POINT D'ATTENTION SUR LES SPA

DES SPA PARTOUT ... FRAMEWORK DEPENDENT ... ÉVOLUTION TRÈS RAPIDE

2014

2016



2017

2019

```
var Component2014 = React.createClass({
    render(props){
        ...
    }
    ...
});
```

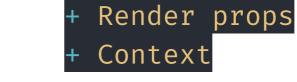
```
const StatelesComponent2016 = (props) ⇒ <div>... </div>;
class StatefullComponent extends React.Component {
   constructor(props) {
      super(props);
      ...
   }
   render(props) {
      ...
   }
}
```

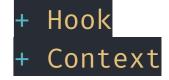
```
class StatelesComponent2917 extends React.PureComponent {
    rencer(props) {
        ...
    }
} class StatefullComponent extends React.Component {
    constructor(props) {
        super(props);
        ...
    }
    render(props) {
        ...
}
```

```
const futureComponent = (props) ⇒ {
   ...
   return <div> ... </div>
}
```

```
+ Mixins
+ Flux
```

```
+ HOC
+ REDUX
```





+ API changes

+ API changes

POINT D'ATTENTION SUR LES WEB COMPONENT

UN STANDARD W3C QUI A MIS TRÈS LONGTEMPS À ARRIVER ... ET PLUS FORCÉMENT ADAPTÉ AUX ATTENTES ACTUELLES

<u>Web components</u> = Shadow DOM + Custom Element + HTML Template

Une API assez bas niveau plus destinée à construire des frameworks qu'à être utilisée directement

Approche de templates dirigés par les données (à la angular) préférée à une approche expressive (à la react)

Framework star: lit-element

Recyclage de vieux frameworks : ionic, vue.js

Tous les frameworks SPA mainstream (react, angular, vue.js, svelte) permettent d'inclurent des web components dans leurs composants propres

Mais c'est un standard 🏰