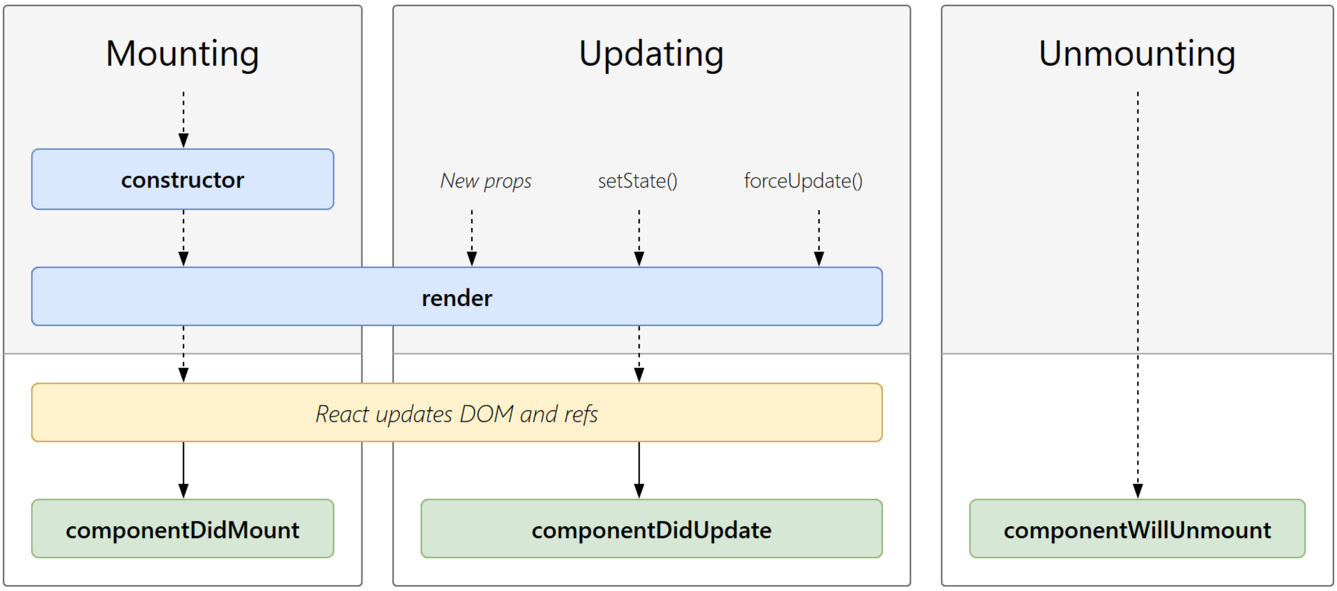
9 - CardNav

React Lifecycle Hooks



Each component has several “lifecycle methods” that you can override to run code at particular times in the process. You can use this lifecycle diagram as a cheat sheet.

The list below are the commonly used lifecycle methods.

- constructor()
- render()
- componentDidMount()
- componentDidUpdate()
- componentWillUnmount()
-

In out app:: Info Container

```
class InfoContainer extends Component {
  //=== BEGINS Life cycle hooks
  constructor() {
    super();
    this.state = {
      currentArticle: {
        title: null,
        img: null,
        list: [],
        src: null
      }
    };
  }

  componentDidMount() {
    this.setArticle();
  }

  componentDidUpdate(prevProps) {
    if (this.props.match !== prevProps.match)
      this.setArticle();
  }

  render() {
    return <Info article={this.state.currentArticle}/>;
  }

  componentWillUnmount() {

  }
  //=== ENDS life cycle hooks

  //=== BEGIN Helper functions
  findArticle = _ => {
    const current = articles.find(article => article.id ===
this.props.match.params.id);
    return current;
  }

  setArticle = _ => {
    this.setState(_ => {
      return { currentArticle: this.findArticle() }
    })
  }
  // === ENDS Helper Functions
}
```

Router variables

React router injects some props to every component that is wrapped inside a route. One of the most used is match, that stores the content of the route and its params. We can access it by the props => match => params => parameter name (we define it on the router, in this case :id === params.id)

Read More:

- Lifecycle Hooks

<< return