

React time!

What about the circle of life?? 🦁 🤖

Has a component **life**?

Yes.

It follows a cycle.

Creation, grown and dead.

Has a component **life?** x2

Some key concepts:

- *Stateful vs Stateless*
- Can components remember their own *state*?

Wait...

- Dude, WTF is the *state*???

A way to remember (you?)!!

The state in react is similar to the props, but it is private and fully controlled by the component.

The only way to add state to our components is using...

A CLASS!! DAMN IT!!!

Converting a Function to a Class

1. Create an ES6 class, with the same name, that extends `React.Component`.
2. Add a single empty method to it called `render()`.
3. Move the body of the function into the `render()` method.
4. Replace `props` with `this.props` in the `render()` body.
5. Delete the remaining empty function declaration.

Common life cycle process

Mount

- `constructor()` *
- `static getDerivedStateFromProps()`
- `render()` *
- `componentDidMount()` *

The methods with * are common ones.

Common life cycle process x2

Update

- `static getDerivatedStateFromProps()`
- `shouldComponentUpdate()`
- `render()` *
- `getSnapshotBeforeUpdate()`
- `componentDidUpdate` *

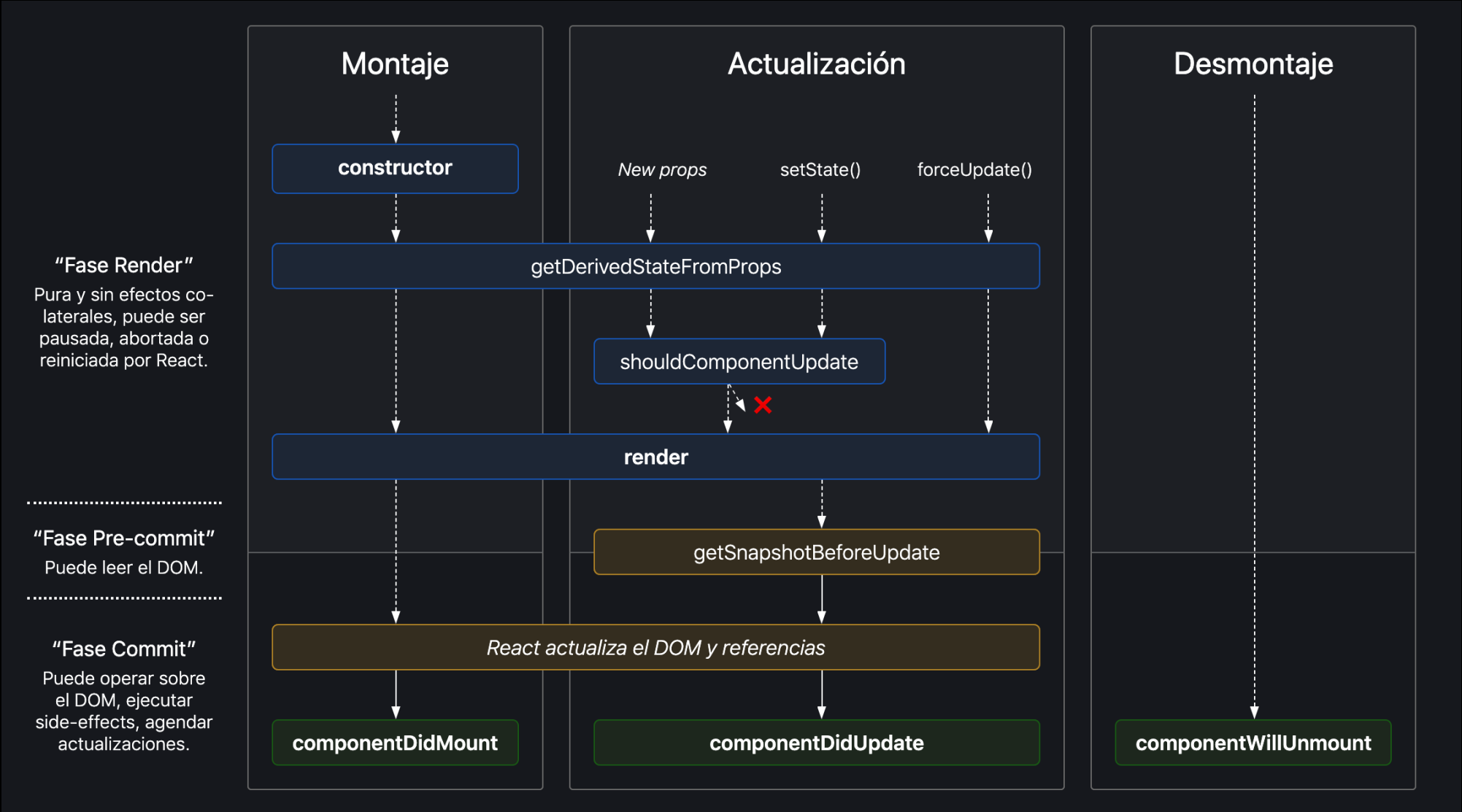
The methods with * are common ones.

