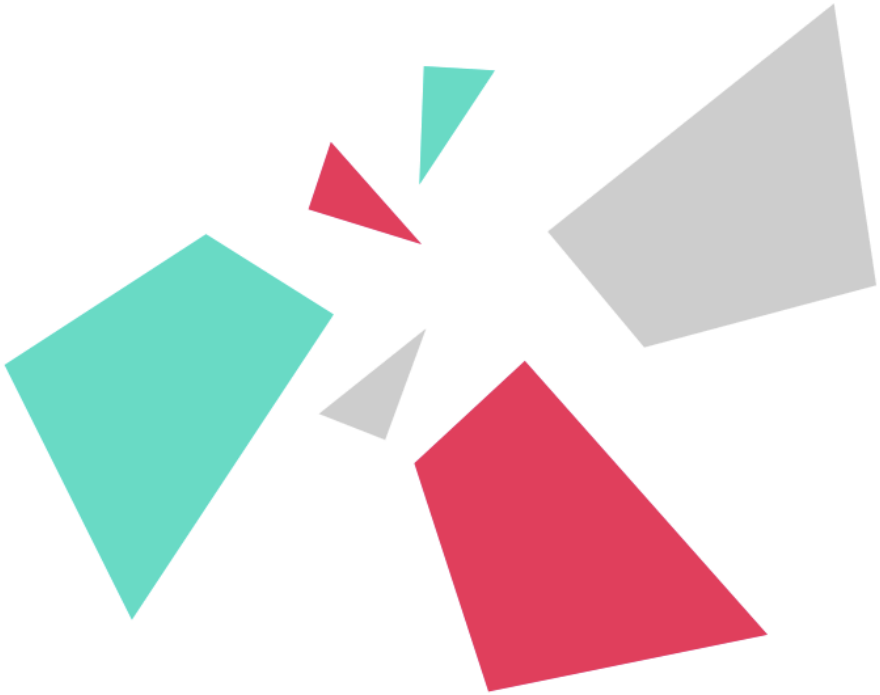
React

Présentation de la librairie et prototypage d'un jeu de type "clicker"

