RESUME

L'Internet des Objets (IdO) comprendra des milliards de dispositifs qui pourront détecter, communiquer, calculer et potentiellement actionner. Les flux de données provenant de ces dispositifs vont défier les approches traditionnelles de gestion des données. Dans ce contexte, il est primordial d'explorer de nouvelles méthodes d'analyse, de gestion et de stockage des données en s'appuyant sur le paradigme naissant du Big data. Beaucoup de conversations qui ont lieu autour de l'Internet des Objets sont incomplètes sans une mention de Big data. "Le succès ou l'échec de l'Internet des Objets repose sur le Big data ", d'après Brian Hopkins [1], analyste chez Forrester[1]. Dans ce rapport, nous nous intéressons à la problématique de gestion de données dans l'Internet des Objets et aux perspectives prometteuses qu'offre l'utilisation des Technologies de Big data dans ce contexte, plus particulièrement à l'aide de la plateforme Hadoop. Notre objectif est de proposer une méthodologie pour la mise en œuvre d'une solution de traitement de données volumineuses et peu structurées basée sur Hadoop et ses différentes briques logicielles.

Mots clés

Internet des Objets, Big data ,Hadoop, Cloudera CDH

ABSTARCT

The Internet of Things (IoT) will include billions of devices that can detect, communicate, compute and potentially operate. The flow of data from these devices will challenge traditional approaches of data management. In this context, it is essential to explore new methods of data analysis, management and storage based on the emerging big data paradigm . Many conversations that take place around the Internet of Things (IoT) are incomplete without a mention of Big data. Indeed, according to Brian Hopkins, Forrester[2]analyst : "The success or failure of the Internet of Things is based on big data,". In this report, we focus on data management problematic in the Internet of Things and the promising prospects offered by the use of big data technologies in this context especially within hadoop platform.Our goal is to propose a methodology for the implementation of a solution processing

large and unstructured data based on Hadoop and its various software components

Keywords

Internet of Things, Big data ,Hadoop, Cloudera CDH

[1] Forrester est une entreprise indépendante qui fournit à ses clients des études de marché sur l'impact des technologies dans le monde des affaires

[2] Forrester is an independent company that provides its clients with market studies on the impact of technology in the business world

DEDICACES

Je dédie ce mémoire :

À ma très chère mère pour tout son amour
À mon père qui m'a donné un magnifique modèle de persévérance
À mes chers frères pour leur encouragement indéfectible
À mes adorables sœurs pour leur soutien moral tout au long de mon memoire
À tous mes amis et à tous mes chers
J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour

Mimane Nafla

REMERCIEMENT

Je commencerais par remercier sincèrement ***Madame Meriem Abid*** *, qui, en tant que Encadrante , s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce travail, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu me consacrer et sans qui ce mémoire n'aurait jamais vu le jour. et j'aimerais aussi que mes remerciements s'adressent également à tous ceux qui m' ont soutenu tant moralement que matériellement.*

SOMMAIRE

CHAPITRE III : Installation et prise en main de Hadoop

CHAPITRE IV :Cas réel d'application du Big data

LISTE DES FIGURES

LISTE DES ABREVIATION

Abréviation	Désignation
IdO	**I**nternet **d**es **O**bjets
IoT	**I**nternet **o**f **T**hings
UIT	**U**nion **I**nternationale des **T**élécommunications
GPS	**S**ystème de **P**ositionnement **G**lobal
RFID	**R**adio **F**requency **I**dentification **D**evices
NTIC	**N**ouvelles **T**echnologies de l'**I**nformation et de la **C**ommunication
SGBD	**S**ystèmes de **G**estion de **B**ases de **D**onnées
IEEE	**I**nstitute of **E**lectrical and **E**lectronics **E**ngineers
NASA	**N**ational **A**eronautics and **S**pace **A**dministration
IoE	**I**nternet **o**f **E**verything
UPS	**U**nited **P**arcel **S**ervice
ORION	**O**n-**R**oad **I**ntegrated **O**ptimization and **N**avigation
DIKW	**D**ata, **I**nformation, **K**nowledge, **W**isdom
GFS	**G**oogle **F**ile **S**ystem
PLC	**P**ower **L**ine **C**ommunication
3V	**V**olume, **V**ariété, **V**élocité
NoSQL	**N**ot **O**nly **SQL**
NDFS	**N**utch **D**istributed **F**ile **S**ystem
HDFS	**H**adoop **D**istributed **F**ile **S**ystem

INTRODUCTION GENERALE

L'Internet des Objets est devenu un phénomène réel durant ces dernières années, elle est promis à un bel avenir, selon le cabinet d'étude Gartner [4], ce secteur devrait peser 1 900 milliards de dollars et compter plus de 30 milliards d'objets connectés à l'horizon 2020.

En 2010, la quantité totale de données sur la terre a dépassé un zettaoctets (1 Zo $=10^{21}$ octets), À la fin de 2011, le nombre a augmenté jusqu'à 1,8 Zo. En outre, il est prévu que ce nombre va atteindre 35 Zo en 2020 [15].

Parmi le flot infini de données amenées à être échangées, celles issues des objets connectés, selon plusieurs études [2], [3] menés par de nombreuses entreprises et cabinets d'études dans le domaine des technologies de l'information et de la communication[3], l'Internet des Objets contribuera ***« à doubler la taille de l'univers numérique tous les deux ans, lequel devrait peser 44.000 milliards de giga-octets en 2020, soit 10 fois plus qu'en 2013»***.

mais les techniques actuelles pour le traitement de ces données massives (Téraoctets vers Zettaoctets), structurées ou non structurées, sont limitées et inadaptées pour traiter ces milliards de giga-octets, d'une manière ou d'une autre.

L'internet des Objets va également accélérer l'émergence de gisements de données personnelles issues de l'accumulation de toutes les traces numériques, qui représenteront un immense potentiel d'innovation et demanderont un soin tout particulier sur leur conservation et leur protection .

À l'échelle de l'individu, ils vont modifier l'environnement et les pratiques des personnes, dans de nombreux domaines en particulier celui de la santé, du bien-être et de la forme physique, de la voiture, du domicile et de la sécurité.

À l'échelle d'un quartier, d'une ville, d'espaces ruraux ou d'une région, ils vont optimiser la gestion de l'énergie, fluidifier les transports, contribuer à la sécurisation d'un lieu, apporter de l'information pertinente et contextuelle aux personnes [39].

[3] EMC : entreprise américaine de logiciels et de systèmes de stockage
IDC : International Data Corporation » l'entreprise américaine spécialisée dans la réalisation des études de marché dans les domaines des technologies de l'information et de la communication

Dans le cadre de ce mémoire de Master, nous nous intéressons à la gestion des données massives, très hétérogènes et pas ou peu structurées de l'Internet des Objets,. Ce mémoire comporte quatre chapitres:

- Dans le premier chapitre, une présentation générale de **l'Internet des Objets** est donnée, et des principales causes de l'émergence on passe à la problématique de gestion de ses données massives et hétérogènes

- Dans le deuxième chapitre, on va plonger dans l'océan des **Big data** qui explore de nouvelles méthodes d'analyse, de traitement et de stockage des données afin d'offrir des services toujours plus performants.

- Dans le troisième chapitre, on va montrer que la vague du Big Data n'a été rendue possible que par une démocratisation des outils rendant de plus en plus accessible le traitement massif de données, et pour cela on va choisir l'incontournable outil **Hadoop**

- Dans le quatrième chapitre , on va présenter les résultats sur **Cloudera** (une des distribution de Hadoop)

CHAPITRE I : L'INETRNET DES OBJETS

I.1. Introduction

I.2. L’Internet des Objets

I.3. Domaines d’applications de l'Internet des Objets

I.4. La Problématique : les données générées par l’Internet des Objets.

I.5. Conclusion

CHAPITRE I : L'Internet des Objets

I.1. Introduction

Le terme « Internet des objets » n'est pas nouveau. Il y a près de 20 ans, des professeurs du MIT[4] décrivaient un univers peuplé d'« objets » (appareils ou capteurs) connectés permettant le partage de données. Les données en provenance de ces appareils et capteurs fournissent des informations stratégiques jusque-là inaccessibles. Ce sont ces précieuses informations, tirées de l'exploitation et de l'analyse des données issues de ces appareils connectés, qui font tout l'intérêt de l'Internet des objets.[41]

I.2. L'Internet des Objets

L'Union Internationale des Télécommunications (UIT) définit l'Internet des Objets comme une extension de l'Internet tel que nous le connaissons aujourd'hui, par la création d'un réseau omniprésent et auto-organisé d'objets physiques connectés, identifiables et adressables permettant le développement d'applications au sein de secteurs verticaux clés et entre ces secteurs par le biais de puces intégrées [3].

L'Internet des Objets est un réseau de réseaux qui permet, via des systèmes d'identification électronique normalisés et unifiés, et des dispositifs mobiles sans fil, d'identifier directement et sans ambiguïté des entités numériques et des objets physiques et ainsi de pouvoir récupérer, stocker, transférer et traiter, sans discontinuité entre les mondes physiques et virtuels, les données s'y rattachant [6]. Le potentiel de objets qui pourraient être connectées d'ici 2020 est estimé entre 30 et 212 milliards selon de nombreuses études (Gartner, Berg Insight, Cisco ISBG, Idate, IDC) comme le présente la Figure I.1 [3]. Ces objets vont contribuer à la transformation digitale de très nombreux métiers, non seulement par leur usage mais aussi et surtout, parce que les données qu'ils produisent vont être à la source de changements profonds et de la naissance de services inédits [39].

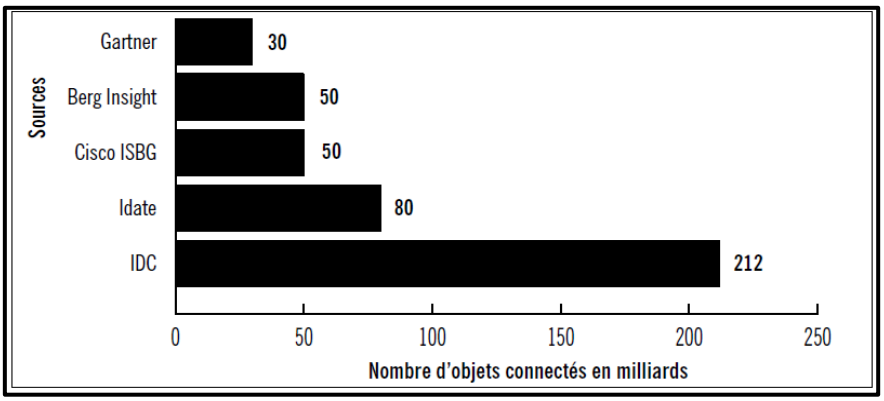


Figure I.1: Estimation du nombre d'objets connectés en 2020 [3].

L'Internet des Objets fait référence à un réseau qui interconnecterait l'ensemble des objets en leur donnant la capacité de communiquer entre eux par l'intermédiaire d'Internet pour échanger des informations (sur leurs identités, leurs caractéristiques physiques, leur environnement…) ou pour réagir à des commandes à distance. Toutes les entités connectées

[4] MIT Institut de Technologie du Massachusetts

peuvent être considérées comme des objets : Smartphone, capteur, GPS, réfrigérateur, cardio-fréquencemètre…Un objet connecté intègre au minimum un module de communication et éventuellement un capteur, un actionneur, une unité de traitement et de la mémoire [7].

I.2.1. Au fond, qu'est-ce que l'Internet des Objets ?

Pour savoir ce qu'est, en définitive, cet Internet des Objets dont nous parlons, il est important de comprendre ce qu'il n'est pas. Les contributeurs nous l'ont fait comprendre, chacun à sa manière: ce n'est pas réellement une révolution technique. C'est plutôt le produit de mutations qui conjuguent leurs effets: l'Internet, le cloud, les technologies de transmission sans fil, le Big Data… Elles se combinent à une mutation des mentalités: les individus ont pris conscience, aujourd'hui, du fait qu'ils formaient, eux aussi, un réseau. Les médias sociaux ont joué un grand rôle dans cette prise de conscience. Enfin, le troisième élément, c'est une révolution des usages. Là où il y a encore 20 ans, l'ordinateur était une chose qui avait sa place attitrée au bureau ou dans le foyer, nous en transportons tous maintenant avec nous en permanence, sous la forme de smartphones, tablettes et désormais montres ou bracelets connectés. On pourrait résumer tout ceci en une simple formule [40]:

Internet des Objets =
Mutations des technologies * Mutation des mentalités * Mutation des usages

Il s'agit bien d'une multiplication et non d'une simple somme : tous ces phénomènes démultiplient mutuellement leurs effets.

I.2.2. Architecture de l'internet des Objets

L'Internet des Objets représente l'interconnexion des différents équipements utilisés dans la vie quotidienne et intégrés dans l'Internet. Il vise à automatiser le fonctionnement de différents domaines tels que les appareils ménagers, les systèmes de soins de la santé, les systèmes de sécurité et de surveillance, les systèmes industriels, les systèmes de transport, les systèmes militaires, les systèmes électriques, et beaucoup d'autres. Afin d'obtenir un processus entièrement automatisé, les dispositifs dans les différents domaines doivent être équipés avec des microcontrôleurs, des émetteurs-récepteurs, et des protocoles pour faciliter et standardiser leur communication entre eux ou avec des entités externes. L'architecture de l'Internet des Objets est composé de trois couches [8] : la couche perception, la couche réseau, et la couche application comme le représente la Figure I.2.

I.2.2.1. La couche perception (The perception layer)

Cette couche comprend un groupe d'appareils compatibles avec l'Internet qui peuvent percevoir, détecter, recueillir, et échanger des informations avec d'autres dispositifs par l'intermédiaire des réseaux de communication. Les capteurs, Les GPS, les caméras, et les RFID (Radio Frequency Identification Devices) sont des exemples de dispositifs qui existent au niveau de la couche perception.

I.2.2.2. La couche réseau (The network layer)

Cette couche est responsable de la transmission des données de la couche perception vers la couche application sous des contraintes liées à la capacité des dispositifs, la limitation du réseau et les contraintes des applications. Les systèmes de l' Internet des Objets utilisent une combinaison de réseaux à courte portée et à longue portée. Les technologies de communication à courte portée tels que Bluetooth et ZigBee, sont utilisées pour transporter l'information à partir d'appareils de perception à une passerelle à proximité. D'autres technologies portent l'information sur de longues distances telles que le Wifi, 2G, 3G, 4G, et PLC (Power Line Communication) la technique de communication par courants porteurs sur le réseau électrique.

I.2.2.3. La couche application (The application layer)

La couche supérieure est la couche application, où l'information entrante est traitée pour induire des idées sur lesquelles nous pouvons concevoir de meilleures stratégies de distribution et de gestion de puissance. Les applications visent à créer des maisons intelligentes, des villes intelligentes, de la surveillance du système de puissance, de la gestion de l'énergie sur la demande, de la coordination de stockage d'énergie distribuée, et de l'intégration de générateurs d'énergie renouvelable.

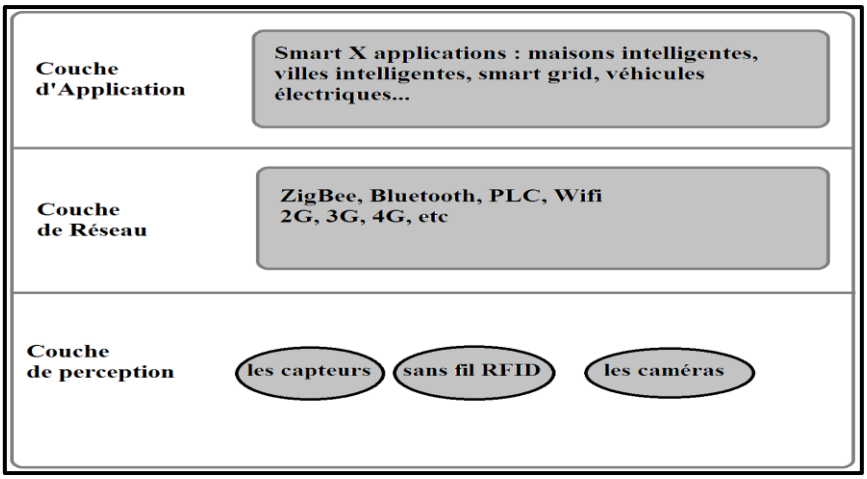


Figure I.2: Architecture de l' Internet des Objets [8].

I.3. Domaines d'applications de l'Internet des Objets

Au total, le cabinet McKinsey [2]évalue l'impact de l'Internet des Objets à 6,2 trillions (milliards de milliards) de dollars en 2025. Le cabinet estime qu'il y aura 9 sujets touchés par l' Internet des Objets:

1. la domotique (automatisation et sécurité de la maison).
2. l'automobile (autonomie, maintenance et assurance).
3. la ville (santé publique et transports).
4. l'externe (transports, logistique et navigation).
5. l'humain (sport et santé).
6. la construction (optimisation des travaux, sécurité et santé).
7. le commerce de grande consommation (automatisation, marketing).
8. les usines (gestion des opérations et des équipements) .

9. le lieu de travail (sécurité et énergie) .

Désormais, nous pouvons interagir avec des objets réels. Pour la première fois, nous pouvons vivre dans les villes intelligentes pleines de capteurs qui nous aident à améliorer notre mode de vie et des machines qui parlent à d'autres machines. Il existe de nombreux domaines d'application inhérents à l' Internet des Objets. Nous nous contentons de décrire seulement quelques domaines d'applications de l'Internet des Objets [9] :

I.3.1. Ville intelligente (Smart city)

Figure I.3: La ville intelligente Songdo, en Corée du Sud[12].

