

# Soutenance de stage

"Analyse des données en IoT"

- ▼ Entreprise d'accueil: Maya Technologies
- ▼ Durée: 6 mois
- ▼ Réalisé par: GANZA Mykhailo
- ▼ Responsable: Mme LECOMTE Christelle
- ▼ Encadrant: M NACABAL François



# Introduction

- ▼ Maya Technologies
- ▼ IoT (*Internet of Things*)
- ▼ MayaNet

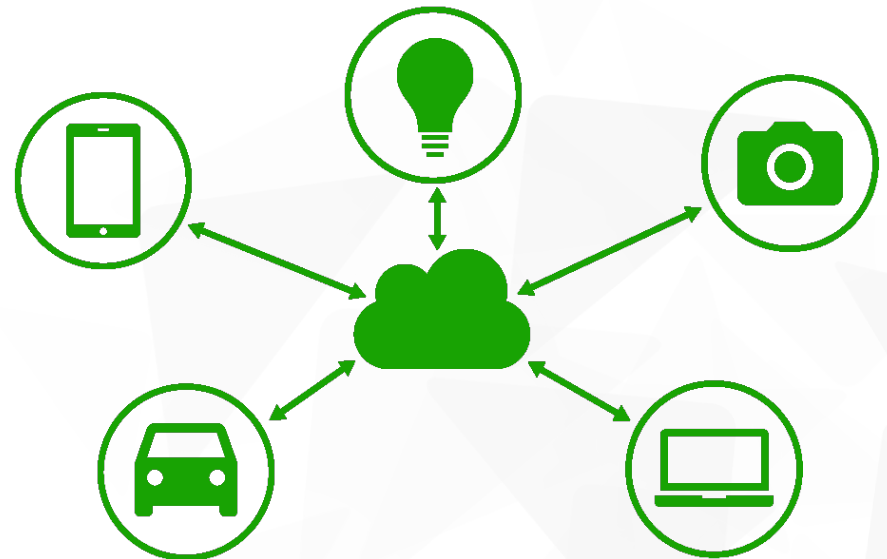
# Introduction: Maya Technologies

- ▼ Fondée en 2007 par Philippe Mattia à Grenoble
- ▼ Domaine: IoT, systèmes embarqués
- ▼ Spécialités: développement software, design hardware
- ▼ Industries: aéronautique, multimédia, santé
- ▼ Villes: Grenoble, Valence, Aix-en-Provence, Paris, Toulouse