Michaël Hoste

michael@80limit.com • @michaelhoste • github.com/michaelhoste



80LIMIT

http://80limit.com

CREATIVE
MONKEYS

http://creativemonkeys.eu

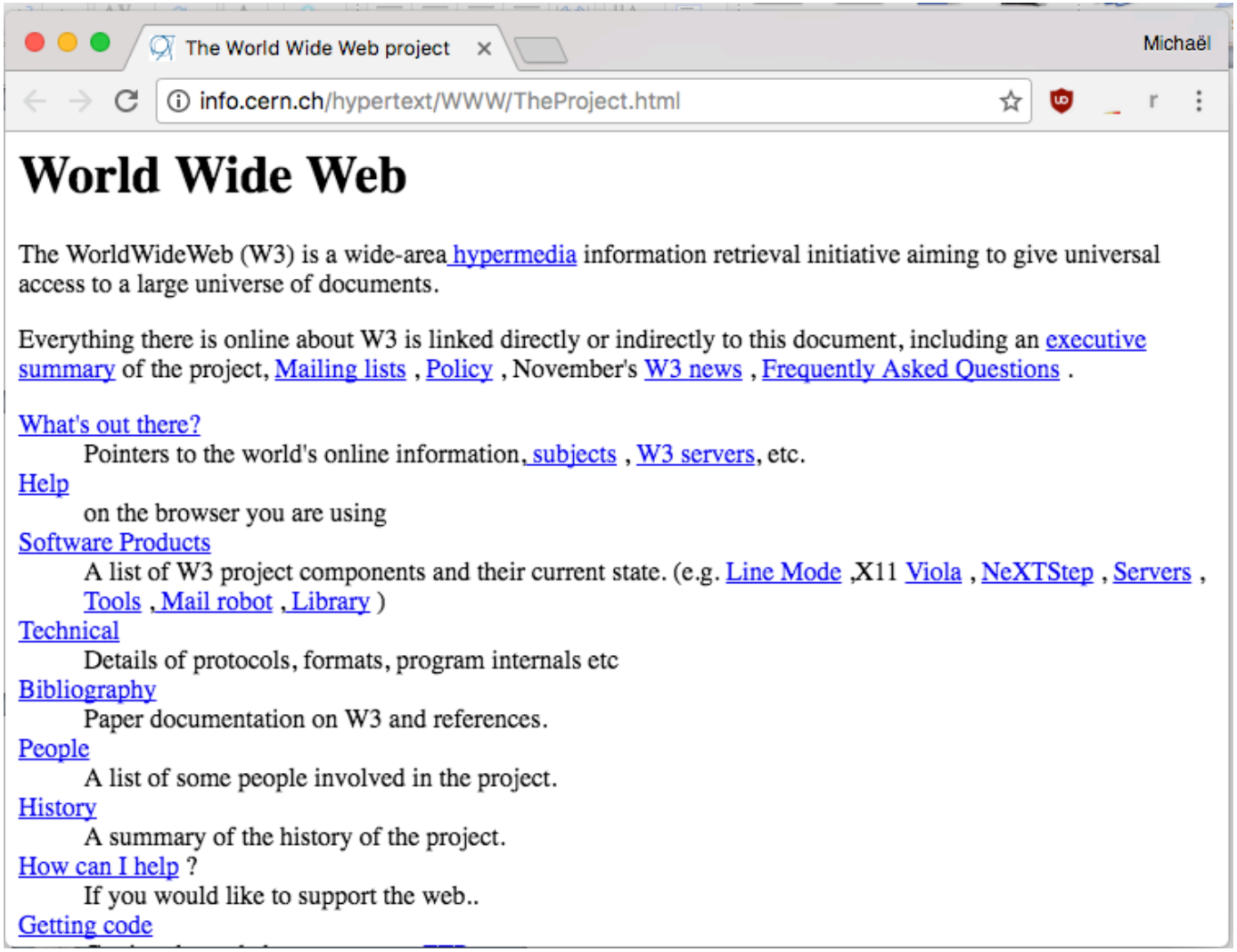

B501
Imagix Mons
Rue des Grands Prés
Rue Paul Verlaine
de Nimy
Rue Jules Cornet
Avenue du Tir
R50
Boulevard John Fitzgerald Kennedy
N556
Université de Mons - Campus...
Lotto Mons Expo
Rue Valenciennoise
CHU Ambroise Paré
Avenue des Bassins
Avenue Mélina Mercouri
Rue de Nimy
Rue Fariaux
Rue de la Sucrerie
Mons
Collégiale Sainte-Waudru de Mons
Galerie Du Pistolet D'Or
N90
N51
N544
Rue de Bouzanton
Université de Mons - Campus Polytech
Avenue de Saint-Pierre
Rue Vilaine
Avenue du Pont Rouge
Rue du Foyer
Avenue d'Hyon
Rue du Delta
Dique de Cuesmes
Les Peupliers

Backend (serveur)

Frontend (navigateur web)

Une brève histoire du web

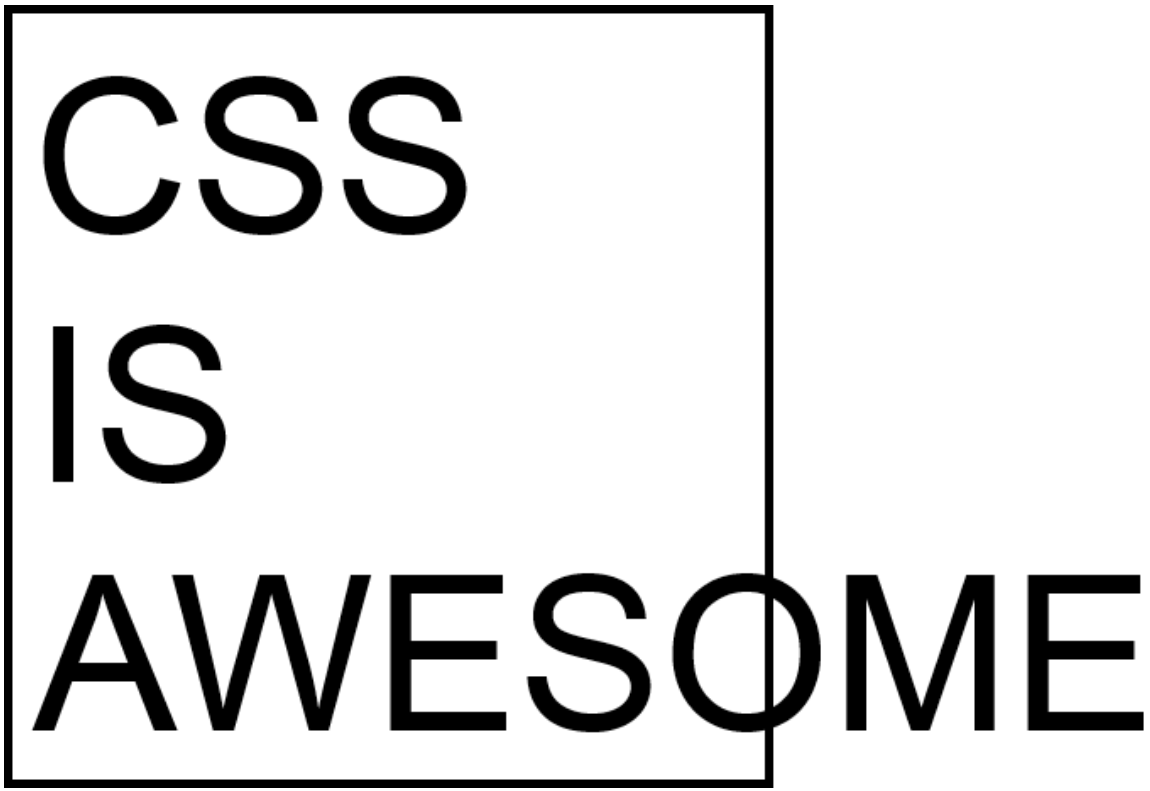
1990


The World Wide Web project
Michaël
info.cern.ch/hypertext/WWW/TheProject.html
World Wide Web
The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.
Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists , Policy , November's W3 news , Frequently Asked Questions .
What's out there?
Pointers to the world's online information, subjects , W3 servers, etc.
Help
on the browser you are using
Software Products
A list of W3 project components and their current state. (e.g. Line Mode ,X11 Viola , NeXTStep , Servers , Tools , Mail robot , Library)
Technical
Details of protocols, formats, program internals etc
Bibliography
Paper documentation on W3 and references.
People
A list of some people involved in the project.
History
A summary of the history of the project.
How can I help ?
If you would like to support the web..
Getting code

```
<tr width="850" height="420" valign="top">
    <td>
        <table border="0" width="768" height="420" cellpadding="0" cellspacing="0">
            <tr>
                <td width="130" height="420" valign="top">
                    <table border="0" width="130" height="100%" cellpadding="0" cellspacing="0">
                        <tr>
                            <td width="130" height="25" background="bordercadre.jpg">
                            </td>
                        </tr>
                        <tr>
                            <td width="130" height="100%" valign="top" background="fond2.gif">
                                <table border="0" width="130" height="100%" cellpadding="0" cellspacing="0">
                                    <tr valign="top">
                                        <td width="130" height="10" colspan="2"><center><b>INFOS</b></center>
                                        </td>
                                    </tr>
                                    <tr valign="top">
                                        <td width="15" height="10" align="right">-
                                        </td>
                                        <td width="110"><a href="page.php?nom=description"><font size="2" color="#0
                                        </td>
                                    </tr>
                                    <tr valign="top">
                                        <td width="15" height="10" align="right">-
                                        </td>
                                        <td width="110"><a href="page.php?nom=endroit"><font size="2" color="#00000
                                        </td>
                                    </tr>
                                    <tr valign="top">
                                        <td width="15" height="10" align="right">-
                                        </td>
                                        <td width="110"><a href="page.php?nom=prix"><font size="2" color="#000000">
                                        </td>
                                    </tr>
                                    <tr valign="top">
                                        <td width="15" height="10" align="right">-
                                        </td>
                                        <td width="110"><a href="page.php?nom=date"><font size="2" color="#000000">
                                        </td>
                                    </tr>
                                    <tr valign="top"><td colspan="2" height="10" width="130"></td></tr>
                                    <tr valign="top">
                                        <td width="130" height="10" colspan="2"><center><b>MATERIEL</b></center>
                                        </td>
```