La ville intelligente (ou Smart city) cherche à concilier les piliers sociaux, culturels et environnementaux à travers une approche systémique qui allie gouvernance participative et gestion éclairée des ressources naturelles afin de faire face aux besoins des institutions, des entreprises et des citoyens [10]. Songdo, en Corée du Sud (voir la Figure I.3), est présentée comme le parfait exemple : une ville construite de toute pièce où tous les bâtiments se trouvent dans un rayon de 6 km². Des capteurs et des ordinateurs sont également placés le long des routes et des édifices pour évaluer et ajuster la consommation d'énergie. Songdo a coûté 35 milliards de dollars (25 milliards d'euros environ) et forme le plus grand projet immobilier privé du monde. Elle devrait être achevée en 2017. Parmi les autres villes sur la liste, on compte Masdar aux Émirats Arabes Unis [11], où 500 foyers sont alimentés en énergie grâce à des panneaux solaires et aux sources renouvelables et où les voitures sont interdites. A la place, les habitants se déplaceront à vélo, à pieds ou emprunteront les transports en commun. La ville sera aussi le quartier général de l'Agence internationale de l'énergie renouvelable. Vienne s'est également fixé des objectifs pour devenir une ville neutre en carbone d'ici 2020 [2], [12].Il existe beaucoup d'autres Smart-X Applications comme le smart grid, smart environment, smart agriculture...etc.

I.3.2. Réseau électrique intelligent (Smart grid)

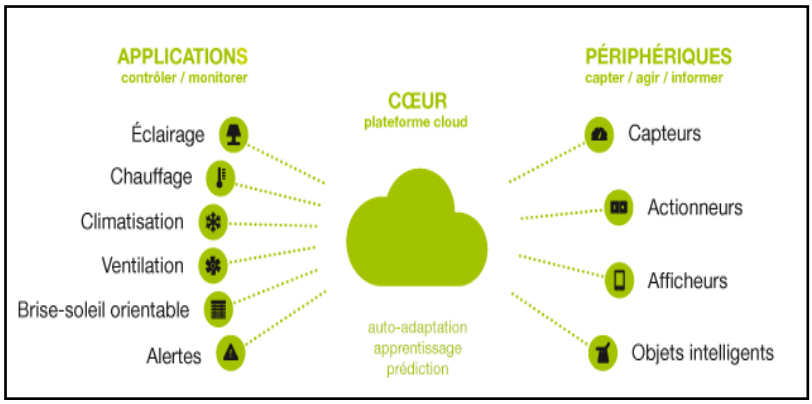


Figure I.4: L'écosystème Hemis [13]

Le smart grid ou « réseau de distribution et de gestion d'énergie intelligent » est un réseau électrique communicant qui intègre les NTIC (Nouvelles Technologies de l'Information et de la Communication) dans son fonctionnement. Cela permet d'établir des interactions entre les réseaux d'électricité et les bâtiments auxquels ils sont raccordés afin d'améliorer la gestion de l'énergie et de rationnaliser la consommation. A titre d'exemple Ubiant la start-up Française qui équilibre le confort et l'énergie dans la maison, a développé Hemis (voir la Figure I.4)qui permet de réduire la consommation d'énergie des bâtiments tout en maximisant le bien-être de leurs occupants. Hemis est un écosystème qui comprend son environnement en analysant en temps réel un grand nombre d'informations collectées : température, humidité, luminosité, CO2, présence humaine,…, il équilibre entre l'efficacité énergétique et le confort des personnes. Hemis est une solution adaptée à la gestion de l'électricité, du gaz, de l'eau et des énergies renouvelables [2] [13].

I.3.3. Textile connecté (Smart sensing)

Figure I.5: le textile connecté chez Cityzen [14]

Un autre domaine d'application est celui du textile connecté où un textile est doté de micro-capteurs intégrés, capables d'effectuer le monitoring d'individus : température, fréquence cardiaque, vitesse et accélération, géolocalisation [2]. Par exemple l'entreprise française CityzenSciences (voir sa publicité la Figure I.3) qui est spécialisée dans la conception, la création, le développement de textiles « connectés », est le pilote d'un projet industriel textile, Smart Sensing, menée par un consortium d'entreprises, fortement soutenu

par la BPI[5] France. Son but : créer, concevoir et développer l'industrie française du vêtement connecté. Le sport professionnel et amateur est la cible première du projet Smart Sensing [14].

I.4. La Problématique : les données générées par l'Internet des Objets.

L'Internet des Objets contribuera « à doubler la taille de l'univers numérique tous les deux ans, lequel devrait peser 44.000 milliards de giga-octets en 2020, soit 10 fois plus qu'en 2013» [2], [3]. Et pour cela l'évolution de l'Internet des objets et les données massives, et la plupart du temps non structurées qu'elle génère, dévoile une problématique de gestion de ses données.

Les données sont la nouvelle ressource brute à exploiter qui est produite en permanence et utilisée dans les processus de décision. Dans l'Internet des objets, le rapprochement entre le monde physique et le monde numérique passe par la capacité du monde numérique à observer le monde physique, cela grâce par exemple à un déploiement massif de capteurs connectés aux infrastructures.

Notre objectif : Afin que le monde numérique comprenne ce qui se passe dans le monde réel, il faut tout d'abord transformer ces données capturées en des informations, de la connaissance et de la cognition, c'est-à-dire un apprentissage qui permettra au monde numérique de prendre des décisions de plus en plus complexes et de manière autonome [5].

Néanmoins, les approches informatiques classiques peinent à prendre en compte de manière satisfaisante les besoins de la gestion des données de l' Internet des Objets. Plus précisément : Le temps de latence qui dépend de rotation du disque, le temps de recherche, et le temps de transfert entre le disque dur et les processeurs deviennent critiques dans un environnement étendu comme celui de l'Internet des Objets.

De plus les SGBD (Systèmes de Gestion de Bases de Données) traditionnels sont conçus pour fonctionner en mode transactionnel, exécutant rapidement des requêtes complexes portant sur un volume raisonnable de données : ils ne sont aucun cas capables de traiter de manière séquentielle des volumes de données se chiffrant au minimum en dizaines ou centaines de To, même des systèmes distribués ont bien été développés pour essayer d'accélérer les temps de traitements, mais sans être vraiment convaincants.

En outre, la limite des techniques actuelles d'exploitation et de traitement de ces données massives, très hétérogènes et la plupart du temps non structurées nous a incité à explorer de nouvelles méthodes d'analyse, de traitement et de stockage des données afin d'offrir des services toujours plus performants.

[5] Banque publique d'investissement

Dans ce projet, nous allons nous pencher sur une nouvelle approche de gestion de données fondée sur le Big data. Cette approche est actuellement explorée par plusieurs équipes autour de projets de l'Internet des Objets. Par ailleurs, il est à noter que l'organisme international de normalisation IEEE [15] (Institute of Electrical and Electronics Engineers) identifie chacun des deux domaines : Internet des objets (les capteurs) et les Big data parmi les cinq technologies qui formeront le monde. Dans ce qui suit nous allons voir comment la technologie Big data pourra faciliter la gestion des données de l'Internet des Objets.

I.5. Conclusion

Tandis que beaucoup se concentre sur les « objets » qui composent l'écosystème de l'Internet des Objets, ce sont les données générées par ceux-ci qui seront la clé de leur succès. Leur bonne utilisation ainsi que la capacité d'analyse des données seront une condition préalable à la prise en charge de cette infrastructure. Dans ce domaine, les entreprises munies d'outils pour utiliser ces informations à leur avantage concurrentiel seront alors les vainqueurs incontestés [36], parce que connecter des objets, c'est relativement "facile" ; la difficulté, c'est d'exploiter les Big data qu'ils produisent, et d'y ajouter de la valeur [40]. Autrement dit, il faut transformer les données brutes en information porteuse de sens pour l'utilisateur.

CHAPITRE II : BIG DATA

II.1. Introduction
II. 2. La mise en donnée du monde
II.3. Aux origines du Big data
II.4. Le Big data : Volume ou Technologie ?
II.5. Caractéristiques des Big data
II.6. L'internet des objets et le Big data
II.7. Quelques exemples de l'intégration de l'Internet des Objets et du Big Data dans des macrostructures
II.8. Problématique de gestion des données massives (Big data)
II.9. Conclusion

II.1. Introduction

Le Big Data est un phénomène qui a vu le jour avec l'émergence de données volumineuses qu'on ne pouvait pas traiter avec des techniques traditionnelles. Les premiers projets de Big Data sont ceux des acteurs de la recherche d'information sur le web « moteurs de recherche » tel que Google et Yahoo. En effet, ces acteurs étaient confrontés aux problèmes de la scalabilité (passage à l'échelle) des systèmes et du temps de réponse aux requêtes utilisateurs. Très rapidement, d'autres sociétés ont suivis le même chemin comme Amazon et Facebook. Le Big Data est devenu une tendance incontournable pour beaucoup d'acteurs industriels du fait de l'apport qu'il offre en qualité de stockage, traitement et d'analyse de données. [18]

II.2. La mise en donnée du monde

Au delà d'un volume gigantesque, c'est la diversité des sources de données qui donne au Big Data toute son ampleur. Deux leviers principaux soutiennent cette croissance de la production de données : l'effacement de la frontière entre comportements online et offline et la mise à disposition des données publiques. On identifie aujourd'hui quatre grands facteurs responsables de l'explosion de la production de données par nos comportements connectés [25].

II.2.1. Les réseaux sociaux

A chaque minute écoulée, on compte sur internet au niveau mondial : 98 000 tweets, 695 000 mises à jour de statuts et onze millions de messages instantanés sur Facebook. Ce dernier s'occupe également de la gestion de 50 milliards de photos.

II.2.2. Les objets connectés

L'Internet des Objets contribuera « à doubler la taille de l'univers numérique tous les deux ans, lequel devrait peser 44.000 milliards de giga-octets en 2020, soit 10 fois plus qu'en 2013» [2], [3].

II.2.3. Les technologies mobiles

On considère qu'un smartphone génère environ 60 gigabytes chaque année. Si on multiplie ce chiffre par le nombre de smartphones dans le monde soit environ un milliard, on obtient une production de données par an de 56 exabytes soit la totalité de la bande passante consommée
en 2013, dans le monde. Le terme Big Data prend alors tout son sens. En 2018, les prévisions estiment qu'il y aura 3,3 milliards de smartphones dans le monde.

II.2.4. Les comportements numériques scrutés, analysés et stockés

A chaque minute écoulée, on compte sur Internet 700 000 recherches Google, 12 000 annonces sur Craigslist, 600 nouvelles vidéos Youtube et 1 500 articles de blogues...etc.

II.3. Aux origines du Big data

C'est en 1944 que le problème de stockage et d'accès aux données a été mentionné pour la première fois par un bibliothécaire de l'Université Wesleyan. Il avait estimé que la bibliothèque de Yale comporterait en 2040 environ 200 millions d'ouvrages, ce qui représenterait 10 000 kilomètres de rayons. Le terme « Big data » aurait été employé pour la première fois en 1997 par des chercheurs américains à la NASA (National Aeronautics and Space Administration). Ils affirmaient alors que l'augmentation du volume des données devenait problématique pour les systèmes informatiques de l'époque. C'est ce qu'ils ont appelé le « problème des Big data ». À la fin des années 1990, la puissance du matériel informatique s'est considérablement développée. Les sources de données se sont multipliées : l'Internet, les réseaux sociaux, la téléphonie mobile. A la fin des années 2000, l'avènement d'outils comme le cloud computing a permis de stocker des données à moindre coût. Depuis, l'espace sur le cloud a fortement augmenté et a contribué à l'essor du phénomène « Big data » [16].

L'avènement du Big data constitue comme jamais auparavant un défi pour notre mode de vie et modifie notre relation avec le monde : « il ne s'agit plus de connaître le pourquoi, mais seulement le quoi. La révolution ne réside pas dans les calculs effectués par les machines mais dans les données elles-mêmes et la façon de nous en servir » [3].

II.4. Le contexte du Big data : Volume ou Technologie ?

Le terme anglo-saxon « Big data » n'a pas d'équivalent en français. On parle parfois de « données massives », ou de « données de grande dimension ». Mais on parle également de concept, de phénomène, ou encore de discipline Big data. Ce terme désigne le traitement automatisé de grandes quantités de données pour en extraire des informations. Pour cela, on recourt à des nouvelles procédures de transfert, de stockage et d'analyse. Le terme « Big data » est aussi devenu synonyme de « data analysis » (analyse des données). Le Big data correspond donc à la fois à une masse considérable de données, mais aussi aux technologies, processus et techniques mise en œuvre pour gérer des données à grande échelle dans le but d'en extraire des connaissances [16], [17].

On peut parler de « Big Data » dés lors que :

- ✓ Les volumes à traiter atteignent des tailles « plus grandes » que les problèmes courants : Peta (web), Terra, Exa, Zettaoctets, …
- ✓ Le problème ne peut pas être traité par les outils existants : SGBD relationnels, moteurs de recherche, … [18].

II.5. Caractéristiques des Big data

Les principales caractéristiques concernant les Big data retrouvées dans les articles sont leur taille importante et leur complexité. Les Big data ne concernent pas seulement l'ampleur et l'étendue des nouveaux jeux de données mais aussi leur complexité croissante. Pour décrire la complexité des Big data, une approche largement utilisée est celle des trois « V » [3], [15], [16], [17], [18], [38]: Le volume, la variété et la vélocité (voir la Figure II.1). « Les Big data

sont un terme utilisé pour décrire les données dont le traitement est problématique du fait de leur taille (volume), de la fréquence de leur mise à jour (vélocité), ou de leur diversité (variété) ». La véracité est un quatrième « V » parfois ajouté pour décrire un challenge posé par les Big data. Certains auteurs mentionnent même un cinquième « V » : la valorisation [16].

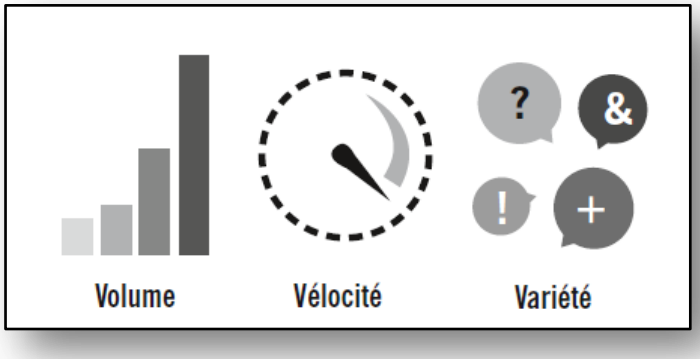


Figure II.1: La formule des 3V du Big data [3]

II.5.1. Volume

Le volume est la principale caractéristique qui se traduit aujourd'hui en des téraoctets (10^{12} octet), des pétaoctets (10^{15} octet), des exaoctets (10^{18} octet), des zettaoctets (10^{21} octet), ou des yottaoctets (10^{24} octet)…etc. par exemple EMC [15] l'entreprise américaine des logiciels et des systèmes de stockage a reconnu une présentation PowerPoint de 40 mégaoctet en tant que Big data (1Mo= 10^6 octet), parce que 40Mo est très grande par rapport à la taille typique d'une présentation PowerPoint. Par ailleurs, une animation de 1 pétaoctet (1Po=10^{15} octet) et une image médicale de 1téraoctet (1To=10^{12} octet) sont considérées comme Big data comme ils sont de grande taille par rapport à la taille typique de chacune. Maintenant on ne va pas mentionner un seuil pour les Big data parce que la taille est un terme relatif quand il s'agit de données et on ne sait pas comment ça va être le « Big » de Big data. Enfin, les données qu'on les considère comme Big data aujourd'hui, peuvent ne pas être considérée comme Big data demain en raison des progrès dans le traitement, le stockage et autres fonctionnalités du système [15].

II.5.2. Variété

La variété se traduit en l'agrégation des données provenant des sources très diverses ou le regroupement de données provenant de sources indépendantes. On désigne l'origine variée des sources de données qui sont soit structurées ou non structurées (images, mails, tweets, données de géolocalisation,…) [17].

II.5.3. Vélocité

L'augmentation rapide des données est une autre caractéristique des Big data. Il s'agit de données en temps réel ou en temps quasi-réel, La vélocité correspond à la vitesse à laquelle les données d'aujourd'hui sont générées et traitées simultanément. L'augmentation du volume, de la vitesse, et la diversité de données, posent des nouveaux défis pour les processus de prise de décisions [17].

II.5.4. Valorisation

La valorisation signifie que quelque part à l'intérieur de ces données, il y a quelques précieuses informations, quelques données d'or à extraire, si la plupart des morceaux de données, individuellement, peuvent sembler sans valeur [15].

II.5.5. Véracité

La véracité recouvre la précision et l'exactitude des données, les Big data doivent être interprétées avec précaution pour être utiles, Ils peuvent être difficile à valider. Les Big data ont une faible véracité et ne peuvent jamais être exactes à 100 % [17].

II.6. L'internet des objets et le Big data

II.6.1. Des évolutions technologiques favorisent la nouvelle ère de l'Internet des objets

Trois évolutions technologiques accompagnent le développement de l'Internet des objets et du Big Data et permettront de transformer la donnée en information utile pour prendre les bonnes décisions, au bon moment et en toute sécurité :

- ✓ L'essor des Smartphones a engendré une baisse du coût des capteurs et favorisé l'essor de tout un ensemble d'objets connectés qui intègrent des myriades de capteurs produisant des données.
- ✓ L'amélioration des performances des réseaux en termes de débit et/ou de consommation d'énergie pour véhiculer ces données.
- ✓ L'amélioration des algorithmes dans le traitement des données et pour des volumes fortement croissants [39].

