Paul-Arthur THIÉRY, Alexandre Van Wesel

Project Report

1. Data integration

We use a NodeJS script to spawn a MongoDB client (mongoose), then used a CSV parser to parse the CSV files and add them to our collections.
To get the original CSV files, we ran a curl function on the url to download the CSV files in a directory.

2. Data Modeling

We used 2 collections (sites and consumptions), which contain respectively :

- The site information contained in all_sites.csv
- The consumption information contained in all files with the id of the site corresponding to each consumption (embedded parent model)

3. Queries

1) Simple Queries

```
db.consumptions.findOne({timestamp: 1325382900})
db.sites.find({})
db.consumptions.update({value: 20.9518}, {timestamp: 1324628200, dttm_utc: 2015-01-09, value: 22.2222, estimated: 0, anomaly: "", site: [6]})
db.consumptions.find().sort( { sites: 1 } )
db.sites.find({siteId:{$nin:db.consumptions.find({value: 82.0151}).siteId}})
```

2) Calculate the sum LD for the 100 sites (timestamp interval: 5 minutes)

```
db.consumptions.aggregate( [ { "$limit": 100 }, {"$group": {_id: "$consumptions.site", count: {$sum :"$value"} } }] )
```

3) Calculate the average LD by sector of activity

```
db.consumptions.aggregate( [ {$limit: 10}, { $unwind: "$site" }, { $lookup: {from: "sites",
```

```
localField: "site", foreignField: "_id", as: "current_site"} }] )
```

Github Repository : https://github.com/paul-arthurthiery/BigDataProject