~1998

~2000



2006

Contenu	Style	Interaction
DOM	CSS	JavaScript

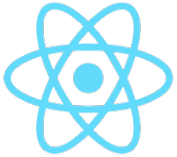
XMLHttpRequest

jQuery, frameworks et ReactJS

2006...
Librairie

2010-...
Frameworks

2013-...
Librairie



React

ReactJS

- **Librairie** permettant de créer des interfaces utilisateurs de manière déclarative.
- Créé et maintenu principalement par Facebook.
- Open-Source (MIT).
- Utilisé par Facebook (20 000 components), WhatsApp, Netflix, AirBnB, WordPress, etc.

Peut cohabiter avec d'autres technos !

Note : Utilisez ES6

- ES6 vous permettra d'utiliser les classes !
  - Sans ES6 => React.createClass()

- Anciennes Mixins plus valables.

- Utilisez un compilateur (Babel) en attendant la compatibilité totale des navigateurs (bientôt !).

ReactJS - Techniquement

1. Une approche en "components".

2. Le "One-Way Data Flow".

3. Le "Virtual DOM".

4. Une API (environ 10 fonctions).

1. Components

« Il faut toujours séparer le HTML, le CSS et le JavaScript dans une application web »

~~« Il faut toujours séparer le HTML, le CSS et le JavaScript dans une application web »~~

En ReactJS, un component réunit le HTML et le JavaScript au sein d'un même fichier.

~~« Il faut toujours séparer le HTML, le CSS et le JavaScript dans une application web »~~

En ReactJS, un component réunit le HTML et le JavaScript au sein d'un même fichier.

(et le CSS !?)

```
1  class MyComponent extends Component {
2    render() {
3      return (
4        <div>
5          Hello world
6        </div>
7      );
8    }
9  }
```

```
class Items extends React.Component {

  constructor(props) {
    super(props)

    this.state = {
      items: []
    }
  }

  componentDidMount() {
    this.reloadItems()
  }

  reloadItems() {
    $.get('/items', (data) => {
      this.setState({ items: data })
    })
  }

  render() {
    return (
      <div className="container">
        { this.renderItems() }
      </div>
    )
  }

  renderItems() {
    return _.map(this.state.items, (item) => {
      return this.renderItem(item)
    })
  }

  renderItem(item) {
    return (
      <div className="item" key={item.id}>
        { item.name }
      </div>
    )
  }
}

module.exports = Items

```

```
class Items extends React.Component {

  constructor(props) {
    super(props)

    this.state = {
      items: []
    }
  }

  componentDidMount() {
    this.reloadItems()
  }

  reloadItems() {
    $.get('/items', (data) => {
      this.setState({ items: data })
    })
  }

  render() {
    return (
      <div className="container">
        { this.renderItems() }
      </div>
    )
  }

  renderItems() {
    return _.map(this.state.items, (item) => {
      return this.renderItem(item)
    })
  }

  renderItem(item) {
    return (
      <div className="item" key={item.id}>
        { item.name }
      </div>
    )
  }
}

module.exports = Items
```

JSX

```
1  class MyComponent extends Component {
2    render() {
3      return (
4        <div>
5          Hello world
6        </div>
7      );
8    }
9  }
```

```
1  class MyComponent extends Component {
2    render() {
3      return React.createElement(
4        "div",
5        null
6        "Hello world"
7      );
8    }
9  }
```

JSX

```
<label class="label" for="name">
```

devient

```
<label className="label" htmlFor="name">
```

2. Le "One-Way Data Flow"

Comparaison jQuery, AngularJS, ReactJS

- jQuery :
  - Etat lié au DOM .
  - https://jsfiddle.net/michaelhoste/5hp42rfg/
- AngularJS :
  - Two-Way Data Binding
  - https://jsfiddle.net/michaelhoste/jo33cqc0/
- ReactJS :
  - One-Way Data Flow
  - https://jsfiddle.net/michaelhoste/oohj66p1/

PROPS

Immutables

Passés par le component parent

this.props

Component "stateless"

STATE

Mutables

Appartiennent au component

this.state
this.setState()

Component "stateful"

Quand le **state** d'un component change (via "setState"), le component et tous les sous-components sont re-rendus.