II.6.2. Pourquoi l'Internet des Objets est il -il intimement lié au Big data ?

"Quand on parle Big Data, on va tout de suite parler volume de données. Mais au delà du volume, rien que la variété de ces dernières va constituer un enjeu crucial, ce phénomène est amplifié par l'avènement des objets connectés." dit Tania Aydenian directrice de datavenue (Orange Technocentre) [38].

L'Internet de Objets est devenu un réel phénomène en 2014 et 2015. Si les technologies sont apparues depuis quelques années, le sujet s'est trouvé au coeur des discussions tout au long de l'année. Et pour cause. Selon une étude EMC-IDC, l'internet de objets contribuera « à doubler la taille de l'univers numérique tous les deux ans, lequel devrait peser 44.000 milliards de gigaoctets en 2020, soit 10 fois plus qu'en 2013». On estime qu'actuellement seulement 22% des données sont exploitables pour le Big Data, chiffre qui sera porté à 35% grâce aux données numériques issues de l'internet des objets [2].

Le Big data et les objets connectés représentent un important relais de croissance économique selon de nombreuses études tels que Cisco, McKinsey, Idate, Inspection générale des finances, Gartner, Boston Consulting Group, A.T. Kearney. Ils ouvrent la possibilité de connecter les personnes ou les objets de manière plus pertinente, de fournir la bonne

information au bon destinataire et au bon moment, ou encore de faire ressortir les informations utiles à la prise de décision [3].

On vive aujourd'hui une révolution numérique globale, une nouvelle révolution industrielle, alimentée par l'essor des objets connectés associé à l'exploitation du Big data, qui est appelée parfois Internet of Everything (IoE), l'Internet du Tout connecté. Selon Yannick Lacoste et Jean-François Vermont, « l'offre d'objets connectés est très en avance sur les usages. Le flot grandissant d'objets connectés soutient la croissance du Big data qui, à son tour, facilite l'explosion des usages [3].

II.7. Quelques exemples de l'intégration de l'Internet des Objets et du Big Data dans des macrostructures

La société multinationale américaine IBM «International Business Machines Corporation » a attribué la quantité croissante de Big data vers un monde instrumenté, interconnecté et intelligent qui est envisagé par l'Internet des Objets [15]. Et aujourd'hui beaucoup de conversations qui ont lieu autour de l'Internet des Objets sont incomplète sans une mention de Big data. « Le succès ou l'échec de l'Internet des Objets repose sur le Big data » dit Brian Hopkins, analyste chez Forrester [1]. Comme les organisations entrent dans l'Internet des Objets, elles doivent comprendre la relation symbiotique entre elle et le Big data. Voici quelques exemples de l'Internet des Objets et Big data fonctionnent bien ensemble pour fournir l'analyse et la perspicacité.

II.7.1. La poste

Figure II.2: Les véhicules intelligentes de UPS [19]

United Parcel Service "UPS" est une entreprise postale qui utilise des capteurs sur ses véhicules de livraison (voir la Figure II.2.) pour surveiller la vitesse, le kilométrage, le nombre d'arrêts, et le moteur…pour le but d'économiser de l'argent, d'améliorer l'efficacité et de réduire son impact environnemental. Selon une infographie publiée par UPS, les capteurs enregistrent plus de 200 points de données pour chaque véhicule dans une flotte de plus de 80 000 chaque jour [19]. Ceux-ci aident l'entreprise à réduire le temps de ralenti, la consommation de carburant, et les émissions nocives. Autrement UPS utilise le Big data dans

son projet ORION qui signifie On-Road Integrated Optimization and Navigation (optimisation et navigation intégrées sur la route). ORION connaît le chemin et les adresses de livraison des clients, les lieux et les heures de livraison et de transport ainsi que les règles syndicales pour les chauffeurs. Il garde également en mémoire 250 millions d'adresses de livraison. ORION analyse toutes ces données et prépare des instructions de transmission optimisées jusqu'à la minute même du départ du chauffeur [20].

II.7.2. Le tourisme

Figure II.3: Le bracelet connecté MagicBand [21]

Le plus bel exemple de cette interaction est au parc d'attractions 'Disneyland Resort aux États-Unis'. La société a investi plus d'un milliard de dollars pour déployer son bracelet connecté connu sous le nom de MagicBand (voir la Figure II.3). Un bracelet qui emporte l'influence des visiteurs puisque 90% d'entre eux se sont déclarés satisfaits des bénéfices du port de ce bracelet, suite à une enquête. Le bracelet permet aux parents de géolocaliser leurs enfants, de payer dans les boutiques et restaurants ou encore de récupérer les photos prises dans les attractions. La maison Disney recueille ces données et les emploie en vue d'améliorer l'expérience des visiteurs [21], [1].

II.7.3. Le sport

Figure II.4: l'application Nike + iPod / iPhone [15].

Big data est partout, même dans le jogging, Un exemple d'application de technologies de détection dans notre vie quotidienne est l'application Nike + iPod / iPhone (voir la Figure II.4.) C'est une application qui recueille et suit les informations telles que les détails d'entraînement, la distance, les calories brûlées, etc. d'un jogger en utilisant des chaussures Nike + iPhone / iPod. Une autre application similaire est iSmoothRun (www.ismoothrun.com). En outre, il permet le téléchargement des données sur les réseaux

sociaux de fitness tels que (www.RunKeeper.com). Les données deviennent «Big data» lorsque l'on considère des millions d'utilisateurs [15].

II.8. Problématique de gestion des données massives (Big data)

Les Big data soulèvent des défis à toutes les étapes de la gestion des données : le stockage, le traitement, l'analyse...etc. Les Big data ne sont pas simplement de grands volumes de données, elles se déplacent rapidement, sont difficiles à valider et à valoriser [16]. Le stockage du Big data est une chose, son traitement est une autre. Maintenant Il faut alors s'adapter et tenter de nouvelles méthodes de traitement. La question n'est donc plus d'identifier quelles données stocker, mais, qu'est-ce qu'on peut faire avec ces données ? Cette masse de données qui arrive en flot continu, provenant des sources très diverses, son traitement posent des problèmes en particulier dans l'extraction de connaissances [17].

II.8.1. La sémantique dans le Big data

Le Big data se réfère ainsi à ce qui peut être accompli à grande échelle et ne peut pas l'être à une échelle plus petite. Le Big data s'appuie sur le développement d'applications à visée analytique, qui traitent les données pour en extraire de la sémantique. Une chaîne de transformations bien connue existe dans le domaine du management de l'information, c'est la chaîne [22] : « donnée → information → connaissance → sagesse » que représente le modèle DIKW (Data, Information, Knowledge, Wisdom). La représentation graphique la plus populaire pour DIKW est une pyramide, avec les données à la base et la sagesse à son sommet (Figure II.5). Cette représentation suppose implicitement que les éléments les plus hauts dans la pyramide nécessitent les éléments inférieurs pour être définis, et qu'ils peuvent être atteints après un processus de transformation des éléments inférieurs. Le modèle DIKW est alors une chaîne où l'information est le résultat du traitement des données, la connaissance est le résultat du traitement de l'information, et la sagesse est le résultat du traitement de la connaissance.

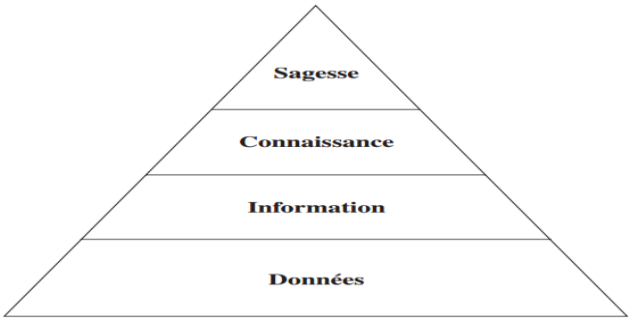


Figure II.5: La pyramide DIKW [35]

II.8.2. La révolution de Big data par la technique

Les progrès techniques et la baisse des prix associée dans la gestion de la donnée sont les premiers facteurs d'émergence du Big Data. Ces progrès concernent à la fois les logiciels de traitement de données et l'architecture informatique nécessaire à son transit et à son stockage. Les données massives existent déjà depuis très longtemps car nous avons toujours stocké les

données. Ce qui fait un projet «Big Data», c'est la technologie que l'on utilise. Avec ces technologies, ce qui change, c'est la puissance et la rapidité de gestion de ces données **[25]**

II.8.2.1. Le Big Data et le Cloud

Jusqu'à l'émergence du concept de "big data", les données étaient principalement traitées de façon locale, dans des entrepôts de données (ou data warehouse) constituées de plusieurs bases de données structurées. Peu à peu les sources de données se sont largement diversifiées, sont devenues relativement hétérogènes (format des données très variable) et ont été surtout localisées sur Internet. Autre particularité, ces informations sont produites en permanence, avec une cadence soutenue [31] [18].

Pour pouvoir exploiter ces mines d'informations et ces flux de données, d'importantes capacités de calcul sont nécessaires, souvent uniquement disponibles dans de grands data centers. Le cloud computing permet donc de "louer" une puissance de calcul et un espace de stockage adaptés pour un traitement big data. En effet, seuls peu d'acteurs sont en mesures d'effecteur ce traitement avec leurs propres infrastructures, au vu des équipements informatiques nécessaires. Le cloud va donc mettre le big data à la portée des PME et des acteurs non experts du traitement des données.

II.8.2.2. Big data et Hadoop

La vague du Big Data n'a été rendue possible que par une démocratisation des outils rendant de plus en plus accessible le traitement massif de données. Dans cette jungle toujours plus touffue de solutions logicielles et de langages de programmation, il n'est pas toujours évident de s'y retrouver. Sont regroupés ici les plus populaires, de l'incontournable Hadoop [38] .

Figure II.6: Doug cutting le créateur de Hadoop [37]

Hadoop a été créé en 2004 et le nom "Hadoop" était initialement celui d'un éléphant en peluche, jouet favori du fils de Doug Cutting l'inventeur de Hadoop (qui apparaît dans la Figure II.6**)** [37], qui voulait agrandir la taille de l'index de son moteur Open Source Nutch. Le terme ne désigne pas un logiciel particulier mais un environnement technologique dont le but est de réaliser des traitements sur des volumes massifs de données. Son fonctionnement se base sur le principe des grilles de calcul : répartir l'exécution d'un traitement sur des grappes

de serveurs c'est-à-dire plusieurs ordinateurs indépendants. La grande innovation de Hadoop réside dans cette distribution de l'information. Les architectures plus traditionnelles adossent le traitement de données à une grappe unique. L'étude de l'institut IDC16 souligne que l'écrasante majorité (98 %) des entreprises portant des projets Big Data ont recourt à Hadoop. Néanmoins, le prix pour la migration de ses bases de données sur Hadoop reste un frein : 45 % des entreprises interrogées ont dû dépenser entre 100.000 $ et 500.000 $ et 30 % d'entre elles, plus de 500.000 $. Troquer une architecture basée sur un entrepôt de données pour un projet Hadoop représente donc un coût elevé. Néanmoins, cette dernière technologie est en moyenne cinq fois moins chère qu'un datawarehouse classique. Ce chiffre comprenant le matériel, le logiciel et le déploiement de l'infrastructure. Sans compter qu'une plateforme Big Data stocke environ cinq fois plus d'informations qu'un datawarehouse traditionnel [25].

II.9. Conclusion

L'offre d'objets connectés est très en avance sur les usages. Le flot de données grandissant d'objets connectés soutient la croissance du Big Data qui, à son tour, facilite l'explosion des usages [25]. En prenant conscience de l'importance grandissante qu'allaient être amenées à jouer les Big Data, les entreprises se sont retrouvées confrontées à une foule de grandes notions, aux contours flous, dont il s'agit désormais de tirer parti. Algorithmes, Smart Data, temps réel, objets connectés… La maîtrise de ces nouveaux domaines riches en promesses passe d'abord par la compréhension de ce que les Big Data impliquent d'un point de vue business [38].

A ce stade on peut dire que le Big Data est un écosystème large et complexe. Il nécessite la maitrise des technologies matérielles et logicielles diverses (stockage, parallélisation des traitements, virtualisation, …). Le Big Data demande de la compétence et de l'expertise dans la maitrise et l'analyse des données [18].

CHAPITRE III : Installation et prise en main de Hadoop

III.1. Introduction
III.2. Une vue global de Hadoop
III.3. Un cluster Hadoop
III.4. Hadoop et ses distributions
III.5. Installation de Hadoop à l'aide de la distribution Cloudera
III.6. Prise en main de Hadoop à l'aide d'un exemple : WordCount
III.7. Conclusion

CHAPITRE III : Installation et prise en main de Hadoop

III.1. Introduction

L'émergence du phénomène Big Data est intrinsèquement liée au fait que l'information et la capacité à la traiter sont devenues l'un des facteurs clé dans le succès d'une entreprise. Soumise à de tels enjeux de volumétrie et d'hétérogénéité, les technologies utilisées jusqu'alors n'ont pas tardé à montrer leurs limites et il a été nécessaire de réinventer un certain nombre d'outils pour qu'ils puissent s'adapter à ces nouvelles contraintes: stockage et traitement de données qui doit désormais être distribué, collecte de données hétérogènes et multi-sources, restitution des données. Il est intéressant de noter que l'essentiel des briques fondamentales de la mouvance Big Data sont open source et structuré autour de Hadoop. Plus encore : certains acteurs historiquement hostiles à l'open source comme Microsoft sont aujourd'hui en train d'abandonner leurs solutions propriétaires pour se rallier derrière la bannière Hadoop. [34]

Dans ce chapitre, nous allons discuter du fonctionnement de Hadoop car c'est à l'aide de cette plateforme que nous développerons notre méthodologie de traitement des données volumineuses.

III.2. Une vue global de Hadoop

Hadoop est un système matériel et logiciel distribué capable de répondre aux besoins du Big Data, tant au plan technique qu'économique. Hadoop est capable de stocker et de traiter de manière séquentielle, et à des coûts "raisonnables", des volumes de données de plusieurs péta-octets. Il offre une grande flexibilité (possibilité d'enlever ou d'ajouter des machines à chaud) et ses performances évoluent de manière quasi linéaire en fonction du nombre de machines constituant le cluster **[27].** Hadoop a été écrit en Java, qui demeure le langage de prédilection pour développer des programmes Hadoop "natifs". Il est cependant possible, dans certaines limites, d'écrire des jobs Hadoop en Perl, Python, Ruby…

Le cœur de Hadoop comprend deux composants :

- Le système de gestion de fichiers distribué**, HDFS**.
- Le framework logiciel **MapReduce.**

Outre ces deux composants, l'écosystème de Hadoop, comme la Figure III.1 nous le montre, comprend de nombreux autres outils tels que Pig, Hive, Oozie, Sqoop, etc. que nous allons détailler ci-après.

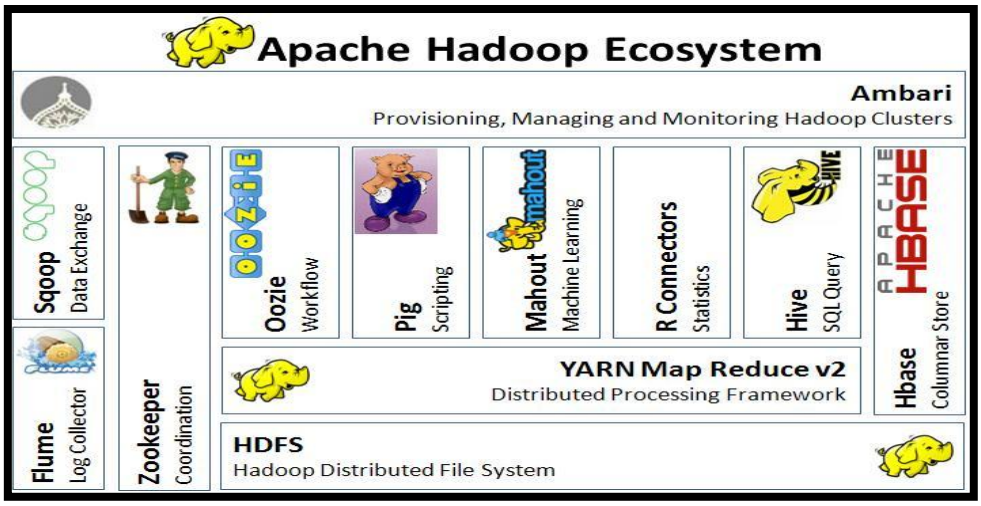


Figure III.1: L'écosystème de Hadoop

III.2.1. Le centre névralgique HDFS et MapReduce

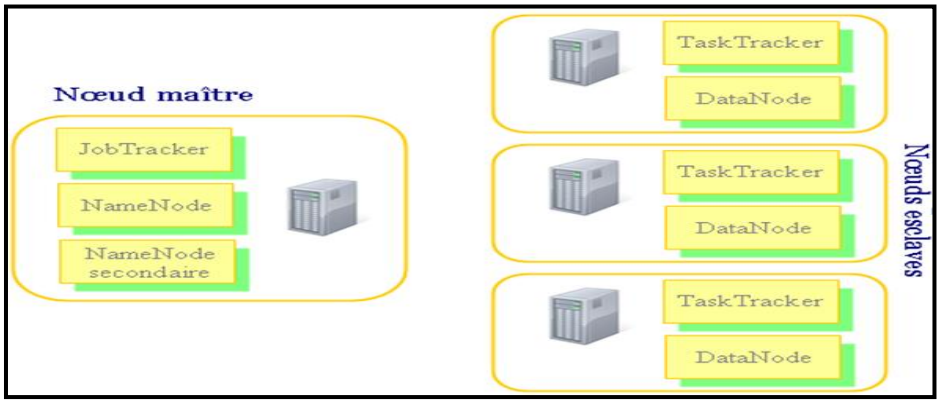


Figure III.2: Les daemons de HDFS [44].