Common life cycle process x3

Unmount

- `componentWillUnmount()`*


The methods with * are common ones.



A great power comes with a great responsibility

Let's use the state correctly!!

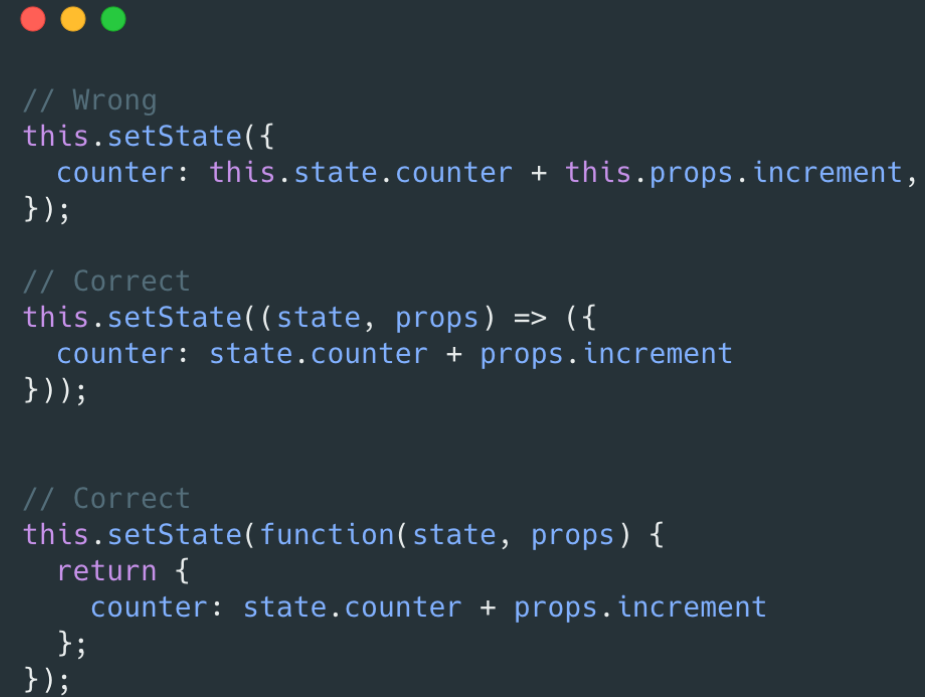
1. Do Not Modify State Directly



```
// Wrong
this.state.comment = 'Hello';

// Good
this.setState({ comment: 'Hello' });
```

2. State Updates May Be Asynchronous



```
// Wrong
this.setState({
  counter: this.state.counter + this.props.increment,
});

// Correct
this.setState((state, props) => ({
  counter: state.counter + props.increment
}));

// Correct
this.setState(function(state, props) {
  return {
    counter: state.counter + props.increment
  };
});
```

3. State Updates are Merged

```
// Component constructor
constructor(props) {
  super(props);
  this.state = {
    posts: [],
    comments: []
  };
}

// Shallow update
componentDidMount() {
  fetchPosts().then(response => {
    this.setState({
      posts: response.posts
    });
  });
}
```

Not luxury Events

TL;DR

1. Synthetic events
2. Binding

Links

- [React Component Documentation](#)
- [Synthetic events Documentation](#)