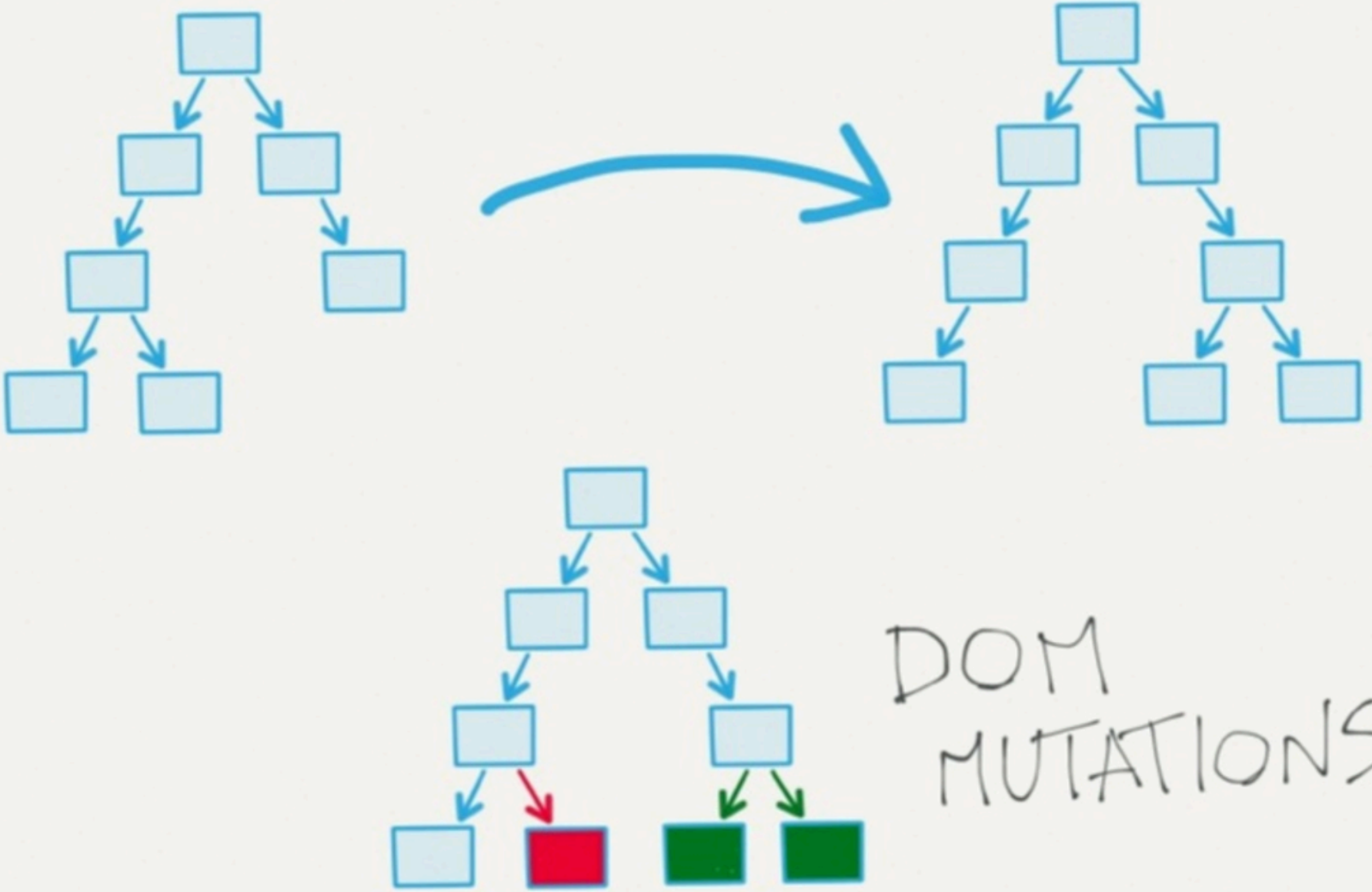
Quand les **props** reçues par un component changent, le component et tous ses sous-components sont re-rendus.