Nous avons dit que le cœur d'Hadoop est composé de deux modules majeurs: HDFS et MapReduce. Inspiré de Google File System et développé en Java, **HDFS (Hadoop Distributed File System)** s'exécute au-dessus du système de gestion de fichiers de chaque noeud d'un cluster. Un noeud où sont stockées les données (et exécutés les traitements qui les concernent) s'appelle un **datanode**. HDFS stocke l'information sous forme de blocs Hadoop (64 Mo par défaut, 128 recommandés). En tant que fichier du serveur, un bloc Hadoop occupe physiquement plusieurs blocs. Si un fichier ou une partie de fichier est plus petit qu'un bloc Hadoop, sa taille s'ajuste. Chaque fichier est découpé en blocs Hadoop répartis sur des **datanodes** différents, et une ou plusieurs copies de chaque bloc sont enregistrées sur des **datanodes** différents. Une capacité qui assure une **tolérance aux pannes** appréciable. Un serveur appelé **Namenode** gère toutes les métadonnées des blocs Hadoop et sait donc les

retrouver. Certains éditeurs traditionnels ou pionniers Big Data remplacent HDFS par leur propre système de gestion de fichiers distribué (voir Figure III.2**)** [37] [38].

MapReduce assume plusieurs rôles. Il gère et alloue aux applications les ressources du cluster, et exécute les traitements appliqués aux données. Lorsqu'une requête est adressée à Hadoop, elle est prise en main par un **JobTracker** qui coordonne les traitements entre Map et Reduce et assure le suivi des tâches. Il distribue les processus parallélisés aux **Task Trackers** sur les noeuds du cluster Hadoop en optimisant les échanges. La fonction **Map** divise la demande initiale en séquences (clé, valeur) auxquelles vont être appliqués le ou les traitements en parallèle (vitesse optimale). Chaque tâche Map renvoie un résultat (clé-valeur). Puis un traitement (Shuffle & sort) remanie les résultats pour regrouper ceux qui ont la même clé. La fonction **Reduce** prend ces résultats et les "réduit" en effectuant une opération sur les valeurs associées à chaque clé (montant total, nombre d'occurrences, etc...) [37] [38]. (Nous allons après donner un exemple de MapReduce (§ III.5.1.).

III.2.2. Les autres outils de Hadoop

III.2.2.1. Hive : Requêtage des données

Hive est à l'origine un projet Facebook qui permet de faire le lien entre le monde SQL et Hadoop. Il permet l'exécution de requêtes SQL sur un cluster Hadoop en vue d'analyser et d'agréger les données. Le langage SQL est nommé HiveQL. C'est un langage de visualisation uniquement, c'est pourquoi seules les instructions de type "Select" sont supportées pour la manipulation des données. Dans certains cas, les développeurs doivent faire le mapping entre les structures de données [47] . Le plus gros avantage de Hive est sa capacité à utiliser une compétence très répandue qu'est la connaissance de SQL rendant les développeurs très rapidement opérationnel pour extraire les données [34].

III.2.2.2. Pig : Scripting sur les données

Pig est à l'origine un projet Yahoo qui permet le requêtage des données Hadoop à partir d'un langage de script [47]. Pig est un outil de traitement de gros volumes de données qui permet l'écriture de scripts qui sont éxécutés sur l'infrastructure Hadoop sans être obligé de passer par l'écriture de tâche en Java via le framework MapReduce [34], il fournit les opérations de filtrage, jointure et classement des données (conçu spécialement pour l'analyse de données) , et c'est cet outil là que nous avons choisi (§ chapitre IV) [18].

III.2.2.3. Sqoop : Intégration SGBD-R

Sqoop est un projet de la fondation Apache qui a pour objectif de permettre une meilleure cohabitation des systèmes traditionnels de type SGBDRs avec la plateforme Hadoop. Il est ainsi possible d'exporter des données depuis la base de données et de procéder aux traitements coûteux en exploitant le cluster Hadoop [47] . Les dispositifs de collecte basés sur une base de données sont à ce jour les plus répandus. Il est ainsi possible de procéder à la collecte de données au sein d'applications traditionnelles n'ayant pas la capacité de se connecter au cluster. Inversement, il est possible d'exporter le résultat d'un traitement vers une base de données tierce afin qu'il soit exploité par une application (à des fins de

restitution par exemple). Sqoop a été conçu avec comme objectif principal d'assurer des performances élevées pour ces opérations d'import ou d'export massifs [34].

III.2.2.4. HBase

HBase est un système distribué de gestion de bases de données NoSQL en colonnes. Projet Apache, il est né suite aux publications de Google sur Big Table en 2006. Installé sur HDFS. Il fonctionne en mode cluster, est horizontalement évolutif et tolérant aux pannes. Le mode colonne réduit les accès à des index et le nombre d'accès disque. Donc performant pour l'analytique [37].

III.2.2.5. Zookeeper

Zookeeper a été conçu sur la base du logiciel Chubby de Google. Il propose une gestion centralisée de configurations pour grands systèmes distribués aussi bien des machines physiques que des services applicatifs Hadoop. Il permet de suivre et de maintenir l'état des services distribués (comme MapReduce ou Hbase) pour les rendre consistants [37].

III.2.2.6. Oozie: Ordonnanceur

Oozie est une solution de workflow (au sens scheduler d'exploitation) utilisée pour gérer et coordonner les tâches de traitement de données à destination de Hadoop. Oozie s'intègre parfaitement avec l'écosystème Hadoop puisqu'il supporte les types de jobs suivants:
MapReduce (Java et Streaming), Pig, Hive, Sqoop, et autres tels que programmes Java ou scripts de type Shell [47] .

III.3. Un cluster Hadoop

Un ensemble de machines fonctionnant avec HDFS et MapReduce s'appelle un cluster Hadoop. Chaque machine s'appelle un nœud. Un cluster peut avoir de un à plusieurs milliers de nœuds. Plus il y a de nœuds, plus les performances du cluster sont bonnes. Hadoop a été conçu pour satisfaire aux objectifs suivants :

- ✓ Un cluster Hadoop doit pouvoir stocker et traiter des volumes de données très importants, dans des délais et à un coût acceptables.

Si un nœud d'un cluster Hadoop tombe en panne :

- ✓ Cela ne doit jamais entraîner de perte de données.
- ✓ Sa charge de travail doit être répartie automatiquement entre les nœuds restants.
- ✓ S'il est en train d'exécuter une tâche pour un job, la panne ne doit pas affecter le bon déroulement du job.
- ✓ Après qu'un nœud défaillant a été réparé, il doit pouvoir réintégrer le cluster sans qu'il soit besoin de redémarrer ce dernier.
- ✓ L'ajout de nœuds dans un cluster doit se traduire par une amélioration proportionnelle de ses performances.

III.3.1. Stocker et traiter des volumes de données très importants (Big data)

Un cluster Hadoop est constitué de plusieurs dizaines, centaines ou milliers de nœuds. C'est l'addition des capacités de stockage et de traitement de chacun de ces nœuds qui permet

d'offrir un espace de stockage et une puissance de calcul pouvant traiter des volumes de données de plusieurs To ou Po. Pour améliorer les performances d'un cluster en lecture/écriture, le système de gestion de fichiers de Hadoop, HDFS, écrit et lit les fichiers par blocs de 64 Mo par défaut (ce paramètre peut être modifié : la valeur recommandée en 2013 est de 128 Mo au moins). Le fait de travailler sur des blocs aussi importants permet de maximiser les taux de transfert des données, en limitant le temps de recherche au niveau des disques durs (seek time).

III.3.2. Garantir la redondance des données

Comme un matériel de milieu de gamme a couramment une durée de vie de trois à cinq ans, la défaillance d'un nœud au sein d'un cluster Hadoop en comportant plusieurs centaines n'est pas un événement rare. Hadoop, ou plus précisément HDFS, intègre une fonction de réplication automatique des données pour limiter les conséquences d'un tel événement. Lors de leur chargement dans un cluster Hadoop, les données sont stockées en trois exemplaires par défaut (ce paramètre, le facteur de réplication, peut être modifié), sur des nœuds différents (voir dans la Figure III.3.). Cette réplication des données répond en fait à deux objectifs :

- En cas de panne d'un nœud, quelle qu'en soit la raison, matérielle ou logicielle, deux copies des données, stockées sur d'autres nœuds, restent disponibles.

- Lors de l'exécution d'un job Hadoop, chaque tâche peut être exécutée sur n'importe quel nœud, surtout s'il stocke une copie des données nécessaires à la tâche. En conséquence, plus il y a de copies des données et plus il y a de nœuds susceptibles d'exécuter la tâche (c'est-à-dire d'être disponibles) dans des conditions optimales à un moment donné.

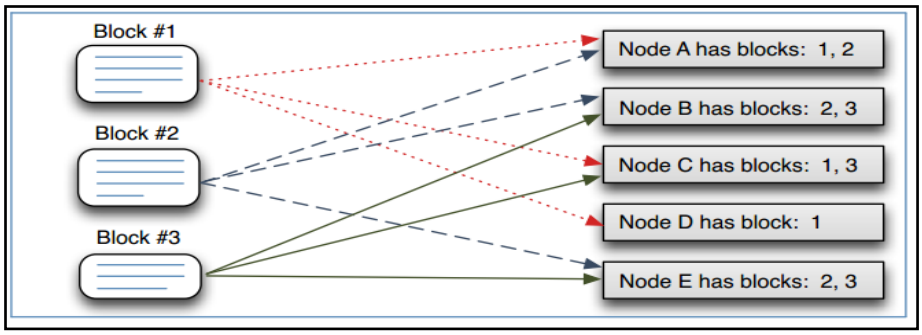


Figure III.3: la réplication des données dans HDFS

La réplication systématique des données, qui est parfois perçue comme un gaspillage d'espace disque, est rendue possible par la baisse du coût de stockage sur disque dur, qui est passé de 150 $ par Go environ en 1997, à 1,05 $ par Go en 2004 et à 0,07 $ par Go en 2009. La réplication des données est un élément capital pour le bon fonctionnement de Hadoop. Elle permet, en outre, dans une certaine mesure, de limiter les besoins en sauvegardes.

III.3.3. Faire face à la panne d'un nœud

III.3.3.1 Réaffecter les taches

Lors de l'exécution d'un job Hadoop, le deamon JobTracker répartit les tâches entre les nœuds de telle sorte que le nœud qui exécute la tâche héberge aussi les données nécessaires à l'exécution de cette tâche. C'est ce que l'on appelle la "Data Locality" (proximité des données) dans le langage Hadoop. Cette approche est le contraire de celle qui prévaut traditionnellement en informatique.

Le choix d'envoyer les programmes (quelques Mo) vers les données (plusieurs To), plutôt que l'inverse, constitue une des grandes originalités de Hadoop. Cela permet de limiter le volume des données circulant dans le cluster et d'économiser à la fois de la bande passante et du temps.

Le JobTracker est capable de détecter la panne d'un nœud et de réassigner automatiquement les tâches concernées à d'autres nœuds. Ce processus se fait de manière transparente pour l'utilisateur.

III.3.3.2 Garantir la bonne fin des jobs en cours

Si une tâche d'un job en cours ne se termine pas normalement, soit parce que le nœud sur lequel elle s'exécute tombe en panne, soit pour une autre raison, Hadoop est capable :
De détecter l'incident.

De déterminer avec précision la tâche concernée (code et données).
De relancer la tâche sur un autre nœud et, si le nœud choisi ne dispose pas des données nécessaires à la bonne exécution de la tâche, d'aller chercher une des deux autres copies des données présentes dans le cluster grâce à la réplication automatique.

III.3.3.3 Le retour à la normale

Hadoop dispose de fonctions natives permettant d'ajouter des nœuds à un cluster Hadoop en fonctionnement, sans arrêter ou relancer celui-ci. Cette opération est généralement menée à bien par l'administrateur Hadoop. L'opération inverse, c'est-à-dire supprimer des nœuds d'un cluster Hadoop en fonctionnement, est possible dans les mêmes conditions

III.4. Hadoop et ses distributions

Dans une distribution Hadoop on va retrouver les éléments suivants (ou leur équivalence) HDFS, MapReduce, ZooKeeper, HBase, Hive, HCatalog, Oozie, Pig, Sqoop, … Ces solutions sont des projets Apache et donc disponibles mais l'intérêt d'un package complet est évident : compatibilité entre les composants, simplicité d'installation, support, …

On évoquera les trois distributions majeures que sont Cloudera, HortonWorks et MapR, toutes les trois se basant sur Apache Hadoop pour choisir une. On peut toutefois les distinguer en fonction de la distance qu'elles prennent avec cette base [27], [32], [47] :
MapR : noyau Hadoop mais repackagé et enrichi de solutions propriétaires.

Cloudera : fidèle en grande partie sauf pour les outils d'administration.
HortonWorks : fidèle à la distribution Apache et donc 100% open source.

Il existe d'autres distributions, voire des offres Cloud, mais qui n'offrent pas l'ensemble des fonctionnalités d'une plate forme Hadoop ou ne sont pas open source (non gratuites) comme Intel Distribution for Hadoop.

Dans ce projet, nous allons utiliser la distribution de Cloudera CDH5 (Cloudera Distribution of Hadoop, version 5). L'employeur de Cloudera c'est Doug Cutting, le créateur d'Hadoop. La société américaine a levé 900 millions de dollars en avril 2014, notamment auprès d'Intel, ce qui lui assure une confortable assise financière. Selon des études, l'offre de Cloudera est probablement la solution Hadoop la plus mature du marché. Elle se compose de deux éditions, l'offre Express et l'offre Entreprise. La première est très limitée. L'éditeur n'assure un support que sur la seconde qui est sa version commerciale [32].

L'atout de Cloudera : son interface unifiée de gestion, le Cloudera Manager. Il s'agit d'un outil propriétaire qui simplifie le déploiement des clusters Hadoop et assure un suivi des performances des nœuds de traitement. En version Entreprise, le Manager assure aussi les backups, reprises sur pannes, etc.

III.4.1. Cloudera Manager

Cloudera Manager est un outil développé par Cloudera pour faciliter l'installation et la gestion d'un cluster Hadoop, quelque soit sa taille. Il se décline en deux versions:
-Express, version gratuite de Cloudera Manager
-Entreprise, version payante de Cloudera Manager, qui dispose de plus de fonctionnalités que Cloudera Express.

III.4.2. Hue (Hadoop User Experience)

Hue est un projet open source qui permet d'interagir avec un cluster CDH et qui comprend principalement [27]:
- Un navigateur de fichiers permettant d'accéder à HDFS
- Un navigateur de jobs MapReduce
- Un navigateur HBase
- Un éditeur de requêtes pour Hive, Pig et Sqoop

III.5. Installation de Hadoop à l'aide de la distribution Cloudera

Il existe 3 modes d'installation de Hadoop c.à.d. que Hadoop peut fonctionner :

1. en mode local (local mode)
2. en mode pseudo-distribué (pseudo-distributed mode)
3. en mode totalement distribué (fully-distributed mode)

1. Le mode local

En mode local, Hadoop fonctionne sur une seule station de travail et les cinq daemons de Hadoop (NameNode, SecondaryNameNode, DataNode, JobTracker et TaskTracker)

s'exécutent tous dans la même JVM (Java Virtual Machine). De ce fait, la portée des variables est très différente de ce qu'elle est dans le mode pseudo-distribué ou dans le mode totalement distribué.

2. Le mode pseudo-distribué

En mode pseudo-distribué, Hadoop fonctionne toujours sur une seule station de travail, mais : Chacun des cinq daemons s'exécute dans sa propre JVM.

Le mode pseudo-distribué est souvent utilisé par les développeurs Hadoop car il permet de développer et tester les programmes dans un environnement simulant assez fidèlement un vrai cluster Hadoop.

3. Le mode totalement distribué

Le mode totalement distribué correspond au fonctionnement d'un vrai cluster Hadoop, avec plusieurs stations de travail interconnectées en réseau. Chacun des daemons de Hadoop s'exécute dans sa propre JVM, souvent sur une machine dédiée

Dans le cadre de ce travail, nous avons opté pour une installation à la fois en mode local et en mode pseudo-distribué.

III.5.1 . Pré-requis matériels et logiciels

III.5.1.1. Pré-requis matériels

Pour pouvoir tester Hadoop soit en mode local ou pseudo-distribué, un micro-ordinateur de milieu de gamme (6 Go à 8 Go de RAM, disque dur de 500 Go ou 750 Go) est suffisant.

à titre indicatif, les exemples de programmes Hadoop présentés dans notre projet ont été mis en œuvre sur un système d'exploitation Linux, disposant 4 Go de RAM et d'un disque dur de 500 Go, mais cette configurations est toute fois très limite en terme de performances, alors on a ajouté 4 Go de RAM et on a réussi d'améliorer les performances.

