Fait par:

* Ferol Tatang Fomekon
* Fares

|  |  |
| --- | --- |
| Rapport du projet TPT : Lacs de Données et les tables externes |  |

Table des matières

[Introduction 2](#_Toc129560671)

[Chapitre 1 2](#_Toc129560672)

[I. Le descriptif du projet 2](#_Toc129560673)

[II. L’architecture du projet avec la ventilation des données par base ou système 2](#_Toc129560674)

[III. Le descriptif de votre démarche pour l’adaptation du fichier CO2.csv et son intégration dans la table catalogue incluant les commandes / programmes que vous avez exécutés et leur résultat / output 3](#_Toc129560675)

[Chapitre 2 : Scripts pour l’architecture 5](#_Toc129560676)

[I. Le script de création de documents et de chargement de données dans MongoDB, le script de création des tables externes MongoDB dans Hive et enfin le script de création des tables ex ternes correspondantes dans Oracle NoSQL. 5](#_Toc129560677)

[II. Le script de création de tables dans Oracle NoSQL 6](#_Toc129560678)

[III. Le script de création des tables externes correspondantes Hadoop HDFS dans Hive 6](#_Toc129560679)

[IV. Le programme java de chargement des données dans la base Oracle NoSQL. 7](#_Toc129560680)

[V. Le script de chargement des données dans Hadoop HDFS 8](#_Toc129560681)

[VI. Code source du programme Hadoop MapReduce utilisé pour l’adaptation du fichier CO2.csv et son intégration dans la table Catalogue. 8](#_Toc129560682)

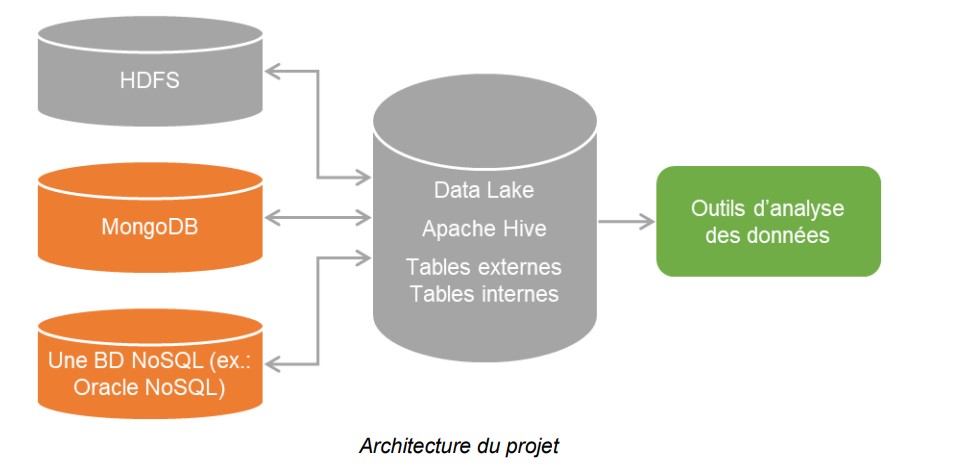
[Conclusion 8](#_Toc129560683)

[REFERENCE 8](#_Toc129560684)

# 

# Introduction

Ce projet consiste à aider un concessionnaire automobile à mieux cibler les véhicules susceptibles d'intéresser ses clients. Pour cela, nous avons accès à son catalogue de véhicules, son fichier clients de l'année en cours, ainsi qu'à toutes les informations sur les immatriculations effectuées cette année. Notre objectif est de proposer un outil qui permettra d'évaluer en temps réel le type de véhicule le plus susceptible d'intéresser les clients qui se présentent dans la concession et d'envoyer une documentation précise sur le véhicule le plus adéquat pour des clients sélectionnés par son service marketing. Mais dans ce rapport, nous allons nous intéresser à la construction de l’architecture de DATA LAKE mixte autour de Hive comme le présente la figure 1(Architecture du projet) avec pour objectif de traiter le fichier co2.cvs grâce aux rdd (resilient Distributed dataset) et d’intégrer les colonnes intéressantes dans le fichier catalogue.



# Chapitre 1

## Le descriptif du projet

Le projet consiste à construire un Data Lake en utilisant des outils de traitement de données tels que MongoDB, Hive, Oracle NoSQL, HDFS et Hadoop. Les données seront réparties entre différentes bases de données et pourront être manipulées via HiveSQL. Le Concessionnaire demande ensuite d'adapter un fichier CO2.csv pour intégrer des informations manquantes dans la table Catalogue du Concessionnaire. Pour cela, il faudra écrire un programme MapReduce avec Hadoop ou Spark. Le but est d'améliorer la qualité des modèles de Machine Learning en disposant de données plus complètes.

## L’architecture du projet avec la ventilation des données par base ou système

Voici l’architecture du projet avec la ventilation des données par base ou système :

ORACLE NoSQL : contient les données « catalogue »

MongoDB : contient les données « Marketing »

Hadoop HDFS : contient des données « Immatriculation , CO2 , et Clients\_31 »   sous forme de fichiers

Hive : contient une table interne avec les données du fichier " Clients\_32", ainsi que des tables externes et internes pour accéder aux autres sources de données dans les différentes bases. Les données doivent être manipulées avec SQL.

Data lake : construit avec l'aide de Spark et des drivers pour se connecter aux différentes bases et systèmes de traitement. Stocke et analyse les données disponibles dans les différentes sources.

Fichier CO2.csv : utilisé pour adapter les informations complémentaires dans la table Catalogue du Concessionnaire grâce à un programme map/reduce avec Spark.