Les propriétés descendent,
les actions remontent

3. Virtual DOM

STATE 1

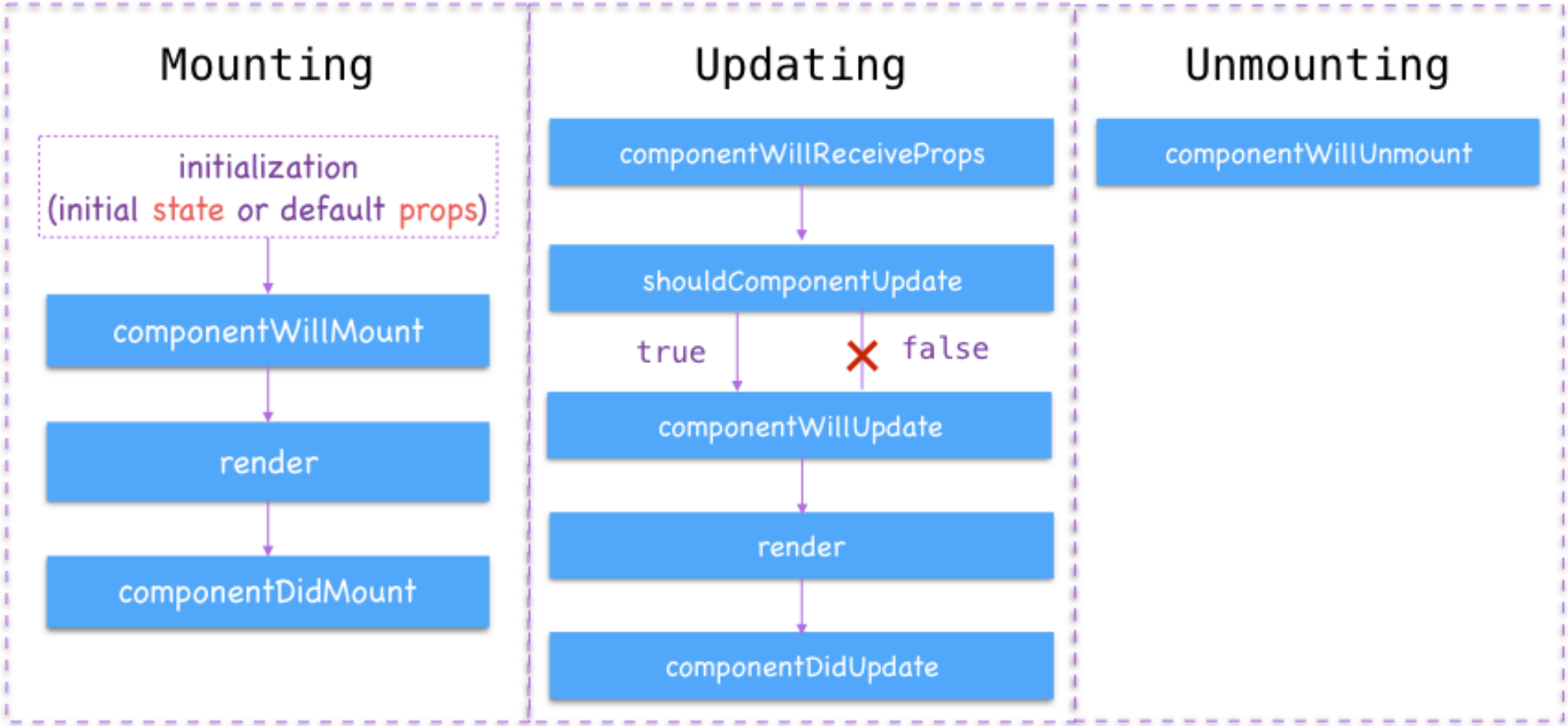
STATE 2



L'API

Cycle de vie des components

- Mounting
  - constructor()
  - componentWillMount()
  - render()
  - componentDidMount()
- Updating
  - componentWillReceiveProps()
  - shouldComponentUpdate()
  - componentWillUpdate()
  - render()
  - componentDidUpdate()
- Unmounting
  - componentWillUnmount()


Mounting
initialization
(initial state or default props)
componentWillMount
render
componentDidMount
Updating
componentWillReceiveProps
shouldComponentUpdate
true
false
componentWillUpdate
render
componentDidUpdate
Unmounting
componentWillUnmount

componentDidMount()

- Exécuté une seule fois après le rendu du component.
- Binder des éléments du DOM (jQuery).
- Lancer des timers.
- Requêtes Ajax.

componentWillReceiveProps()

- la méthode reçoit :
  - nextProps

- Permet de de mettre à jour l'état du component en fonction des nouvelles props reçues (si nécessaire).

shouldComponentUpdate()

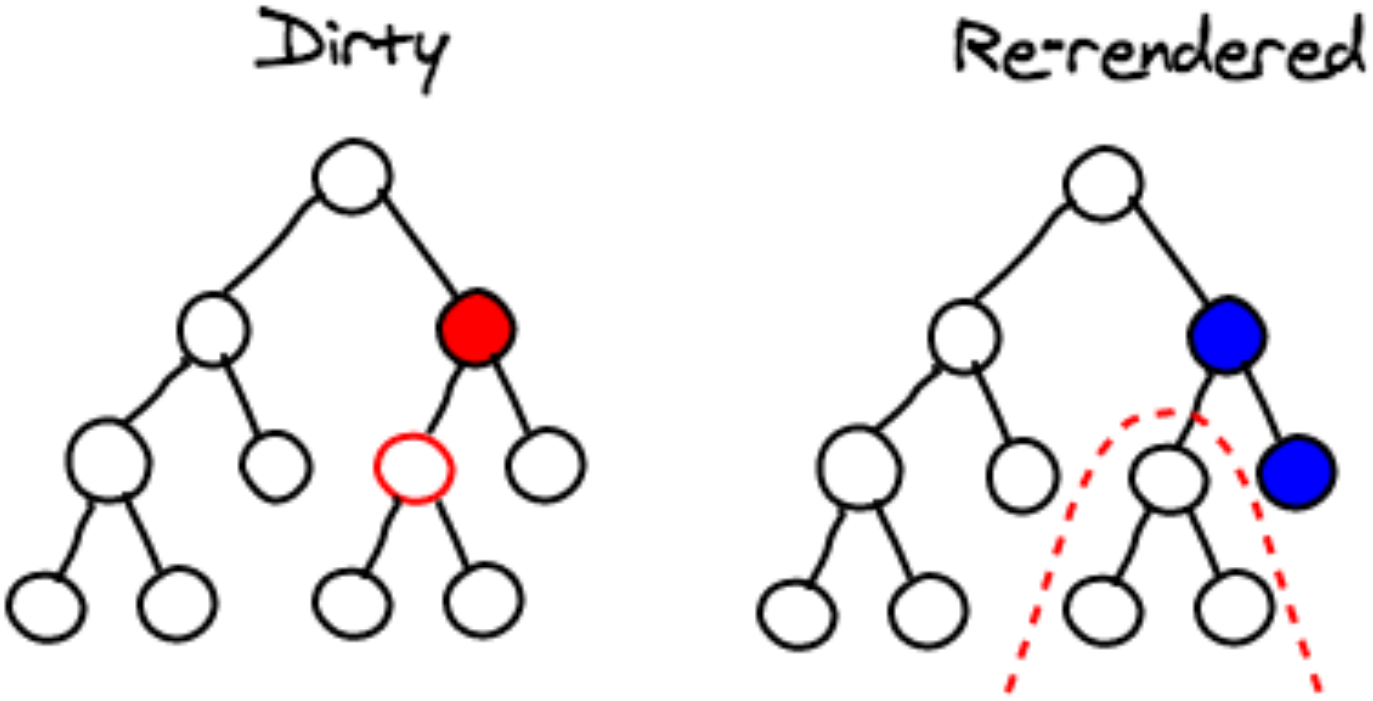
- la méthode reçoit :
  - nextProps
  - nextState

- Permet de choisir si on veut mettre à jour le component en fonction du nouveau *state* ou des nouvelles *props*.

- Retourner "false" permet d'ignorer la mise à jour.

- Pratique si certains éléments du state ou des props n'agissent pas sur l'affichage.

- **Attendre phase d'optimisation !**



http://calendar.perfplanet.com/2013/diff/

componentWillUpdate()

- la méthode reçoit :
  - nextProps
  - nextState

- Permet d'effectuer des actions juste avant l'affichage component en fonction du nouveau *state* ou des nouvelles *props*.

- Pour les animations CSS, par exemple.

componentDidUpdate()

- la méthode reçoit :
  - prevProps
  - prevState
- Requête AJAX si les props ont changé et le justifient.
- Agir sur le DOM (jQuery).

componentWillUnmount()

- S'exécute lorsque le component est supprimé.
- Permet d'arrêter des timers.
- Permet de nettoyer le DOM si nécessaire.