Laptop:	HP Pavilion 15 Notebook PC
Memory:	8GiB System Memory
Processor:	Intel(R) Core(TM) i3-3217U CPU @ 1.80G
Storage disk:	500GB HGST HTS545050A7

III.5.1.2. Pré-requis logiciels

- **Système d'exploitation**

Hadoop a initialement été développé en Java dans un environnement Linux. Linux demeure le système d'exploitation de prédilection de Hadoop. Il est aussi possible d'installer Hadoop sous Microsoft Windows avec la version 2 de Hadoop.

- **Machines virtuelles**

Enfin, il est également possible d'utiliser une machine virtuelle, sous VMware ou Virtual Box par exemple, pour tester Hadoop. Cette option est celle qui a été retenue dans notre projet, car

elle permet de réduire les risques d'interférence entre Hadoop et le reste du système d'exploitation et est simple et rapide à mettre en œuvre surtout pour le mode local. La solution de virtualisation retenue est Virtual Box dans l'installation en mode local et VMware dans l'installation en mode pseudo-distribué. Le système d'exploitation utilisé est CentOS Linux 64 bits version 6.7 .

III.5.2. Installation de Hadoop en mode local:

Dans un premier temps, nous avons commencé par installer Hadoop en mode local c.à.d. sur une seule machine et nous avons procédé comme suit :

III.5.2.1. Installation de Virtual box

La VirtualBox est un logiciel libre de virtualisation facile à télécharger et installer.
Voici quelques lien de téléchargement :
https://www.virtualbox.org/wiki/Downloads
http://www.01net.com/telecharger/windows/Utilitaire/systeme/fiches/37588.html
http://www.commentcamarche.net/download/telecharger-3673479-virtualbox

III.5.2.2. Installation de Hadoop (Distribution CDH5)

Nous avons choisi d'installer **Cloudera-Quick-Starts** qui est une suite complète d'Apache Hadoop préconfiguré pour une machine virtuelle [42]. Cloudera-Quick-Starts a été téléchargé via le lien suivant **:** www.cloudera.com/content/support/en/downloads.html. En choisissant la version et la plateforme Virtual Box, puis en cliquant sur le bouton **Download Now**. le téléchargement du fichier commence. (4.04 Go).
Nous avons décompressé le fichier téléchargé **Cloudera-quickstart-vm-5.4.2-0-virtualbox** et nous avons le sauvegardé dans un répertoire. En lançant VirtualBox. une nouvelle fenêtre, nommée **Oracle VM VirtualBox - Gestionnaire de machines**, s'affiche. Puis en cliquant sur le bouton **Nouvelle** en haut et à gauche de la fenêtre, nous avons saisi un nom à notre nouvelle machine dans le champ **Nom** (**CDH5** dans notre exemple), et nous avons choisi **Linux** dans la première liste déroulante et **Ubuntu (64 bit)** dans la deuxième (en fait c'est la distribution **CentOS 64 bits 6.7** de Linux qui sera installée). (voir **Figure III.4)**

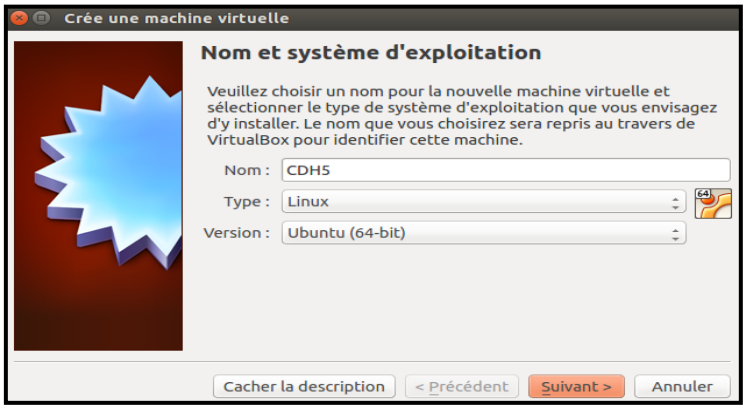


Figure III.4 : le nom et le système d'exploitation de la machine crée .

Nous avons ensuite choisi une taille de mémoire vive de 8 Go, puis en cliquant sur le bouton **Suivant**, une nouvelle fenêtre, intitulée **Disque dur**, s'affiche. En sélectionnant le

bouton **Utiliser un fichier de disque dur virtuel existant**, puis en cliquant sur l'icône de dossier au-dessus du bouton **Créer**, une boîte de dialogue s'affiche et nous avons sélectionné le répertoire que avons sauvegardé précédemment **cloudera-quickstart-vm-5.4.2-0-virtualbox / cloudera-quickstart-vm-5.4.2-0-virtualbox-disk1.vmdk**. (voir Figure III.5)

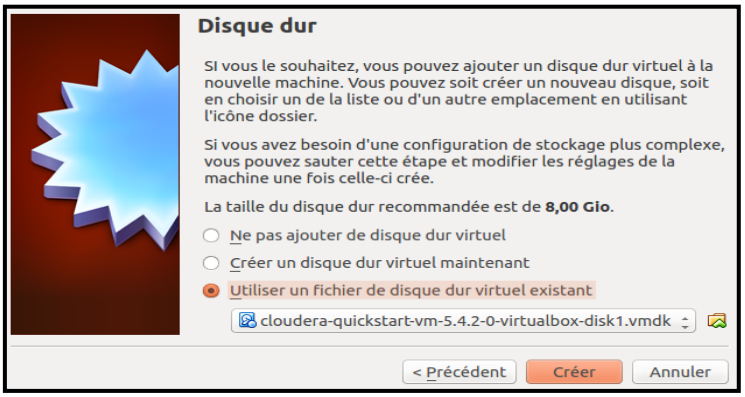


Figure III.5 : Le disque dur virtuel cloudera-quickstart-vm-5.4.2-0-virtualbox-disk1.vmdk

En cliquant ensuite sur le bouton **Ouvrir** puis le bouton **Créer**. La fenêtre principale de VirtualBox s'affichait à nouveau. Une machine virtuelle, intitulée **CDH5**, apparaît à gauche de la fenêtre dans la liste des machines disponibles.
En sélectionnant la machine virtuelle **CDH5** puis en cliquant sur la flèche verte intitulée **Démarrer**, et après 3 à 4 minutes environ, le bureau de CentOS , affiche "Cloudera Ask Bigger Questions". (voir Figure III.6)

Figure III. 6: le bureau de CentOS "Cloudera Ask Bigger Questions".

Nous allons maintenant vérifier que Hadoop est bien installé sur notre station de travail, et que bigdata n'est pas une commande reconnue (**Figure III.7)**

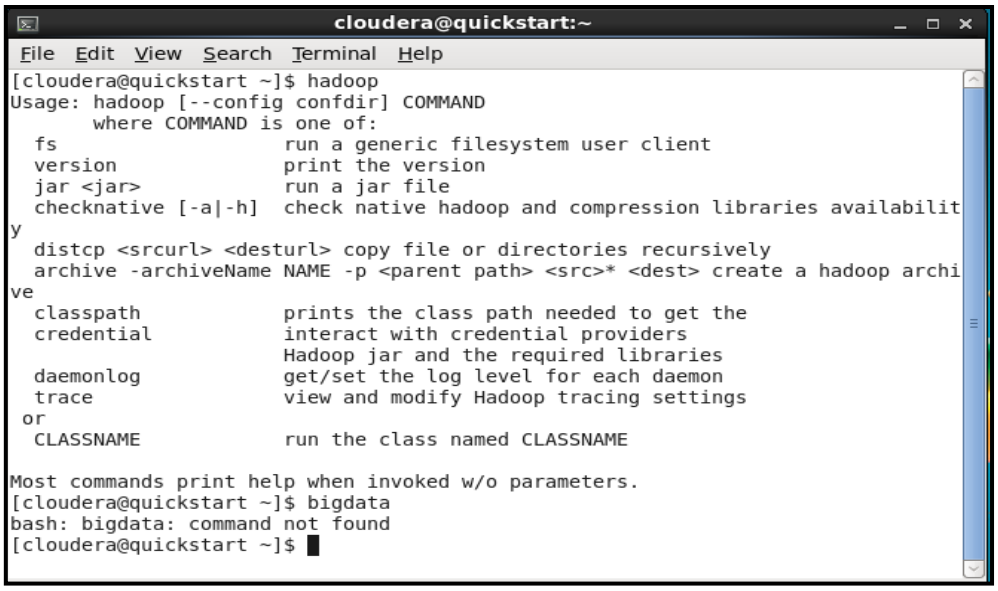


Figure III.7 : Vérification de l'installation de Hadoop

Bref ! Nous venons d'installer Hadoop sur notre station de travail en mode local.

III.5.3. Installer Hadoop en mode pseudo-distribué

Bien entendu, Hadoop n'a d'intérêt que s'il est utilisé dans un cluster composé de plusieurs machines. En effet, utiliser Hadoop dans un mode local (un environnement simple nœud) , comme nous allons le faire, n'a de sens que pour tester la configuration de l'installation ou fournir un environnement de développement MapReduce ou bien quelques applications d'analyse de données.

Nous avons fait quelques recherches et nous avons trouvé des tutoriels et des articles excellents présentant une explication étape par étape sur la façon de configurer un cluster avec une machine virtuelle [42].

III.5.3.1. Sept Actions pour l'installation de cluster CDH5 sur Cent OS 6.7

L'approche globale est simple. Nous créons une machine virtuelle, nous configurons avec les paramètres et les paramètres requis pour agir en tant que nœud de cluster (spécialement les paramètres réseau). Cette machine virtuelle référencée est ensuite clonée autant de fois qu'il y aura des nœuds du cluster Hadoop. Seul un nombre limité de changements sont alors nécessaires pour finaliser le nœud à être opérationnel (uniquement le nom d'hôte et l'adresse IP doivent être définis) [42].

Dans cette installation, je créé un cluster de 2 nœuds. Le premier nœud "le nœud maître ", qui se déroulera la plupart des services de cluster, nécessite plus de mémoire (4 Go) que l'autre nœud "le nœud esclave (2 Go). Dans l'ensemble, nous allons allouer 6GB de mémoire, donc veiller à ce que la machine hôte dispose d'une mémoire suffisante, sinon cela aura un impact négatif sur votre expérience.

CHAPITRE III : Installation et prise en main de Hadoop

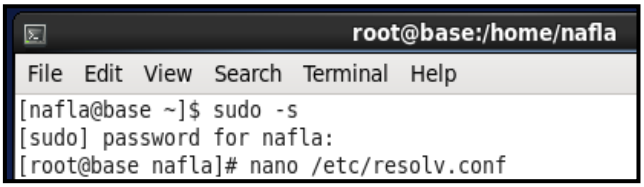
Les conditions préalables pour cette installation est que nous devrions avoir la dernière VirtualBox installé (nous avons la télécharger gratuitement); Nous avons utiliser la distribution Linux CentOS 6.7 (nous avons télécharger l' image de CentOS x86_64bit DVD iso) [42].

Action 1: installer CentOS 6.7 sur VMware Workstation 12 [43]

Action 2: Mettre en place le système de base (pour les nœuds de clonage) [43]

2.1 Connexion en tant que root

2.2 Modifier le fichier /etc/resolv.conf



Ajouter ce qui suit:

```
search example.com
nameserver 192.168.1.1
```

2.3 Modifier le fichier /etc/sysconfig/network

```
[root@base nafla]# nano /etc/sysconfig/network
```

Modifier le contenu pour être:

```
NETWORKING=yes
HOSTNAME=base.example.com
GATEWAY=192.168.235.2
```

2.4 Modifier / etc / selinux / config

```
[root@base nafla]# nano /etc/selinux/config
```

Modifier:

```
SELINUX=disabled
```

2.5 Désactiver le pare - feu , exécuter la commande

```
[root@base nafla]# chkconfig iptables off
```

2.6 Modifier le fichier /etc/yum/pluginconf.d/fastestmirror.conf

```
[root@base nafla]# nano /etc/yum/pluginconf.d/fastestmirror.conf
```

Modifier:

```
enabled=0
```

2.7 Modifier le fichier /etc/sysctl.conf

```
[root@base nafla]# nano /etc/sysctl.conf
```

Ajouter ce qui suit à la fin du fichier:

```
vm.swappiness=0
```

2.8 cd / etc / sysconfig / network-scripts /
cp ./ ifcfg-eth0 ./ ifcfg-eth1

```
[root@base nafla]# cd /etc/sysconfig/network-scripts/
[root@base network-scripts]#  cp ./ifcfg-eth0  ./ifcfg-eth1
```

Modifier le fichier / etc / sysconfig / network-scripts / ifcfg-eth1

```
[root@base network-scripts]# cd /etc/sysconfig/network-scripts/
[root@base network-scripts]# ls
ifcfg-eth0   ifdown-ipv6    ifup          ifup-plip    ifup-wireless
ifcfg-eth1   ifdown-isdn    ifup-aliases  ifup-plusb   init.ipv6-global
ifcfg-lo     ifdown-post    ifup-bnep     ifup-post    net.hotplug
ifdown       ifdown-ppp     ifup-eth      ifup-ppp     network-functions
ifdown-bnep  ifdown-routes  ifup-ippp     ifup-routes  network-functions-ipv6
ifdown-eth   ifdown-sit     ifup-ipv6     ifup-sit
ifdown-ippp  ifdown-tunnel  ifup-isdn     ifup-tunnel
[root@base network-scripts]# nano ifcfg-eth1
```

(En fonction de votre carte réseau, changer le nom du fichier selon le nom qui correspondant , par exemple, vôtre fichier pourrait être:
/ etc / sysconfig / network-scripts / ifcfg-eth0
CentOS 6.7 énumérera la carte réseau comme eth1 par défaut.
Donc nous avons besoin de créer un fichier de script pour eth1 en copiant le fichier de eth0 .)

Modifier contenu:

```
DEVICE="eth1"
BOOTPROTO="dhcp"
HWADDR="00:0C:29:F5:45:19"
IPV6INIT="yes"
NM_CONTROLLED="yes"
ONBOOT="yes"
TYPE="Ethernet"
UUID="d160cc8a-bda2-41e5-8fd5-a009f0d37013"
```

2.9 Redémarrer le réseau, lancer la commande:

```
[root@base network-scripts]# service network restart
```

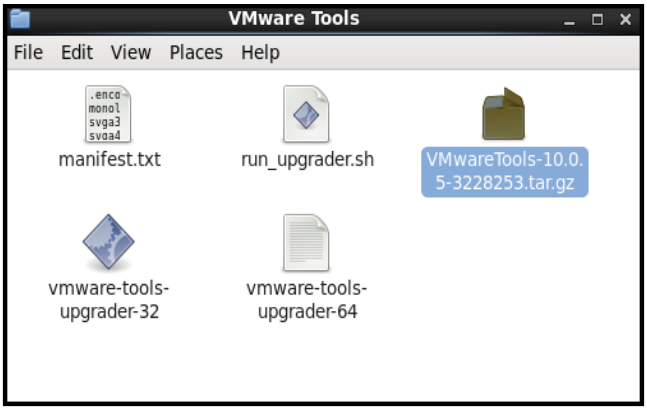
2.10 Installer perl et openssh, exécuter la commande suivante

```
[root@base nafla]# yum -y install perl openssh-clients
```

2.11 Installer VMwareTools:

En cliquer sur VM de la barre de menu de VMware Workstation 12 et sélectionner l'option "VMware Tools Installer"

Nous avons obtenu un fichier /media/VMwareTools-xxx.tar.gz dans le dossier de téléchargement.



exécuter la commande:

tar -xzf /media/VMwareTools-xxx.tar.gz

./vmware-tools-distrib/vmware-install.pl -d

```
[root@base network-scripts]# cd ~
[root@base ~]# ls
anaconda-ks.cfg  install.log.syslog                vmware-tools-distrib
install.log      VMwareTools-10.0.5-3228253.tar.gz
[root@base ~]# tar -xzf *.gz
[root@base ~]# ls
anaconda-ks.cfg  install.log.syslog                vmware-tools-distrib
install.log      VMwareTools-10.0.5-3228253.tar.gz
[root@base ~]# cd vmware*
[root@base vmware-tools-distrib]# ls
bin  doc  FILES    installer  vgauth              vmware-install.real.pl
caf  etc  INSTALL  lib        vmware-install.pl
[root@base vmware-tools-distrib]# ./vmware-install.pl -d
```

2.12 Pour la mise à jour de packages, lancer la commande

```
[root@base ~]# yum update -y
```

2.13 Modifier le fichier / etc / hosts

```
[root@base ~]# nano /etc/hosts
```

et ajouter les lignes suivantes:

192.168.1.13 master.example.com master
192.168.1.10 slave.example.com slave

```
192.168.1.13 master.example.com master
192.168.1.10 slave.example.com slave
```

2.14 Générer les clés privées / publiques:

ssh-keygen
cd .ssh/
cp id_rsa.pub authorized_keys

```
[root@base nafla]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
f5:8e:c8:10:c8:d2:4f:f6:50:3d:6e:79:07:2d:6a:ff root@base.example.com
The key's randomart image is:
+--[ RSA 2048]----+
|         ..   .  |
|   o . .  o o .  |
|  . + =  ..+ o   |
|   . + + .*.. .  |
|      o So o..   |
|       o . o.    |
|        o . ..   |
|               E |
|                 |
+-----------------+
[root@base nafla]#
[root@base nafla]# cd -ssh
bash: cd: -s: invalid option
cd: usage: cd [-L|-P] [dir]
[root@base nafla]# cd /root/.ssh
[root@base .ssh]# ls
authorized_keys  id_rsa  id_rsa.pub
```

2.15 Modifier le fichier /etc/ssh/ssh_config

```
[root@base nafla]# nano /etc/ssh/ssh_config
```

Modifier:

```
StrictHostKeyChecking no
```

(supprimera les messages de ssh)

Redémarrer le système pour que les modifications prennent effet.