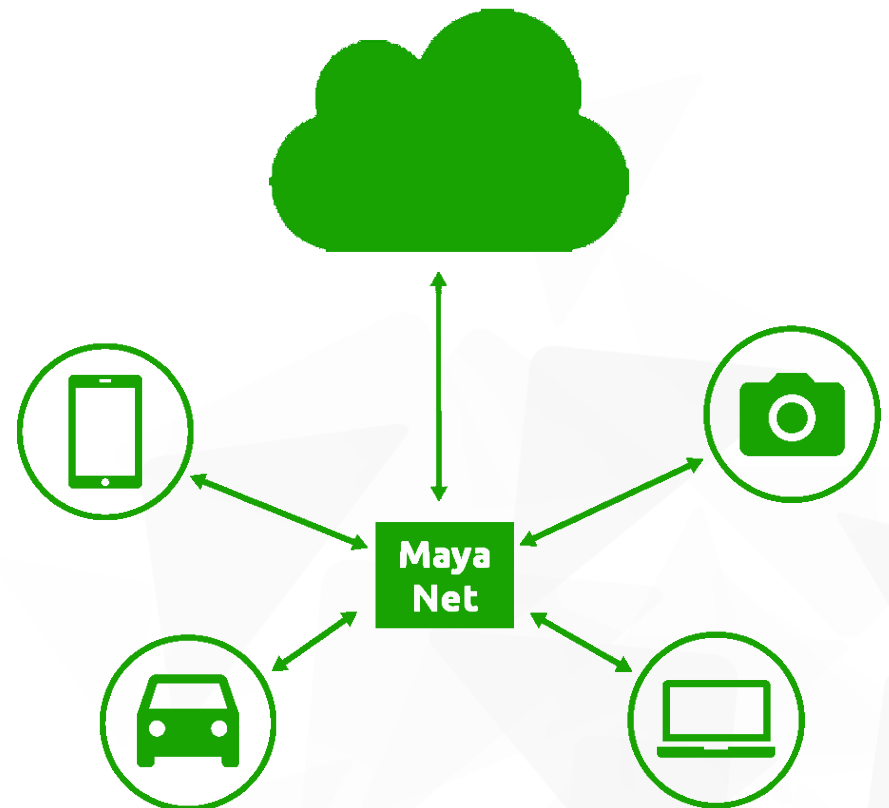
# Introduction: IoT

- ▼ **IoT** (“Internet Of Things”), les objets connectés, est une extension d'internet vers le monde physique grâce aux systèmes embarqués
- ▼ **20.8** milliards des objets connectés prévus pour 2020
- ▼ Utilisation:
  - ▼ médecine (“santé connectée”)
  - ▼ domotique
  - ▼ capteurs environnementaux
  - ▼ etc



# Introduction: MayaNet

- ▼ Une solution IoT de Maya Technologies pour la **collecte, le traitement et l'analyse des données** provenant de différents types d'objets connectés
- ▼ Un logiciel de gestion d'un réseau de systèmes embarqués
- ▼ Protocoles supportés:  
GSM, Wi-Fi, KNX, USB, etc