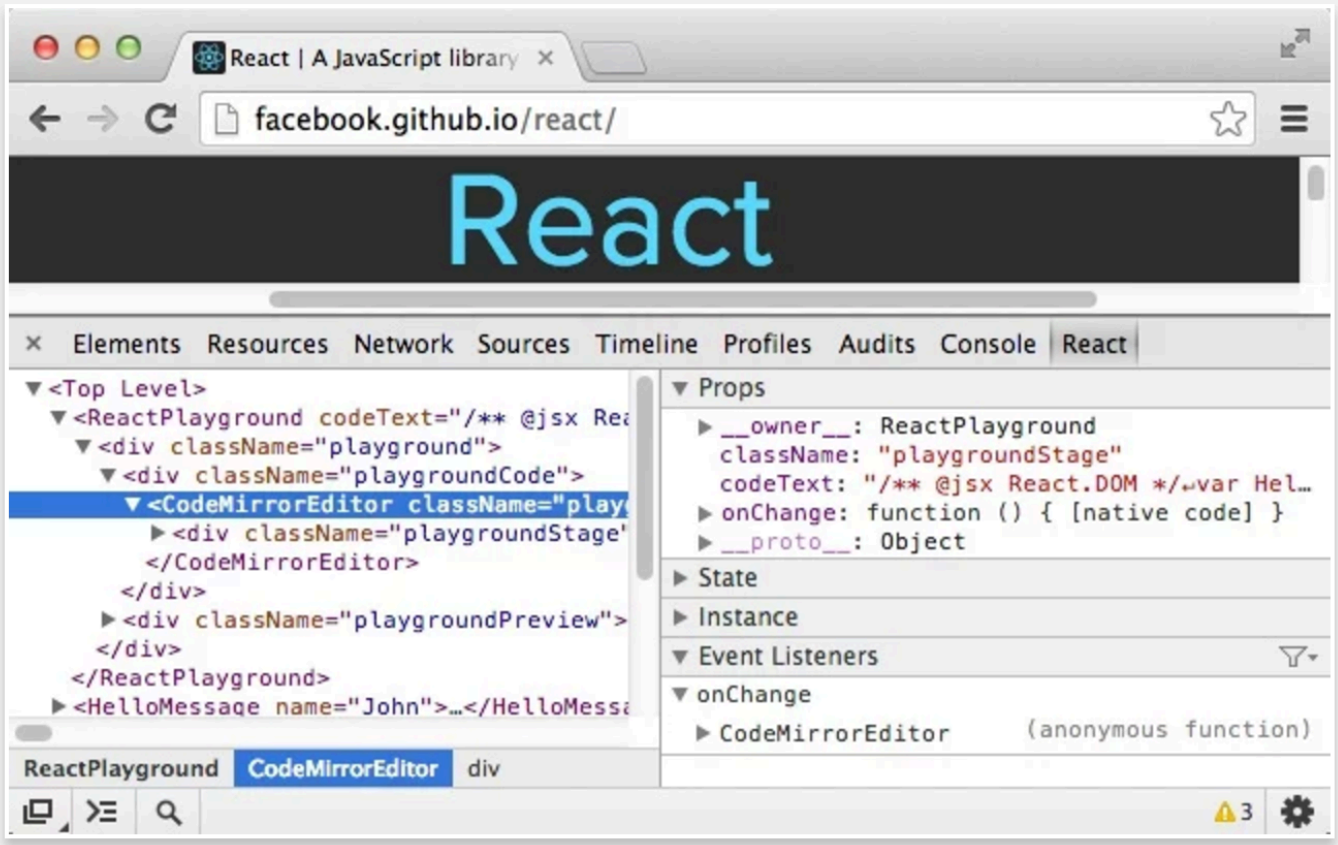
React Developer Tools

proposé par Facebook

★★★★☆ (510) | Outils de développement | 442 084 utilisateurs

AJOUTÉ À CHROME

PRÉSENTATION | AVIS | ARTICLES SIMILAIRES



Compatible avec votre appareil

Adds React debugging tools to the Chrome Developer Tools.

React Developer Tools is a Chrome DevTools extension for the open-source React JavaScript library. It allows you to inspect the React component hierarchies in the Chrome Developer Tools.

You will get a new tab called React in your Chrome DevTools. This shows you the root React components that were rendered on the page, as well as the subcomponents that they ended up rendering.

By selecting one of the components in the

Site Web

Signaler un abus

Informations supplémentaires

Version : 0.15.4

Mise à jour : 3 septembre 2016

Taille : 179KiB

Langue : English

LES UTILISATEURS DE CETTE EXTENSION ONT ÉGALEMENT UTILISÉ

SEOquake ★★★★½ (1719)

Version Check for Prestashop ★★★★☆ (7)

WoW Board Helpers ★★★★½ (4)

Tag Assistant (by Google) ★★★★☆ (595)

Créer une application React

Installation

- Template proposé par Facebook :
  - https://github.com/facebook/create-react-app
- Installer NodeJS
- Executer :
  - npx create-react-app mon-application
  - yarn start

https://www.kongregate.com/games/playsaurus/clicker-heroes

$112K

per click
$68

per second
$1685

Load | Save

LEVEL UP $100	Brainstorming Lvl 10
LEVEL UP $67	Entrepreneurial Coach Lvl 1
LEVEL UP $615	Market Research Lvl 3
LEVEL UP $1822	Co-Founder Lvl 2
LEVEL UP $7290	Prototype Lvl 2
LEVEL UP $27K	Seed Funding Lvl 1