L'action 2 a créé et configuré un système de base.
Tous les nœuds de cluster CDH5 seront clonés à partir de ce nœud de base.

Action 3: Eteindre la machine base (le nœud base) et Cloner 2 nœuds , l'un pour maître et l'autre pour l'esclave (basé sur le nœud base) [43].
Action 4: Lancer les deux nœuds, puis configurer chaque nœud individuellement: [43]

2 machines virtuelles sur VMware, prêts à être mis en place dans le cluster de Cloudera (**Figure III.8)**.

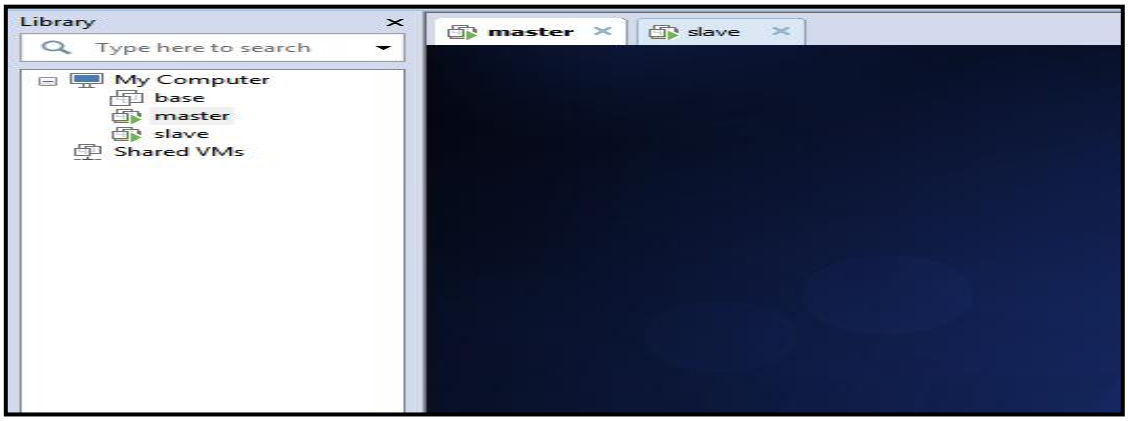


Figure III.8 : 2 machines master et slave

4.1 Changer le nom d'hôte

Par exemple:

connecter au nœud maître (Master) en tant que root

nano /etc/sysconfig/network

```
[nafla@base ~]$ sudo -s
[sudo] password for nafla:
[root@base nafla]# nano /etc/sysconfig/network
```

pour changer le nom d'hôte de base à master

```
NETWORKING=yes
HOSTNAME=master.example.com
GATEWAY=192.168.235.2
```

4.2 Changer l' adresse IP (selon le fichier / etc / hosts)

4.3 Redémarrer le réseau et la machine , exécuter la commande:

```
[root@base network-scripts]# service network restart
```

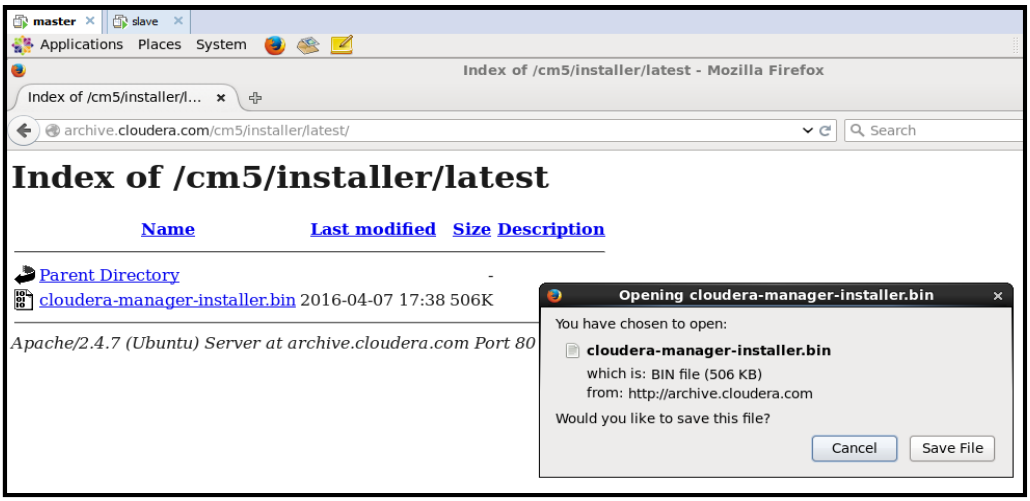
```
[root@base network-scripts]# init 6
```

Après l'action 4, nous pouvons tester la connexion entre les 2 nœuds du cluster (master et slave) avec ping

Action 5: Installer CDH5

5.1 Télécharger Cloudera-manager-installer.bin.

Aller à http://archive.cloudera.com/cm5/installer/latest/



5.2 Installer JDK

Aller au site de Cloudera

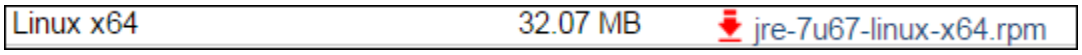
http://www.cloudera.com/downloads/cdh/5-7-0.html

Installer la version recommandée de JDK

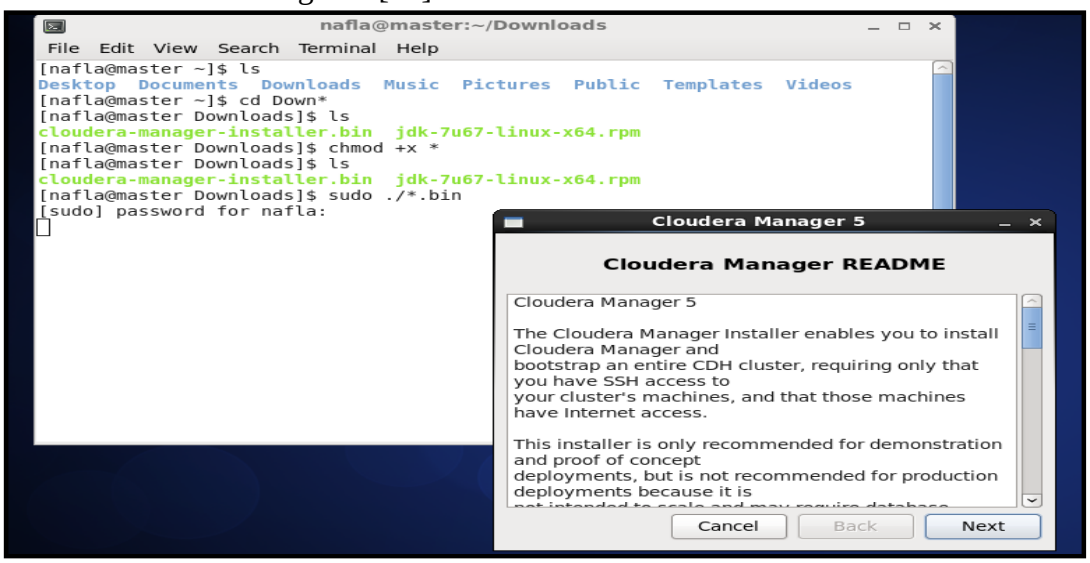
CDH 5.7.x is supported with the versions shown in the following table:

Minimum Supported Version	Recommended Version	Exceptions
1.7.0_55	1.7.0_67, 1.7.0_75, 1.7.0_80	None

Alors je vais installer une des versions recommandées pour ma CDH 5.7.0 , j'ai choisi d'installer JDK 7u67 (disponible à partir du http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html)



5.3 Installer Cloudera Manager [43]



Aperçu :

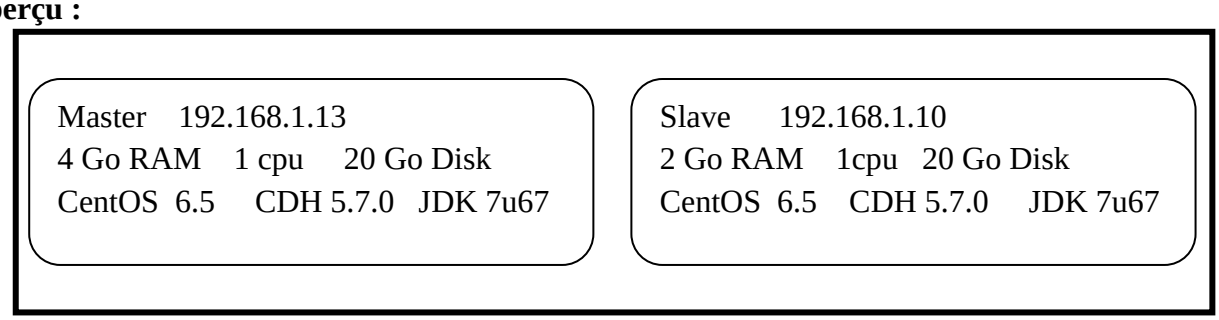
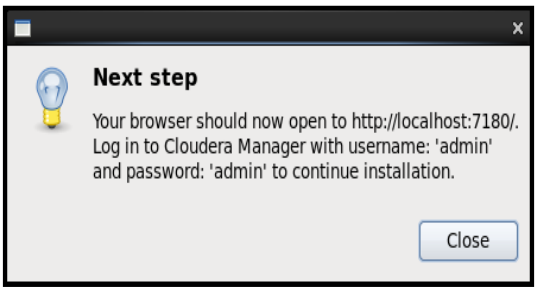


Figure III.9: Diagramme de haut niveau du cluster VirtualBox VM en cours d'exécution nœuds Hadoop.

Action 6: Configurer les paramètres lors de l'installation [42] [43]

6.1. Utiliser un navigateur Web et se connecter à http://localhost:7180.



6.2. Pour poursuivre l'installation, nous devrons sélectionner la version Cloudera de licence gratuite

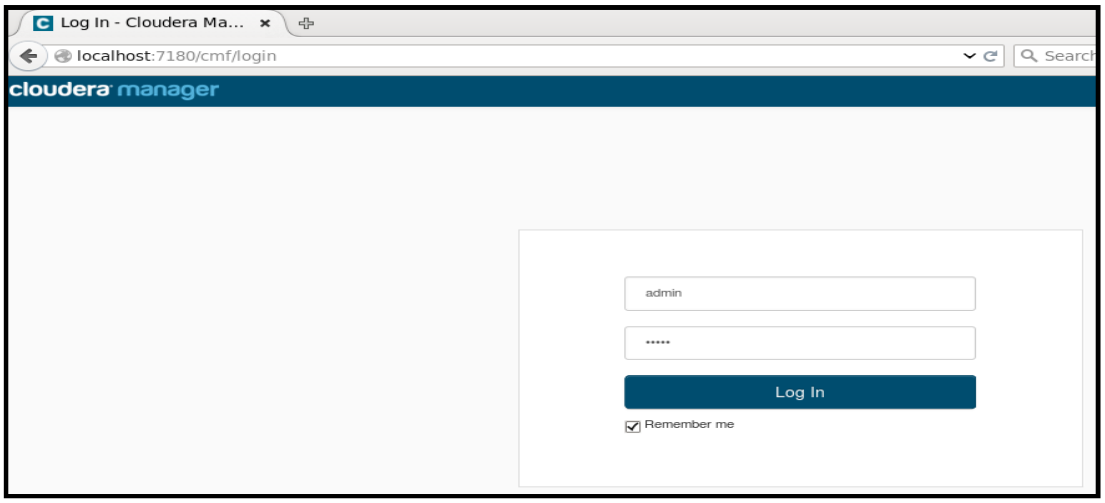
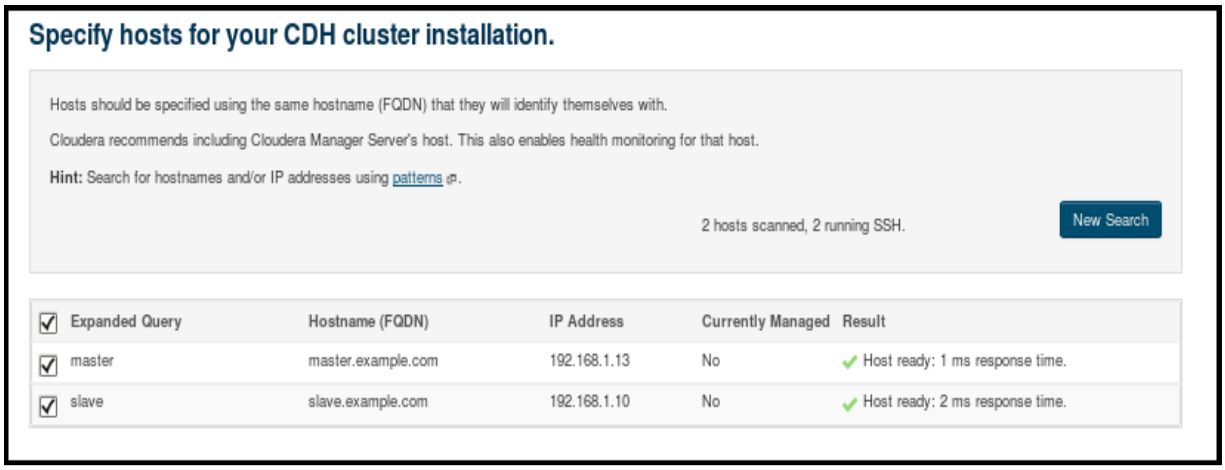


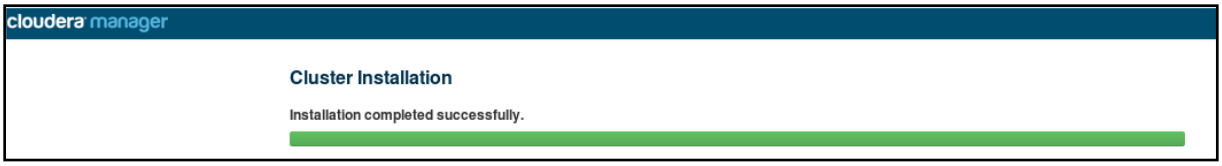
Figure III.10. Cloudera Manager

6.3. Nous devrons alors définir les nœuds qui seront utilisés dans le cluster. Il suffit d'entrer tous les nœuds que nous avons définis dans les étapes précédentes (par exemple master.example.com et slave.example.com) séparés par un espace et de cliquer sur le bouton "Recherche".



6.4. Ensuite utiliser le mot de passe root (ou les clés SSH nous avons générés) pour automatiser la connectivité entre les différents nœuds. Installer tous les paquetages et services sur le premier nœud (master).

6.5. Une fois cela fait, sélectionner les composants de service supplémentaires; il suffit de sélectionner tout par défaut. L'installation se poursuit et se termine.



Action 7: Utilisation du cluster Hadoop

Maintenant, que nous avons un cluster Hadoop opérationnel, il existe deux interfaces principales que nous allons utilisé pour opérer le cluster: Cloudera Manager et Hue.

7.1. Cloudera Manager

Utiliser un navigateur Web et se connecter à http://master.example.com:7180

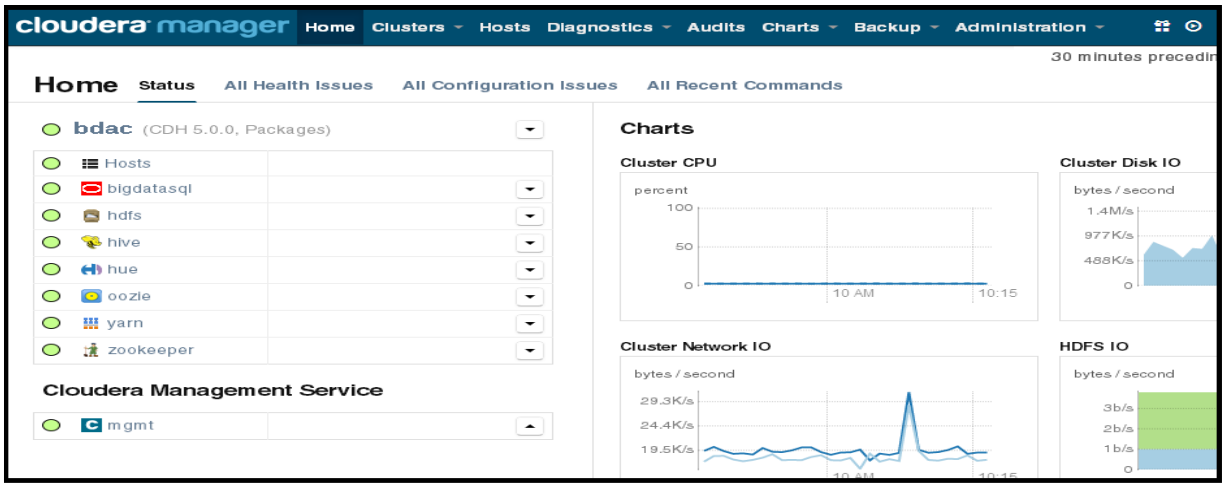


Figure III.11: page d'accueil de Cloudera Manager

7.2. Hue

De même que pour Cloudera Manager, nous pouvons accéder au site d'administration Hue en accédant: http://master.example.com:8888, où nous serons en mesure d'accéder aux différents services que nous avons installés sur le cluster.