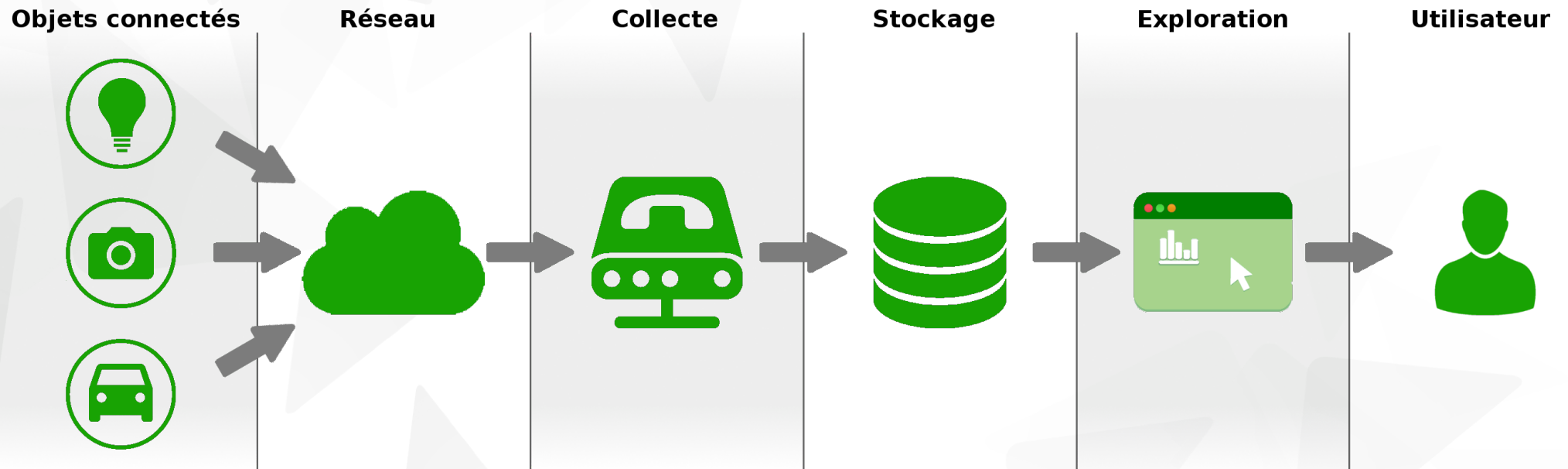


# Objectif initial

# Objectif initial

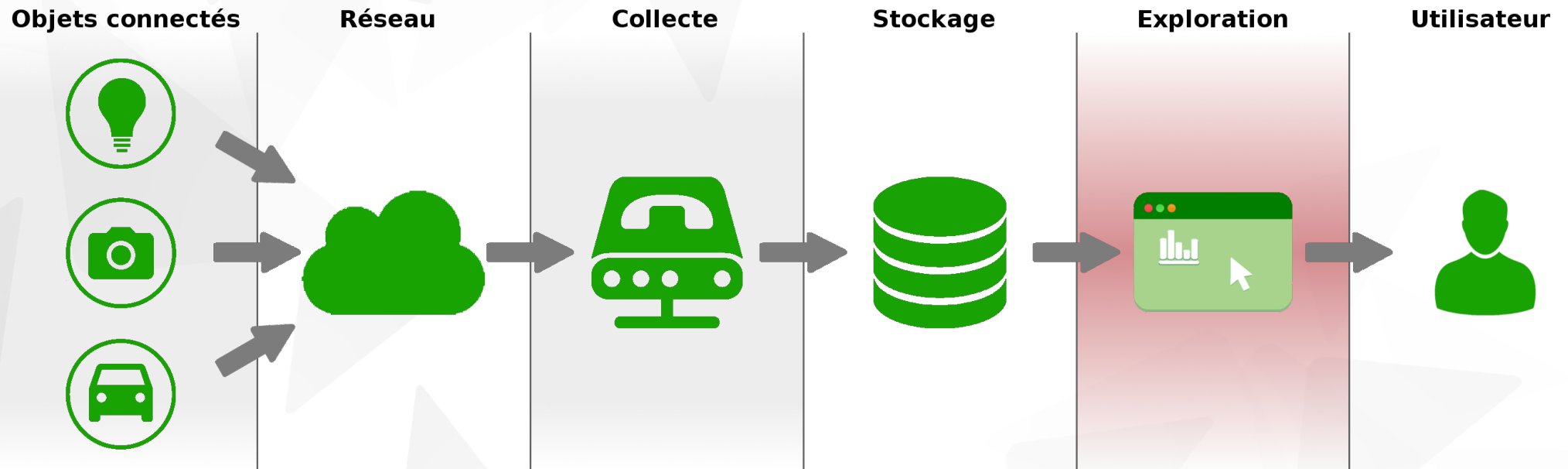
« Concevoir un outil *d'exploration* des *données* et des *relations* entre les données provenant de sources différentes d'objets connectés »

# Objectif initial (2)





# Objectif initial (2)



# Étude

- ▼ Séries temporelles
- ▼ Exploration des données
- ▼ DC.js
- ▼ Objectif concret

# Étude: séries temporelles

- ▼ Une **série temporelle** (eng: *time series*) est une **suite des valeurs numériques** évoluant dans le temps
- ▼ Composé d'un ensemble de **paires clef-valeur**, avec une clef spécifique d'horodatage (eng: *timestamp*)
- ▼ **Sources possibles**: bases de données (ex: *MySQL*, *MongoDB*), fichiers (ex: *JSON*, *CSV*), etc

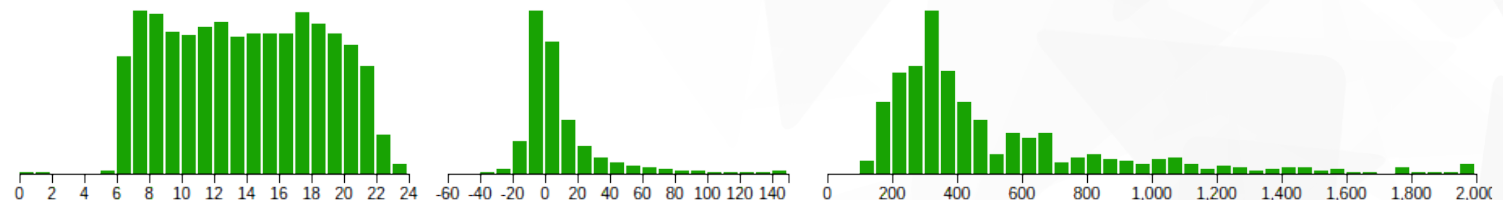
- ▼ Adapté pour l'IoT

```
"year","flows_colorado"  
"1911",18.11  
"1912",21.07  
"1913",15.77  
"1914",24.17  
"1915",14.71
```

```
[  
  {"year": 1911, "flows_colorado": 18.11},  
  {"year": 1912, "flows_colorado": 21.07},  
  {"year": 1913, "flows_colorado": 15.77},  
  {"year": 1914, "flows_colorado": 24.17},  
  {"year": 1915, "flows_colorado": 14.71}  
]
```