$112K

per click
$68

per second
$1685

Load | Save

LEVEL UP
$100
Brainstorming
Lvl 10

LEVEL UP
$67
Entrepreneurial Coach
Lvl 1

LEVEL UP
Market Research

https://github.com/michaelhoste/clicker

LEVEL UP
$1822
Co-Founder
Lvl 2

LEVEL UP
$7290
Prototype
Lvl 2

LEVEL UP
$27K
Seed Funding
Lvl 1

App

$112K

per click
$68

per second
$1685

Load

Save

LEVEL UP
$100

Brainstorming
Lvl 10

LEVEL UP
$67

Entrepreneurial Coach
Lvl 1

LEVEL UP
$615

Market Research
Lvl 3

LEVEL UP
$1822

Co-Founder
Lvl 2

LEVEL UP
$7290

Prototype
Lvl 2

LEVEL UP
$27K

Seed Funding
Lvl 1

$112K

Money

Load Save

per click
$68

per second
$1685

LEVEL UP $100	Brainstorming Lvl 10
LEVEL UP $67	Entrepreneurial Coach Lvl 1
LEVEL UP $615	Market Research Lvl 3
LEVEL UP $1822	Co-Founder Lvl 2
LEVEL UP $7290	Prototype Lvl 2
LEVEL UP $27K	Seed Funding Lvl 1

$112K

per click
$68

per second
$1685

Load Save

LEVEL UP
$100
Brainstorming
Lvl 10

LEVEL UP
$67
Entrepreneurial Coach
Lvl 1

LEVEL UP
$615
Market Research
Lvl 3

LEVEL UP
$1822
Co-Founder
Lvl 2

LEVEL UP
$7290
Prototype
Lvl 2

LEVEL UP
$27K
Seed Funding
Lvl 1

Ideas

$112K

Upgrades

per second
$1685

LEVEL UP $100	Brainstorming Lvl 10
LEVEL UP $67	Entrepreneurial Coach Lvl 1
LEVEL UP $615	Market Research Lvl 3
LEVEL UP $1822	Co-Founder Lvl 2
LEVEL UP $7290	Prototype Lvl 2
LEVEL UP $27K	Seed Funding Lvl 1

Load Save

$112K

per click
$68

per second
$1685

LEVEL UP $100	Brainstorming Lvl 10
LEVEL UP $67	Entrepreneurial Coach Lvl 1
LEVEL UP $615	Market Research Lvl 3
LEVEL UP $1822	Co-Founder Lvl 2
LEVEL UP $7290	Prototype Lvl 2
LEVEL UP $27K	Seed Funding Lvl 1

Load Save

Settings

$112K

per click
$68

per second
$1685

Load

Save

Revenues

LEVEL UP
$100

storming
Lvl 10

LEVEL UP
$67

Entrepreneurial Coach
Lvl 1

LEVEL UP
$615

Market Research
Lvl 3

LEVEL UP
$1822

Co-Founder
Lvl 2

LEVEL UP
$7290

Prototype
Lvl 2

LEVEL UP
$27K

Seed Funding
Lvl 1

$112K

per click
$68

per second
$1685

Load Save

LEVEL UP
$100

Brainstorming
Lvl 10

LEVEL UP
$67

Entrepreneurial Coach
Lvl 1

LEVEL UP
$615

Market Research
Lvl 3

LEVEL UP
$1822

Upgrades

LEVEL UP
$7290

Prototype
Lvl 2

LEVEL UP
$27K

Seed Funding
Lvl 1

$112K

per click
$68

per second
$1685

Load Save

LEVEL UP
$100
Brainstorming
Lvl 10

LEVEL UP
$67
Entrepreneurial Coach
Lvl 1

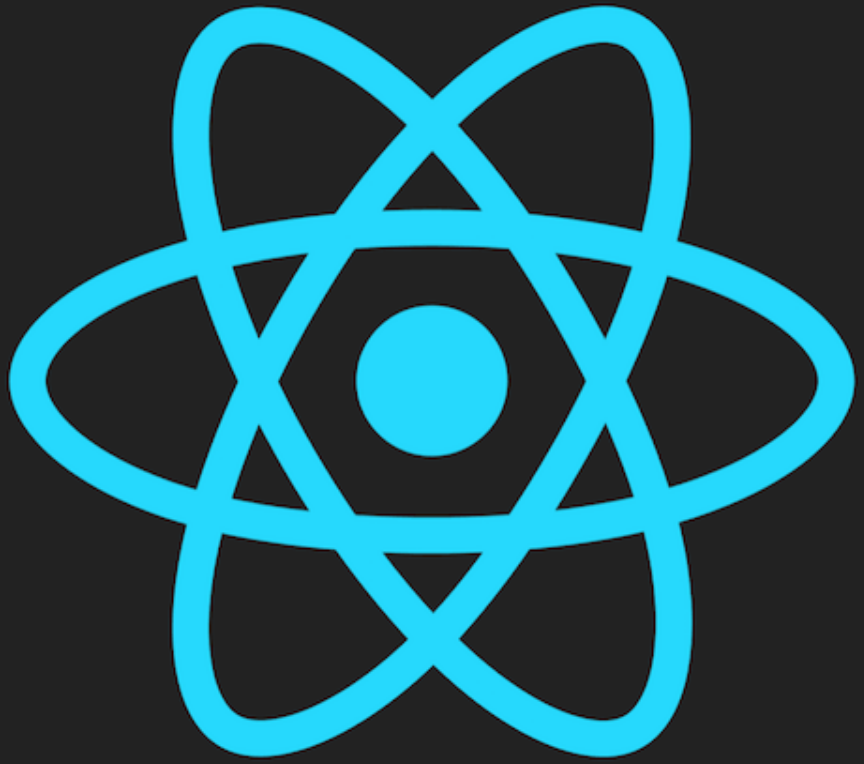
LEVEL UP
$615
Market Research
Lvl 3
Upgrade

LEVEL UP
$1822
Co-Founder
Lvl 2

LEVEL UP
$7290
Prototype
Lvl 2

LEVEL UP
$27K
Seed Funding
Lvl 1

Démo



React Native

Who's using React Native?

Thousands of apps are using React Native, from established Fortune 500 companies to hot new startups. If you're curious to see what can be accomplished with React Native, check out these apps!