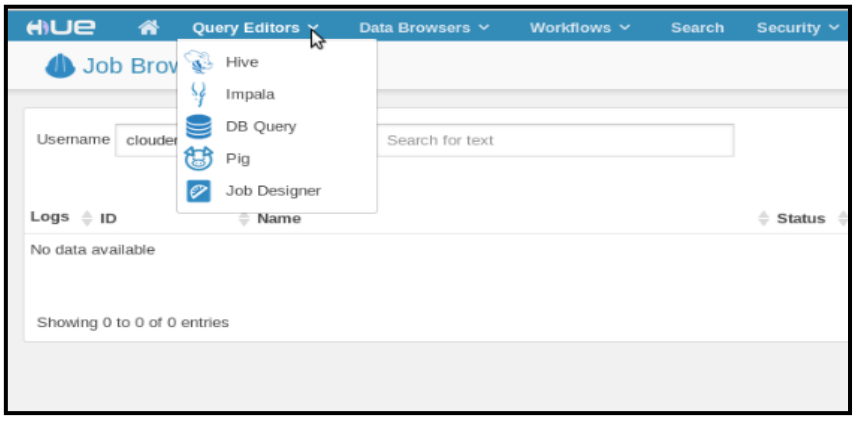


Figure III.12: L'interface Hue

III.6. Prise en main de Hadoop à l'aide d'un exemple : WordCount

MapReduce est un principe qui est au cœur de Hadoop, les données à traiter en entrée sont découpées en “petites” unités (64M par défaut), chacune étant traitée en parallèle par une fonction Map. Le résultat des traitements unitaires est trié par clé pour former des unités de données passées à une fonction Reduce. C’est parce que les programmes MapReduce sont intrinsèquement prévu pour être exécutés en parallèle qu’il est possible de répartir le traitement. Pour expliquer les 2 fonctions Map et Reduce , nous avons opté pour l’exemple de Word count qui permet de compter le nombre d’occurrences de différents mots composant un fichier. En effet, WordCount est surnommé le "Hello World" de Hadoop. Ce programme permet de montrer une bonne illustration du fonctionnement de MapReduce en comptant le nombre d'occurrences de chaque mot dans un document ou un passage de texte.

III.6.1. Illustration du fonctionnement de MapReduce à l'aide de WordCount

Soit le fichier du document en entrée contenant 3 enregistrements et chaque enregistrement contient 3 mots :

```
1. Deer Bear River
2. Car Car River
3. Deer Car Bear
```

Le but de l'illustration est d'appliquer le modèle MapReduce afin de sortir le nombre d'occurrences des mots constituant le texte. L'ensemble du processus est schématisé ci-dessous :

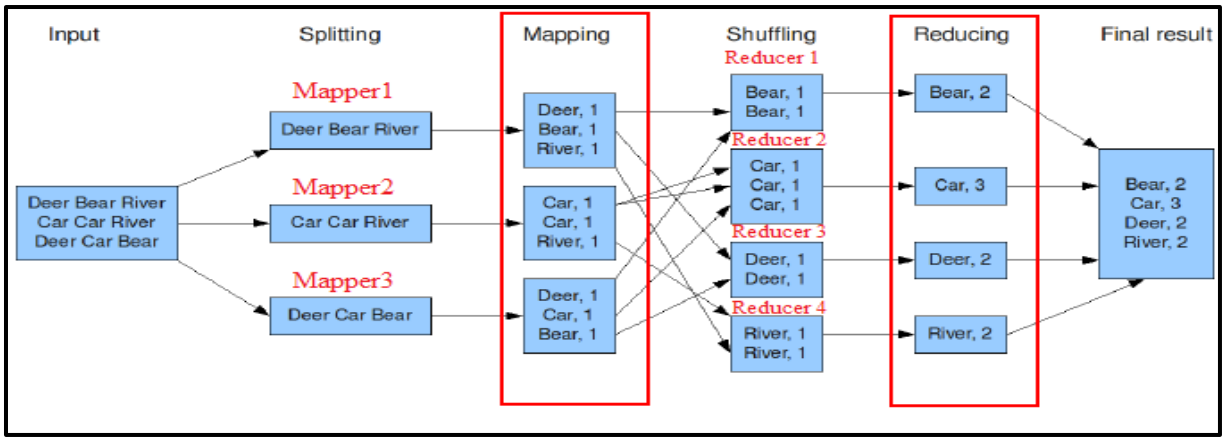


Figure III.13: Les étapes de l'algorithme MapReduce [44].

Les étapes réalisées sont :

• **L'étape Input** : on lit le fichier en entrée

• **L'étape Splitting** : on distribue les données à traiter sur les différents nœuds du cluster

dans notre exemple on a 3 mappers

Mapper n'1 va traiter le premier enregistrement

Mapper n'2 va traiter le deuxième enregistrement

Mapper n'3 va traiter le troisième enregistrement

• **L'étape Map** : les mappers comptent les mots et les fichiers en sortie de ces mappers sont des résultats intermédiaires qui sont stockés sous forme (clé, valeur)

• **L'étape Shuffling** (c'est une étape de tri comme tu peux le voir c'est trié par la clé : les bear ensemble , les car ensemble, …) : Avant le transfert des résultats intermédiaires des mappers vers les reducers

- ✓ Les enregistrements sont triés C.à.d. Les enregistrements correspond à une même clé sont envoyées vers un seul et même reducer

dans notre exemple on a 4 reducer et Mapreduce garantit :

- ✓ Si un reducer recoit le couple `(Bear,1)` de mapper 1, alors il receva aussi le couple `(Bear,1)` de mapper 3

- ✓ donc Tous les enregistrements correspont a la clé Bear seront regroupés et envoyé au même reducer

• **L'étape Reduce** : on effectue la somme de toutes les valeurs de chaque mot
Les fichiers en entré des reducer sont

```
Bear,1        Bear,1
```

Les fichiers en sortie des reducer seront

```
Bear,2
```

• **L'étape Result** : il faut fusionner les fichiers issus des 4 reducers pour obtenir le résultat final.

Enfin j'ai eu de la chance de créer un petit cluster Hadoop. Il est maintenant possible d'exécuter et d'utiliser les différents exemples installés sur le cluster, ainsi que de comprendre les interactions entre les nœuds.

III.6.2 Implemenation de WordCount

Pour l'implémentation de WordCount nous avons choisi un corpus représentant l'ensemble des œuvres de Shakespeare ce qui constitue une importante quantité de mots à considérer.
Le développement et la mise en œuvre d'un programme Hadoop comprennent en général les phases suivantes:

- ➢ Préparation des données
- ➢ Importation des données dans HDFS
- ➢ Ecriture du programme Hadoop et validation en environnement de test
- ➢ Exécution du programme Hadoop en environnement de production
- ➢ Récupération et analyse des résultats

III.6.2.1 Préparation des données

La totalité de l'œuvre de Shakespeare est stockée dans un seul fichier au format Plain UTF-8 provenant du projet Guetenberg .(http://www.gutenberg.org). Ce fichier peut être téléchargé à l'adresse http://www.gutenberg.org/cache/epub/100/pg100.txt . Si cette adresse ne fonctionne pas, lancer une recherche Google sur l'expression `download complets works of William Shakespeare.` Le fichier téléchargé (5.3 MB) est enregistré sur le bureau sous le nom `pg100.txt.`
Le fichier se présente sous la forme d'un ensemble de lignes, chaque ligne se terminant par un signe de nouvelle ligne (\n). Chaque ligne est composée de mots séparés par un espace ou est vide.

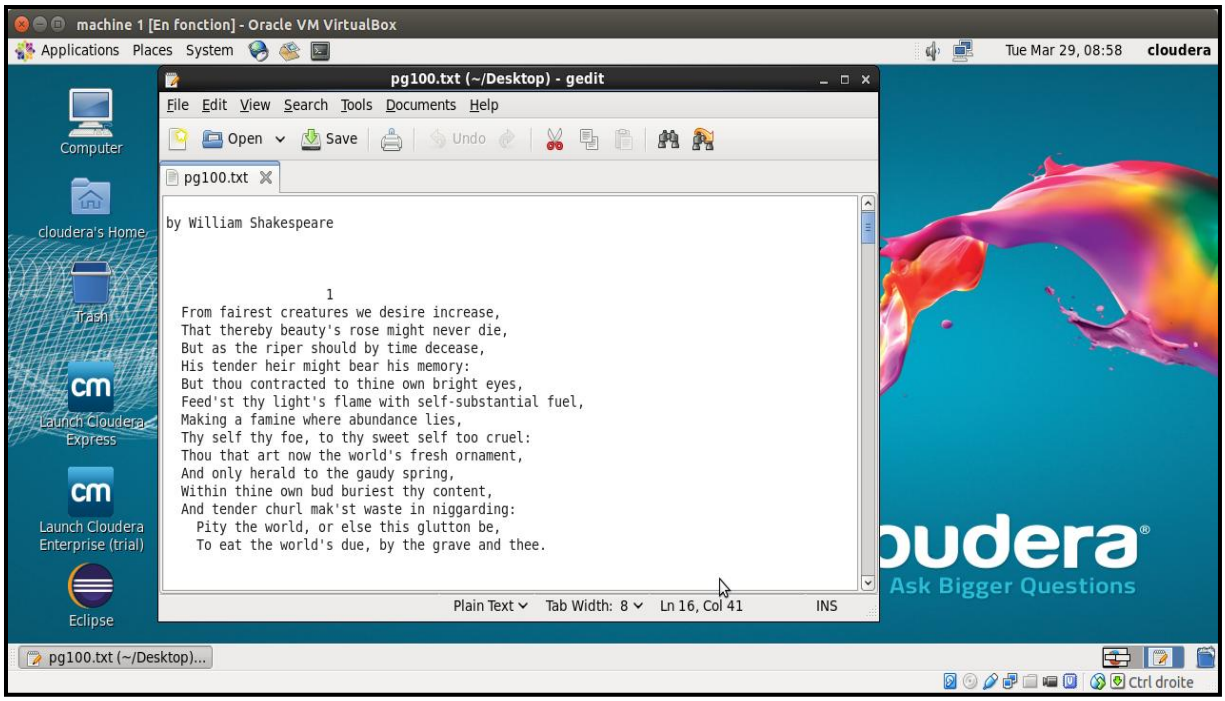


Figure III.14 : Le fichier pg100.txt après préparation des données .

III.6.2.2. Importation des données dans HDFS

Pour importer le fichier pg100.txt dans HDFS, nous avons procédé de la manière suivante :

- ouvrir un terminal
- Dans HDFS, créer le sous répertoire data dans le répertoire courant, puis assure que le répertoire data a bien été crée.
- copier le fichier local pg100.txt dans le répertoire data
- vérifier que le fichier pg100.txt est bien présent

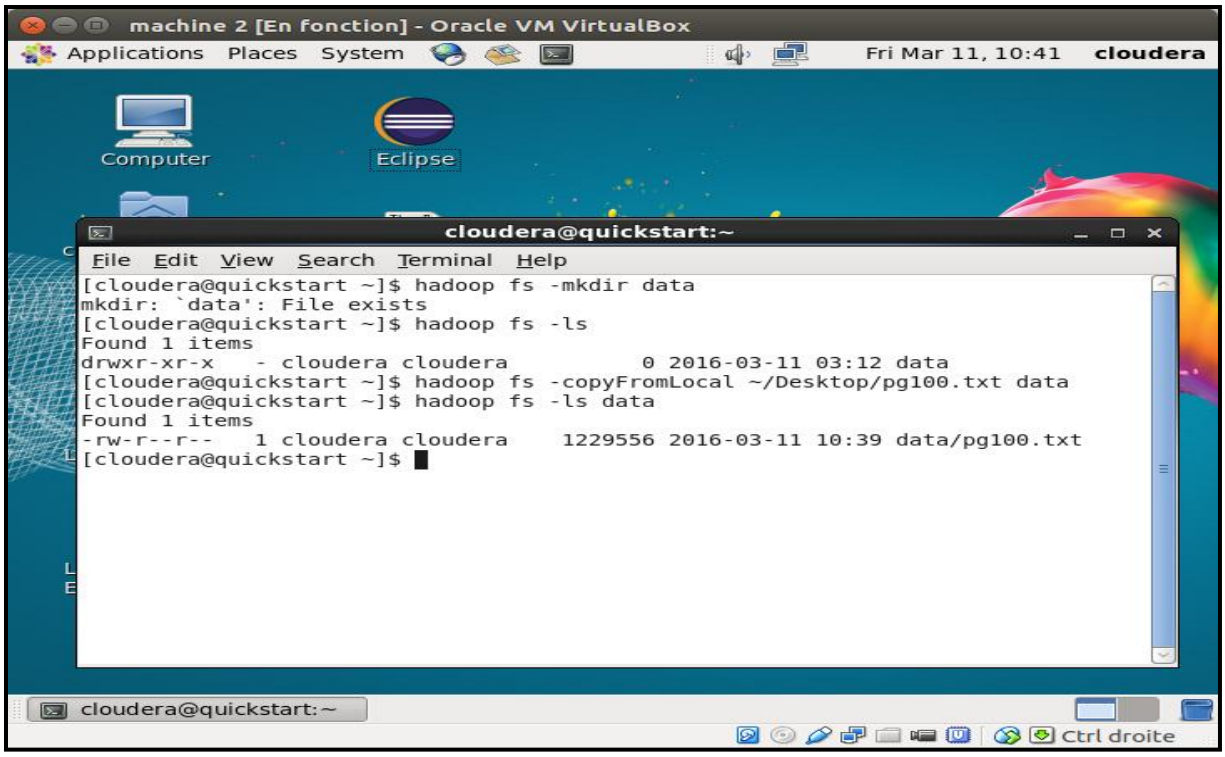


Figure III.15 : Copie de fichier pg100.txt du système de fichier local dans HDFS

III.6.2.3. WordCount en Java

1. Le driver :

Le driver est un programme Java qui s'exécute généralement sur la machine cliente (donc pas dans le cluster Hadoop). Il permet de configurer le job puis de le soumettre au cluster Hadoop pour exécution.

Le code du driver du WordCount se trouve ci-après. Il gère les tâches suivantes ;

- Lignes 1 à 6 : importation des classes Java nécessaire au fonctionnement du driver. Ces classes se retrouvent dans presque tous les drivers Hadoop.
On notera qu'à la ligne 6 le programme fait appel à la nouvelle API, `Org.apache.hadoop.mapreduce.*`, par opposition à l'ancienne API, `Org.apache.hadoop.mapred.*`, utilisée avant la version 0.20.1 de Hadoop.

- Lignes 12 à 15 : vérification dans la méthode main que les deux arguments attendus de la ligne de commande (le répertoire à utiliser en entrée et le répertoire à utiliser en sortie) sont bien présents. Si le nombre d'arguments n'est pas égal à 2, fin du programme et affichage d'un message d'erreur.

- Lignes 17 et 18 : instanciation d'un nouvel objet de type job qui est utilisé pour configurer le job Hadoop (`WordCountDriver.class`).

- Ligne 20 : attribution d'un nom explicite au job Hadoop `(Word Count)` pour permettre de le repérer plus facilement dans les logs d'exploitation.

- Lignes 22 et 23 : spécification des chemins des fichiers, en entrée et en sortie (ces chemins sont passés comme paramètres dans la ligne de commande).

- Lignes 25 et 26 : indication à l'objet job des classes à utiliser telles mapper `(WordCountMapper.class)` et reducer `(WordCountReducer.class).`

- Lignes 28 et 29 : indication du type de couples (key, value) en sortie de mapper (Text , IntWritable).

- Lignes 31 et 32 : indication du type des couples (key, value) en sortie de reducer (Text , IntWritable).
- Lignes 34 et 35 : lancement du job, attente de sa fin d'exécution et retour du code de fin de programme (0= OK, 1= problème).