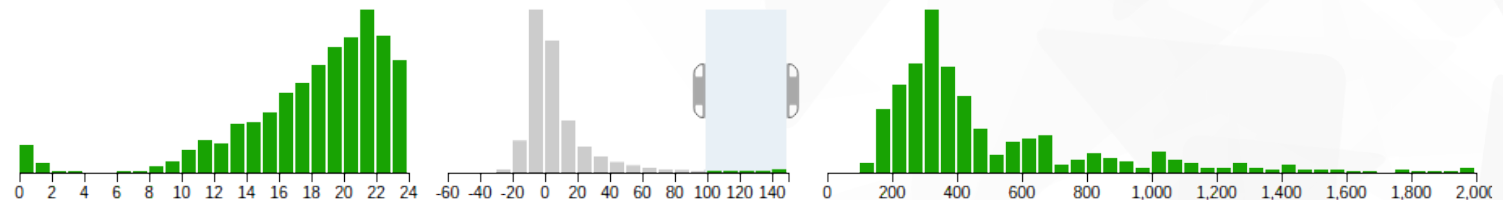
# Étude: exploration des données

- ▼ Observation des données acquises, afin d'en tirer des conclusions
- ▼ Utilisation des graphiques
- ▼ Utilisation des caractéristiques statistiques simples
- ▼ “Relation entre les données”: étude de l'influence de l'évolution d'une valeur numérique sur l'évolution d'une autre valeur numérique



# Étude: DC.js

- ▼ Une bibliothèque **JavaScript** pour la génération de **graphiques dynamiques** à partir de données
- ▼ Basée sur **D3.js** et **Crossfilter.js**
- ▼ Permet d'utiliser des **filtres multidirectionnels** afin d'explorer les données, et les relations entre les données



# Étude: Objectif concret

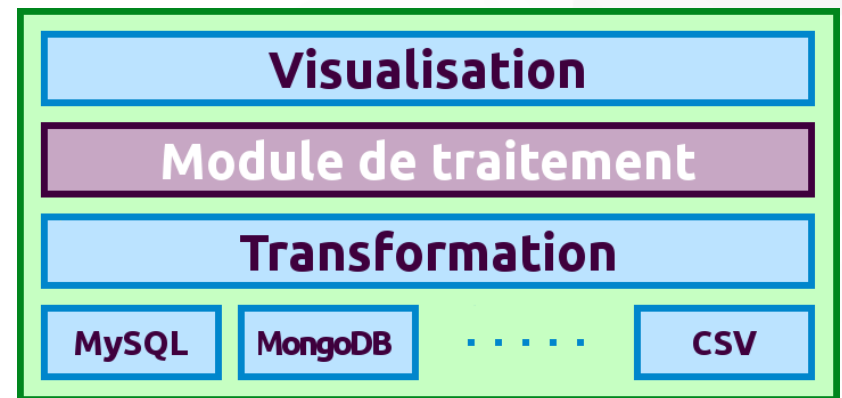
« Concevoir un *outil graphique* d'exploration des *séries temporelles* provenant de différentes sources d'objets connectés »

# Travail réalisé

- ▼ Deux modules
- ▼ Time Series Processor - traitement des séries temporelles
- ▼ Data Exploration - outil graphique

# Travail réalisé: deux modules

- ▼ L'outil est composé de **deux parties** indépendantes:
  - ▼ Un module de **traitement** des séries temporelles
  - ▼ Un module de **visualisation**
- ▼ **Motivation principale**: un module qui traite les données d'un format fixe, et une surcouche qui permet de transformer depuis d'autres formats
- ▼ **Autres motivations**:  
modularité, ré-utilisabilité