Some of these are hybrid native/React Native apps.

```
import React, { Component } from 'react';
import { Text, View } from 'react-native';

class WhyReactNativeIsSoGreat extends Component {
  render() {
    return (
      <View>
        <Text>
          If you like React on the web, you'll like React Native.
        </Text>
        <Text>
          You just use native components like 'View' and 'Text',
          instead of web components like 'div' and 'span'.
        </Text>
      </View>
    );
  }
}
```

Text

A React component for displaying text.

`Text` supports nesting, styling, and touch handling.

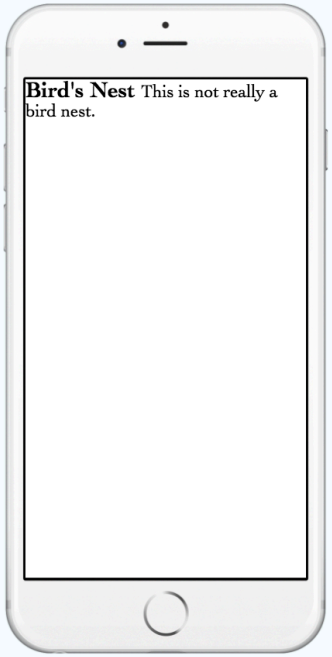
In the following example, the nested title and body text will inherit the `fontFamily` from `styles.baseText`, but the title provides its own additional styles. The title and body will stack on top of each other on account of the literal newlines:

```
import React, { Component } from 'react';
import { AppRegistry, Text, StyleSheet } from 'react-native';

class TextInANest extends Component {
  constructor(props) {
    super(props);
    this.state = {
      titleText: "Bird's Nest",
      bodyText: 'This is not really a bird nest.'
    };
  }

  render() {
    return (
      <Text style={styles.baseText}>
        <Text style={styles.titleText} onPress={this.onPressTitle}>
          {this.state.titleText}{'\n'}{'\n'}
        </Text>
        <Text numberOfLines={5}>
```

No Errors Show Details



Props

accessible bool

When set to `true`, indicates that the view is an accessibility element. The default value for a `Text` element is `true`.

See the Accessibility guide for more information.

ellipsizeMode enum('head', 'middle', 'tail', 'clip')

ReactJS

```
@MoneyClock.Clock = React.createClass(
  getInitialState: ->
    {
      startedAt:                  parseInt(@props.startedAt) || undefined
      startDuration:              parseInt(@props.duration)
      currency:                   @props.currency
      hourlyFee:                  parseInt(@props.hourlyFee)
      syncLoaded:                 false
      editTimeMode:               false
      editPriceMode:              false
      hours:                      0
      minutes:                    0
      seconds:                    0
      milliseconds:               0
      amountUnits:                0
      amountDecimals:             0
      tickIntervalId:             undefined
    }

  componentDidMount: ->
    @dOnChangeCallback = _.debounce(@onChangeCallback, 500)

    @tick()

    if @isPlaying()
      @startTimers()

  componentWillReceiveProps: (nextProps) ->
    if nextProps.startedAt != @props.startedAt
      @setState(
        startedAt:     parseInt(nextProps.startedAt) || undefined
        startDuration: parseInt(nextProps.duration)
        hourlyFee:     parseInt(nextProps.hourlyFee)
        syncLoaded:    false,
        =>
          @tick()

          if @isPlaying()
            @startTimers()
          else
            @clearTimers()
      )

  componentWillUnmount: ->
    @clearTimers()

  onChangeCallback: ->
    if @props.onChange
      @props.onChange(@state.startDuration, @state.startedAt, @state.hourlyFee)
```

React Native

```
class Clock extends Component {
  constructor(props) {
    super(props);

    this.state = {
      startedAt:      parseInt(props.startedAt) || undefined,
      startDuration:  parseInt(props.duration),
      currency:       props.currency,
      hourlyFee:      parseInt(props.hourlyFee),
      syncLoaded:     false,
      editTimeMode:   false,
      editPriceMode:  false,
      hours:          0,
      minutes:        0,
      seconds:        0,
      milliseconds:   0,
      amountUnits:    0,
      amountDecimals: 0,
      tickIntervalId: undefined
    };
  }

  componentDidMount() {
    this.dOnChangeCallback = _.debounce(this.onChangeCallback, 500)

    this.tick();

    if(this.isPlaying())
      this.startTimers();
  }

  componentWillReceiveProps(nextProps) {
    if(nextProps.startedAt != this.props.startedAt) {
      this.setState({
        startedAt:     parseInt(nextProps.startedAt) || undefined,
        startDuration: parseInt(nextProps.duration),
        hourlyFee:     parseInt(nextProps.hourlyFee),
        syncLoaded:    false
      }, () => {
        this.tick();

        if(this.isPlaying())
          this.startTimers();
        else
          this.clearTimers();
      })
    }
  }

  componentWillUnmount() {
    this.clearTimers();
  }

  onChangeCallback() {
    if(this.props.onChange) {
      this.props.onChange(this.state.startDuration, this.state.startedAt, this.state.hourlyFee);
    }
  }
```

```
import { StyleSheet } from 'react-native';

module.exports = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    height: 370,
    marginTop: 20
  },
  lastContainer: {
    marginBottom: 50,
  },
  card: {
    height: 250,
    width: 300,
    backgroundColor: '#ffffff',
    marginBottom: -330,
    borderWidth: 1,
    borderColor: '#ADB8B5',
  },
  title: {
    fontSize: 19,
    fontWeight: '300',
    color: '#878787',
    marginTop: 120,
    textAlign: 'center',
    height: 40,
  },
  description: {
    flex: 1,
    fontSize: 11,
    fontWeight: '300',
    color: '#333333',
    padding: 16,
    paddingTop: 2,
    textAlignVertical: 'top'
  },
});

```

```
import styles from './styles/ClockStyles'
```

```
<View style={styles.card}>
  <View style={styles.title}>
    ...
  </View>
</View>
```

React

Présentation de la librairie et prototypage d'un jeu de type "clicker"

http://80limit.com/slides/jeudis.pdf

Michaël Hoste

michael@80limit.com • @michaelhoste • github.com/michaelhoste