

# React & D3

Esteban Sotillo  
CPNV-ES

11 avril 2018

## 1 Introduction

Documentation de l'utilisation de React et D3 afin de pouvoir commencer un projet avec ces technologies de manières simple et rapides.

Pour faire ce projet j'ai travaillé sur une version de linux donc certaines manipulations risque de changer un peu. Les possibles manipulations supplémentaires sont laissées au lecteur.

## 2 Mise en place

### 2.1 Prérequis

- npm v5.x+
- Syntaxe ES6+
- React

### 2.2 Création de la base du projet

Pour la base du projet il est plus pratique d'utiliser le CLI de création d'app React afin d'avoir une base de code consistante avec tous les modules principaux.

```
1 npm i -g create-react-app
2 create-react-app my-app
3 cd my-app
```

## 2.3 Installation des dépendances

Pour ce projet j'ai utilisé différents packages dont j'expliquerais l'utilité dans les prochaines lignes.

<code>react-move</code>	Utilisé pour aider à l'animation de composants react.
<code>react-chord-diagram</code>	Composant react simplifiant l'utilisation du diagram chord de D3.
<code>d3-ease</code>	Librairie de fonctions servant à définir le flux des transitions.

Pour les installer il suffit d'entrer la commande suivante :

```
1 npm i react-move react-chord-diagram d3-ease
```

## 2.4 Structure de fichier

Voici un rapide descriptif des fichiers utiles pour commencer le projet.

```
1 + public : fichiers statics modifiable
2 + src : dossier de travail
3   - index.js : fichier js de base
4   - App.js : Composant react originel
```

# 3 Utilisation

Dans cette partie, je vais faire un petit projet afin d'expliquer plus facilement l'utilisation des composants.

## 3.1 Importation des composants

Dans le fichier `src/App.js` ajouter ces trois lignes au début du fichier.

```
1 import ChordDiagram from 'react-chord-diagram'
2 import { Animate } from 'react-move'
3 import { easeExpInOut } from 'd3-ease'
```

Puis supprimer le contenu de la première div.

## 3.2 Composant ChordDiagram

Le composant ChordDiagram possède 4 propriétés que nous allons utiliser qui sont "matrix", "componentId", "groupLabels" et "groupColors".

La propriété "matrix" est un tableau à 2 dimensions comprenant en x la quantité partant et en y la quantité de réception.

La propriété "componentId" est un ID pour reconnaître le diagramme.

La propriété "groupLabels" est un tableau de string utilisé comme label pour le diagramme.

La propriété "groupColors" est un tableau de couleurs css utilisé pour le diagramme.

Exemple :

```
1 <ChordDiagram
2   matrix=[[ [1,2] , [3,4] ] ]
3   componentId={ "ID" }
4   groupLabels=[ "foo", "boo" ]
5   groupColors=[ "#333", "#567" ]
6 />
```

## 3.3 Composant Animate

Le composant Animate sert à aider à l'animation des composants react. Il possède 2 propriétés requises qui sont "start" et "update". Ces deux propriétés sont des références de fonction définissant respectivement les status initiaux et les status après execution.

Exemple :

```
1 <Animate
2   start={() => ({
3     x: 0
4   })}
5   update={() => ({
6     x: [1],
7     timing: { duration: 1500, ease: easeExpInOut },
8   })}
9 >
```