# Travail réalisé: tsproc

- ▼ Time-Series Processor (tsproc) est une **boîte à outils** de traitement des séries temporelles
- ▼ Écrit en: **JavaScript** (Node.js)
- ▼ Format des séries temporelles acceptées: **JSON**
- ▼ Traitement des séries temporelles possibles:

*sous-échantillonnage*

*fusion de plusieurs vers une*

*interpolation des données manquantes*

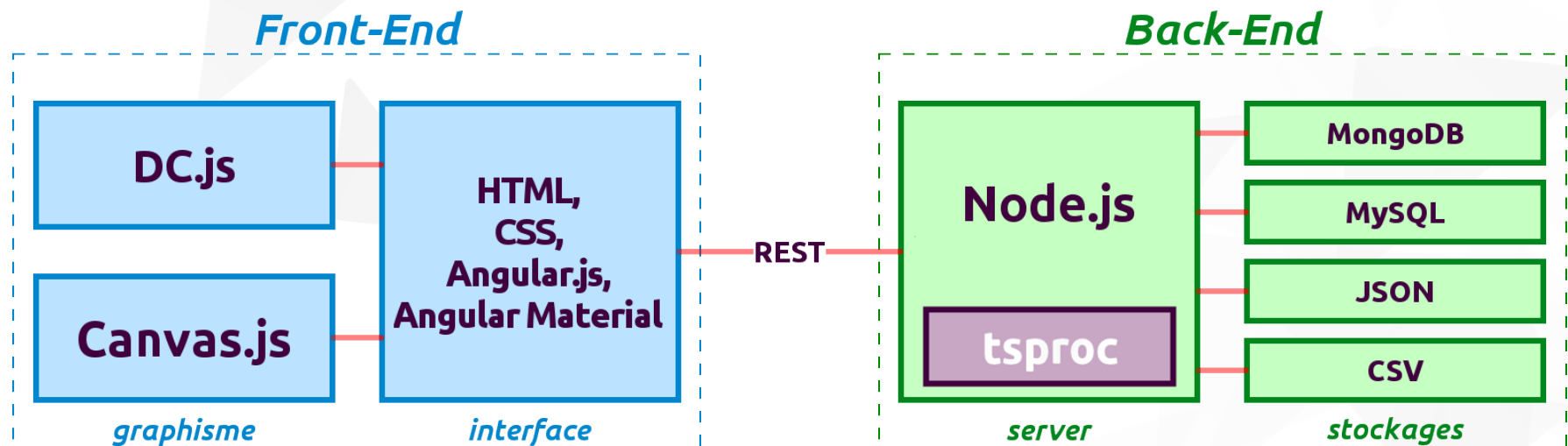
*quantification*

*détection de corrélation*

*etc*

# Travail réalisé: dataexp

- ▼ Data Exploration (dataexp): un outil graphique d'exploration des données
- ▼ Interface web
- ▼ Architecture à deux tiers:



# Demonstration ...

# Résultats

- ▼ Deux modules écrits en JavaScript
- ▼ Exploration des données via DC.js
- ▼ Supporte plusieurs sources
- ▼ Testé pour: Mozilla, Chrome
- ▼ Lignes de code:

	Module de traitement ( <i>tsproc</i> )	Application graphique ( <i>dataexp</i> )
<i>Node.js</i>	954	867
<i>Angular.js</i>		1439
<i>HTML</i>		638
<i>Documentation</i>	234	513
<i>Totale:</i>	4645	

# Résultats (2)

- ▼ Complexités / Barrières / Difficultés / Obstacles / Subtilités:
  - ▼ objectif générique → objectif concret
  - ▼ multitude de formats temporels
- ▼ Perspectives:
  - ▼ ajouter d'autres sources
  - ▼ détection automatique de format de timestamp



**Merci pour votre attention**

# Sitographie

- ▼ <https://material.angularjs.org/latest/> - module de Angular.js utilisé pour l'interface
- ▼ <https://github.com/dc-js/dc.js/wiki> - graphiques DC.js
- ▼ <http://stackoverflow.com/> - débogage