## Le descriptif de votre démarche pour l’adaptation du fichier CO2.csv et son intégration dans la table catalogue incluant les commandes / programmes que vous avez exécutés et leur résultat / output

La démarche pour adapter le fichier CO2.csv et l'intégrer dans la table catalogue est la suivante :

Chargement du fichier CO2.csv dans HDFS en utilisant la commande hadoop fs -put /vagrant/TPT/CO2.csv.

Exploration et nettoyage du fichier CO2.csv pour supprimer les valeurs manquantes et erronées.

Une image contenant texte, reçu, capture d’écran

Description générée automatiquementUne image contenant texte

Description générée automatiquement

Une image contenant table

Description générée automatiquement

Le code est a récupérer dans le noteook

Création d'une table externe dans Hive :

Une image contenant texte

Description générée automatiquement

Table externe CO2 dans hive

Une image contenant texte

Description générée automatiquement

Table externe Catalogue dans hive

Création de la connexion dans le jupyter notebook avec hive et récupérer la tables catalogues enfin de faire la jointure.

# Chapitre 2 : Scripts pour l’architecture

## Le script de création de documents et de chargement de données dans MongoDB, le script de création des tables externes MongoDB dans Hive et enfin le script de création des tables ex ternes correspondantes dans Oracle NoSQL.

Script pour créer des documents et charger des données dans MongoDB :

Nous avons aux préalables convertis le documents Immatriculation.cvs en json ensuite nous avons exécuter ce code suivant ;

Une image contenant texte

Description générée automatiquement

Pour créer la table externe MongoDB dans Hive, nous avons utilisé le script suivant :

Une image contenant texte

Description générée automatiquement

Script de créations des tables externes correspondantes à Oracle Nosql dans Hive :

Une image contenant texte

Description générée automatiquement

## Le script de création de tables dans Oracle NoSQL

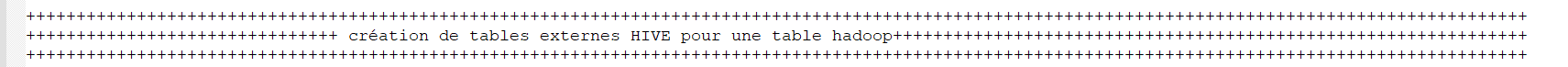
Script de création de tables dans Oracle Nosql :

Une image contenant texte

Description générée automatiquement

## Le script de création des tables externes correspondantes Hadoop HDFS dans Hive

Script de création de tables externes correspondantes Hadoop hdfs dans Hive :



Une image contenant texte

Description générée automatiquement

Table externe Hive pour Client\_31

Une image contenant texte

Description générée automatiquement

Table externe Hive pour C02

Une image contenant texte

Description générée automatiquement

Table externe Hive pour Immatriculation

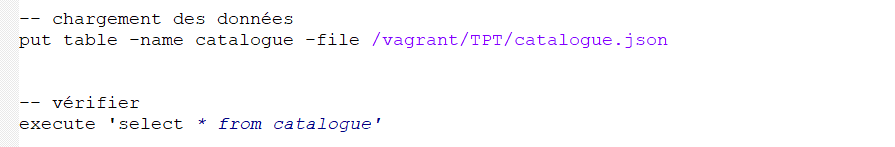
## Le programme java de chargement des données dans la base Oracle NoSQL.

Connection

Une image contenant texte

Description générée automatiquement

Chargement des donnees :

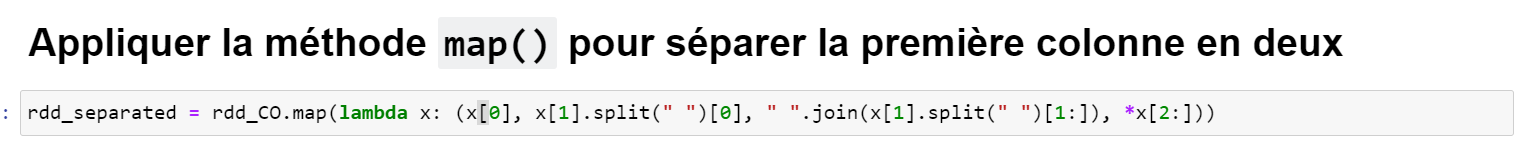


## Le script de chargement des données dans Hadoop HDFS

Une image contenant texte

Description générée automatiquement

## Code source du programme Hadoop MapReduce utilisé pour l’adaptation du fichier CO2.csv et son intégration dans la table Catalogue.



# 

# Conclusion

# Nous avons réussi à remplir les objectifs de notre rapport, qui consistaient à mettre en place un data lake sous Hive, à traiter les données du fichier co2.csv à l'aide de RDD et à fusionner certaines colonnes de ce fichier avec le fichier de catalogue. Nous pouvons conclure que ces différentes tâches ont été menées à bien.

# REFERENCE

[Stack Overflow - Where Developers Learn, Share, & Build Careers](https://stackoverflow.com/)

[ChatGPT - Advanced AI Chatbot by OpenAI (chat-gpt.org)](https://chat-gpt.org/)

[SergioSim/hadoop\_hive\_spark (github.com)](https://github.com/SergioSim/hadoop_hive_spark)

# NOTES IMPORTANTES

* Je joins :
* Ce rapport
* Mon code : TPT\_VersionFinal.md
* Le lien vers mon driver avec les vidéos :

https://drive.google.com/drive/folders/1eryRWSglSa5vgkQV\_DILufUXbd6tA3uT?usp=share\_link