering - March 27, 2018 by Brian Vaughn
- [ReactJS Blog] React v16.0 - September 26, 2017 by Andrew Clark
- [YouTube] Sneak Peek: Beyond React 16 - March 2017 by Dan Abramov, React Conf 2017
★★★
- [YouTube] A Cartoon Intro to Fiber - March 2017 by Lin Clark, React Conf 2017
★★★
- bogdan-lyashenko.github.io/Under-the-hood-ReactJS
★★★
- github.com/acdlite/react-fiber-architecture
- [Slideshare] sabativi - react-fiber

React v16.3

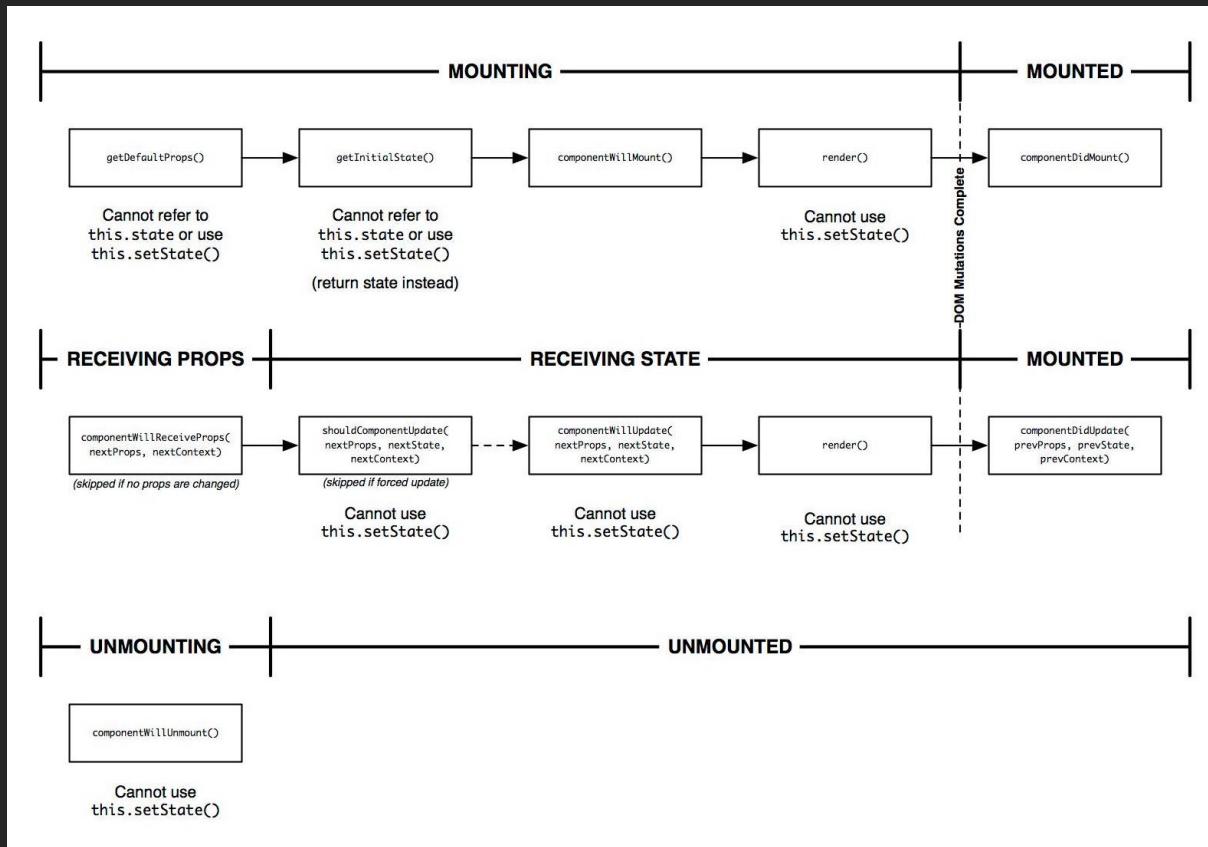
MARCH 29, 2018

so, what's new

NEW LIFECYCLES
NEW CONTEXT API

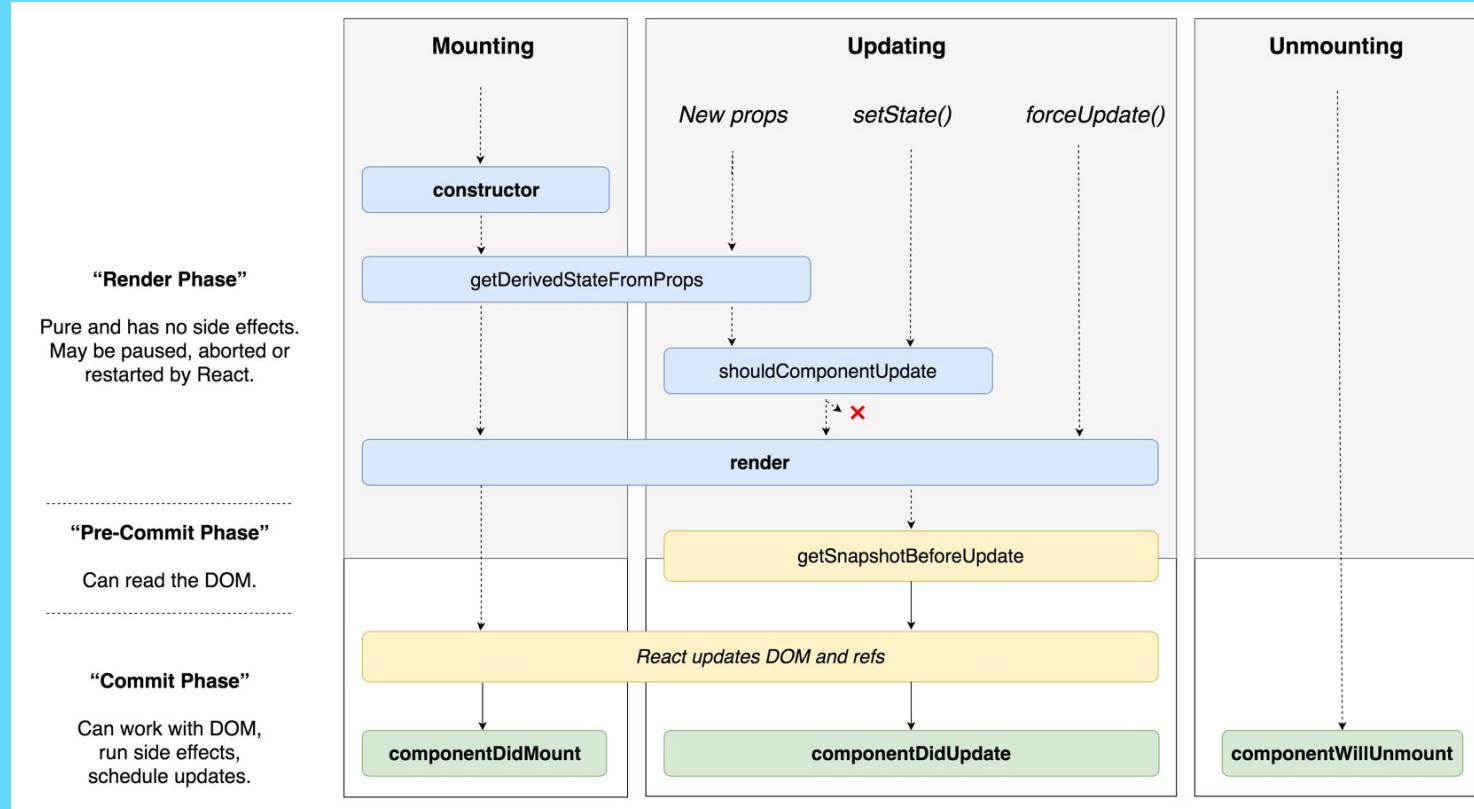
React Blog - v16.3 Announcement

Lifecycle <16.3



@sstur_

Lifecycle ≥ 16.3



@dan_abramov

Lifecycle API Side-by-Side

OLD LIFECYCLE <v16.3		≥ v16.3 NEW LIFECYCLE
getDefaults(props, context)	React.createClass	
getInitialState(props, context)	React.createClass	
constructor(props, context)	✗	constructor(props)
componentWillMount()	✗	
render()	✓	render()
componentDidMount()	✓	componentDidMount()
componentWillReceiveProps(nextProps, nextContext)	✗	static getDerivedStateFromProps(props, state)
shouldComponentUpdate(nextProps, nextState, nextContext)	✗	shouldComponentUpdate(props, state)
componentWillUpdate(nextProps, nextState, nextContext) ⇒ snapshot	✗	getSnapshotBeforeUpdate(prevProps, prevState)
render()	✓	render()
componentDidUpdate(prevProps, prevState, snapshot)	✗	componentDidUpdate(props, state)
componentWillUnmount()	✓	componentWillUnmount()
componentDidUnmount(nextProps, nextState) ⇒ snapshot	✗	

WHO USES THE CONTEXT API'S?

react-redux

react-mobx

flux

react-router

react-apollo

react-emotion

styled-components

react-chartjs

ant-design

material-ui

gatsby

@blueprintjs/*

spectacle

...

The legacy context API will be removed in a future major version. Use the new context API introduced with version 16.3. The legacy API will continue working for all 16.x releases.

- [ReactJS.org - Legacy Context docs](#)

THE END

@derekclair

github.com/derekclair