Le driver peut également indiquer (ce n'est pas le cas dans l'exemple de WordCount)
Les formats de données à utiliser en entrée, par exemple job.setInputFormatClass `(KeyValueTextInputFormat.class),` et/ou en sortie, par exemple job.setOutputFormatClass `(TextOutputFormat.class).` Par défaut, le format utilisé, tant en entrée qu'en sortie, est `TextInputFormat.`

Le code de driver en java:

```
1:  import org.apache.hadoop.fs.Path;
2:  import org.apache.hadoop.io.IntWritable;
3:  import org.apache.hadoop.io.Text;
4:  import
org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
5:import
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
6:  import org.apache.hadoop.mapreduce.Job;
7
8:  public class WordCountDriver {
9:
10: public static void main(String[] args) throws Exception {
11
12: if (args.length != 2) {
13: System.out.printf("Format de la ligne de commande :
WordCount <input dir> <output dir>\n");
15: System.exit(-1);
16: }
17: Job job = new Job();
```

```
18: job.setJarByClass(WordCountDriver.class);
19:
20: job.setJobName("Word Count");
21:
22: FileInputFormat.setInputPaths(job, new Path(args[0]));
23: FileOutputFormat.setOutputPath(job, new Path(args[1]));
24:
25: job.setMapperClass(WordCountMapper.class);
26: job.setReducerClass(WordCountReducer.class);
27:
28: job.setMapOutputKeyClass(Text.class);
29: job.setMapOutputValueClass(IntWritable.class);
30:
31: job.setOutputKeyClass(Text.class);
32: job.setOutputValueClass(IntWritable.class);
33:
34: boolean success = job.waitForCompletion(true);
35: System.exit(success ? 0 : 1);
36:    }
37: }
```

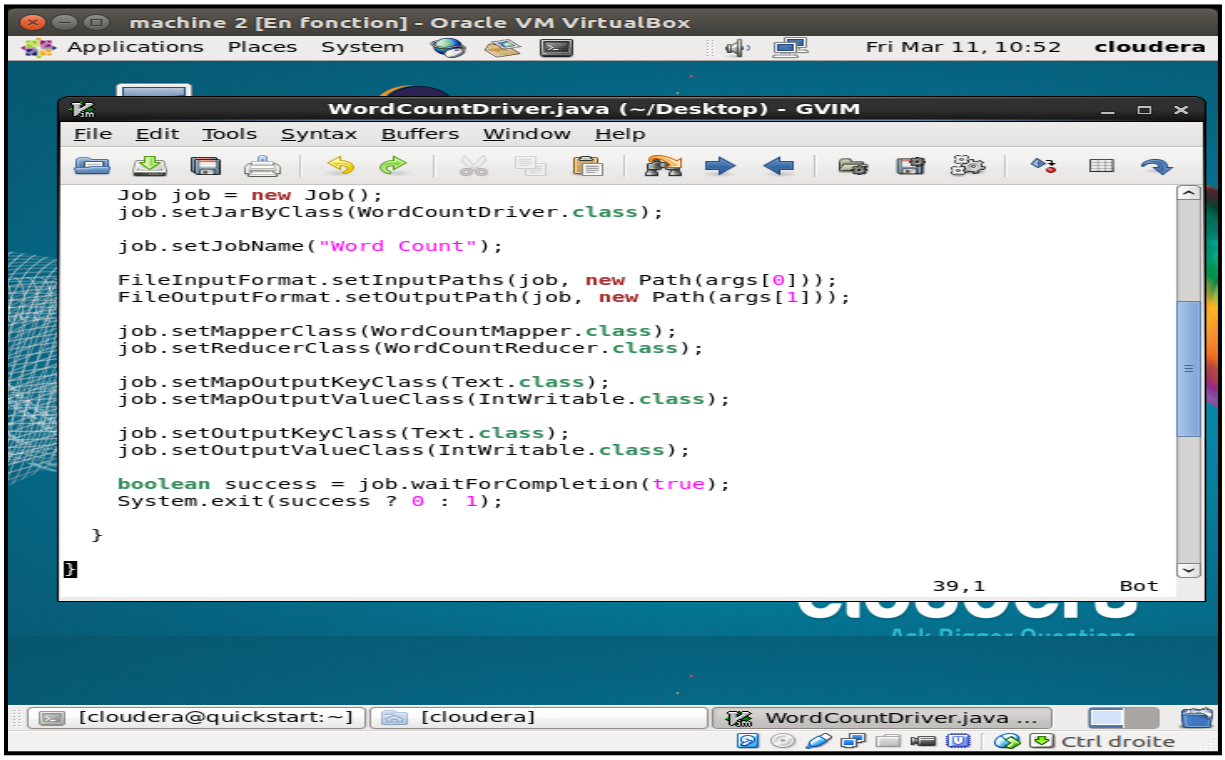


Figure III.16 : Le fichier WordCountDriver.java

2. Le mapper

Le mapper est un programme Java exécuté en parallèle sur plusieurs nœuds esclaves (*slave nodes*) du cluster Hadoop, chaque instance étant un mapper. Chaque mapper comptre le

nombre d'occurrences d'un mot dans une partie des œuvres de Shakespeare (les reducers se chargeant de synthétiser le travail des mappers).

Le code du mapper du WordCount se trouve ci-après. Il gère les tâches suivantes :
-Lignes 1 à 5 : importation des classes Java nécessaire au fonctionnement du mapper.

-Lignes 7 à 20 : définition de la classe `WordCountMapper` qui est appelée par le driver (cf. ligne 25 du driver).

-Ligne 10 : indication du type des couples (key, value) en entrée du mapper `(LongWritable, Text). LongWritable` correspond à l'offset de la ligne depuis le début du fichier et Text à la ligne de texte elle-même.

-Ligne 12 : récupération de la valeur de la variable `value`, qui est donc du type text, dans la variable `line,` qui est de type `string.`

-Lignes 14 à 16 : ces lignes sont de le cœur du mapper.

-Ligne 14 : la ligne de texte de prise en charge par le mapper des découpée en mots à l'aide de la fonction `split () ,` chaque mot étant successivement stocké dans la variable `word (cf.boucle for).`
-lignes 15 et 16 : chaque mot est passé en revue et , s'il comprend au moins un caractère `(word.lenght () > 0),` un coupme `(word, 1)` est émis par le mapper. Dans le cas contraire, rien ne se passe. Dans ces lignes le mapper enregistre simplement une nouvelle occurrence du mot contenu dans la variable `word.` Il n'est pas encore question de comptage : ce sera la tâche du reducer.

Le code de mapper en java

```
1:  import java.io.IOException;
2:  import org.apache.hadoop.io.IntWritable;
3:  import org.apache.hadoop.io.LongWritable;
4:  import org.apache.hadoop.io.Text;
5:  import org.apache.hadoop.mapreduce.Mapper;
6:
7:  public class WordCountMapper extends Mapper<LongWritable,
Text, Text, IntWritable> {
8:
9:  @Override
10: public void map(LongWritable key, Text value, Context
context) throws IOException, InterruptedException {
11:
12:  String line = value.toString();
13:
```

```
14:   for (String word : line.split("\\W+")) {
15:   if (word.length() > 0) {
16:   context.write(new Text(word), new IntWritable(1));
17:       }
18:     }
19:   }
20:}
```

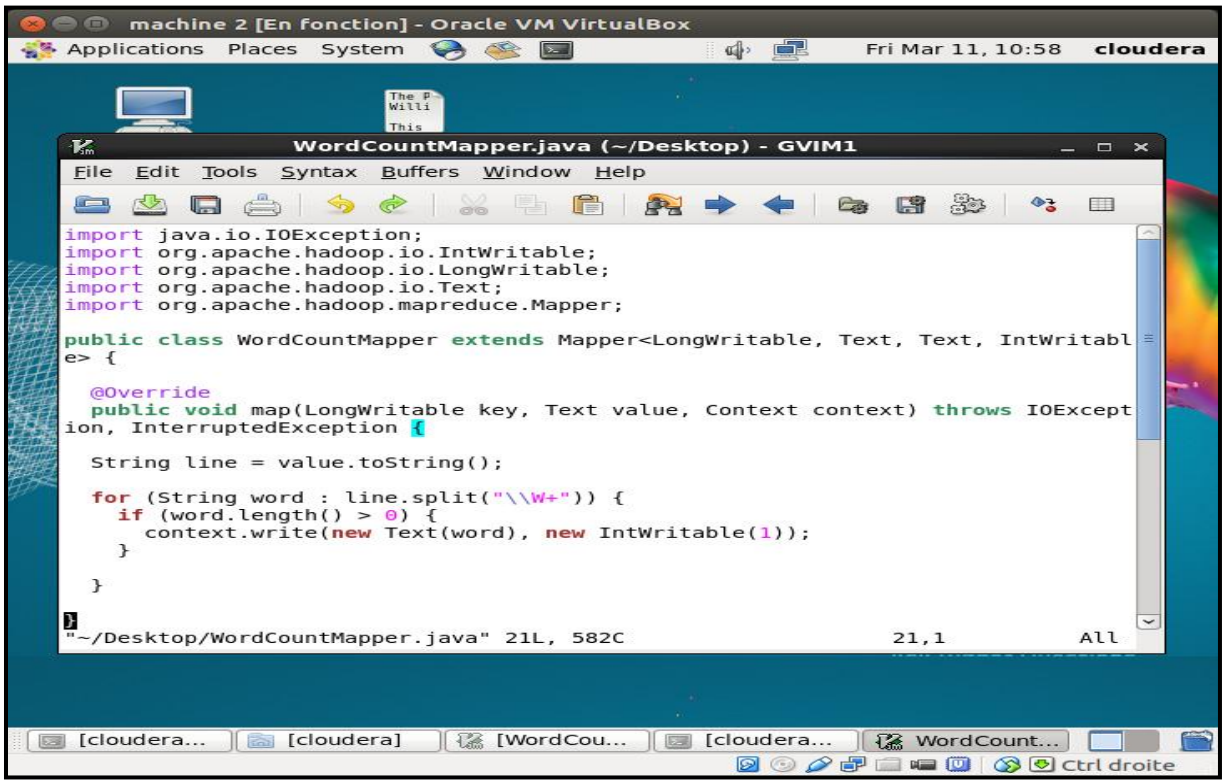


Figure III.17: Le fichier WordCountMapper.java

3. Le reducer

Le reducer est un programme Java exécuté en parallèle sur plusieurs nœuds esclaves (*slave nodes)* du cluster Hadoop, chaque instance étant un reducer.
Chaque reducer ;
-Se voit affecter par Hadoop un sous-ensemble de l'ensemble des mots constituant les œuvres de Shakespeare.
-Est chargé, pour chaque mot de ce sous-ensemble, de cumuler les comptages de ce même mot issus, le cas échéant, de différents mappers.

Important : il convient de garder à l'esprit que les différentes occurrences d'un mot donné des œuvres de Shakespeare, par exemple l'article `"the",` pourront être prises en compte par plusieurs mappers. Par contre, **toutes les occurrences** de `"the"` seront envoyées par Hadoop à

un seul et même reducer même si elles proviennent de mappers différents. C'est ce qui permet de garantir qu'aucune occurrence de "the" ne sera oubliée .

Le code du reducer de WordCount se trouve ci-après. Il gère les tâches suivantes :
-Lignes 1 à 4 : importation des classes Java nécessaires au fonctionnement du reducer.

-Lignes 6 à 20 : définition de la classe WordCountReducer qui sera appelée par le driver (§.ligne 26 du driver).

-Ligne 9 : indication du type des données en entrée du reducer.il s'agit d'une liste de couples (key , value), key etant de type Text (c'est un mot issu des œuvres de Shakespeare, par exemple "the" et value étant une liste (Iterable) de valeurs de type IntWritable (il s'agit d'une liste des 1 émis par le mappers à chaque occurrence de "the").

-Ligne 11 : mise à zéro du compteur des différentes occurrences d'un mot.

-Ligne 13 à 15 : ces lignes sont le cœur du reducer.

-Lignes 13 : les différentes occurrences (§.boucle for) du mot passé dans key sont comptées (ligne14).plutôt que d'ajouter 1 à la variable wordCount , on ajoute value.get( ) ce qui revient exactement au même dans le cas de l'exemple de WordCount.

-Ligne 16 : une fois que toute la liste de comptage a été parcourue, le reducer écrit le mot (key) et le résultat du comptage (wordCount) dans l'objet context. Comme la variable wordCount est initialement définie comme variable du type Java int (§ ligne 16, new IntWritable (wordCount)) avant de pouvoir être écrite.

Le code de reducer en java

```
1:  import java.io.IOException;
2:  import org.apache.hadoop.io.IntWritable;
3:  import org.apache.hadoop.io.Text;
4:  import org.apache.hadoop.mapreduce.Reducer;
5:
6:  public class WordCountReducer extends Reducer<Text,
IntWritable, Text, IntWritable> {
7:
8:  @Override
9:  public void reduce(Text key, Iterable<IntWritable> values,
Context context) throws IOException, InterruptedException {
10:
11:  int wordCount = 0;
12:
13:  for (IntWritable value : values) {
```

```
14:   wordCount = wordCount + value.get();
15:     }
16:   context.write(key, new IntWritable(wordCount));
17:   }
18:}
```

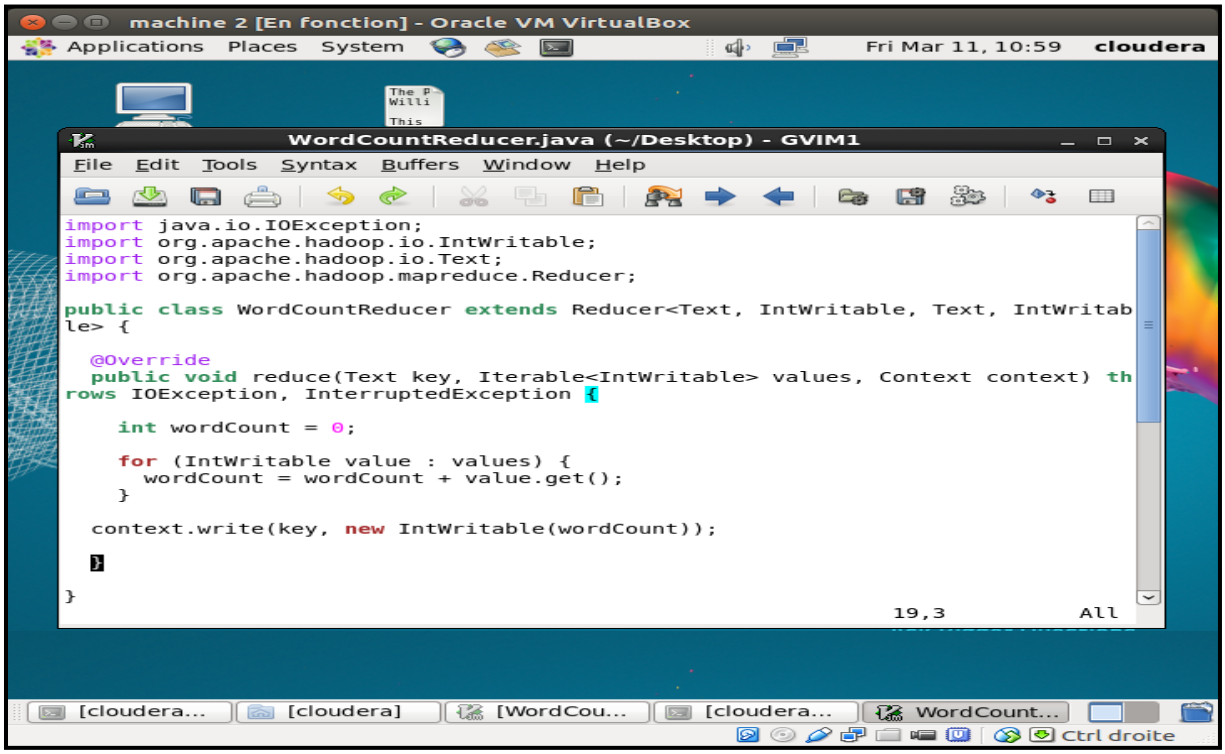
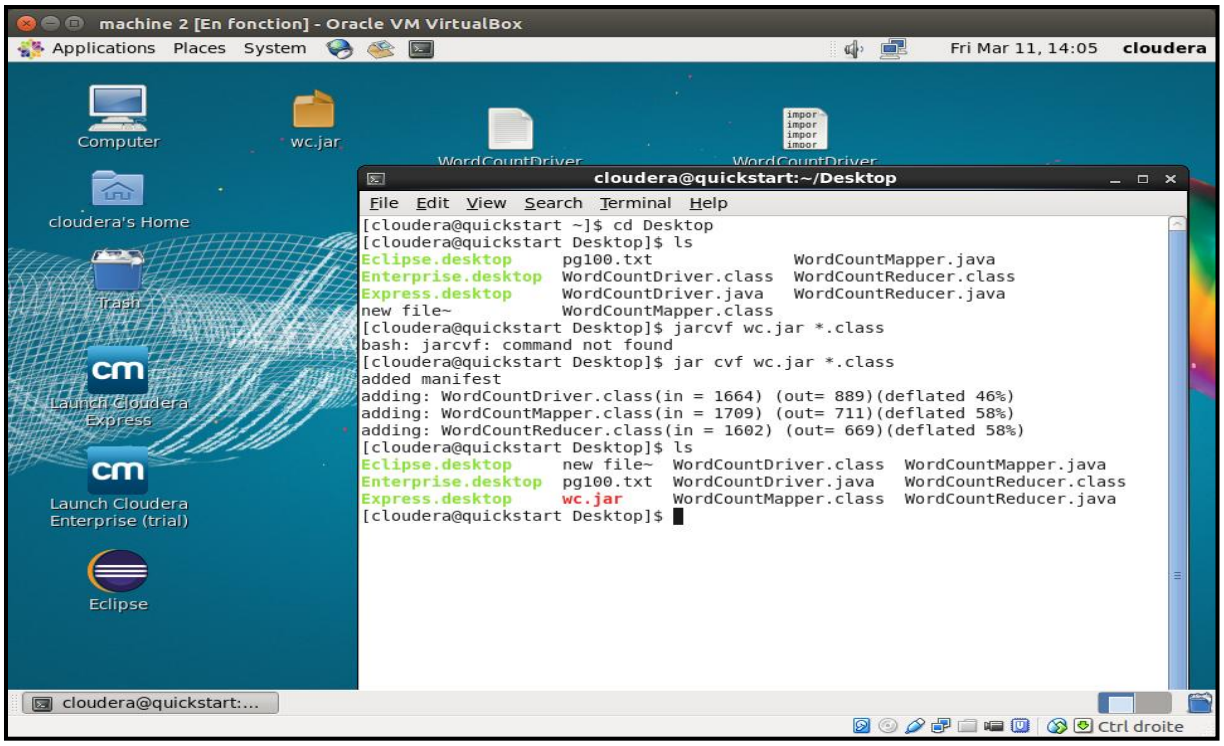


Figure III.18: Le fichier WordCountReducer.java

III.6.2.4 Compilation et Exécution du job

Compilation du Job:

- Ouvrer le Terminal
- Positionnons-nous au niveau du Bureau
- Vérifier que les trois fichiers Java (driver, mapper et reducer) sont présents
- Nous Compilons le driver, le mapper et le reducer.
- Nous vérifions que les trois fichiers compilés en un fichier JAR exécutable



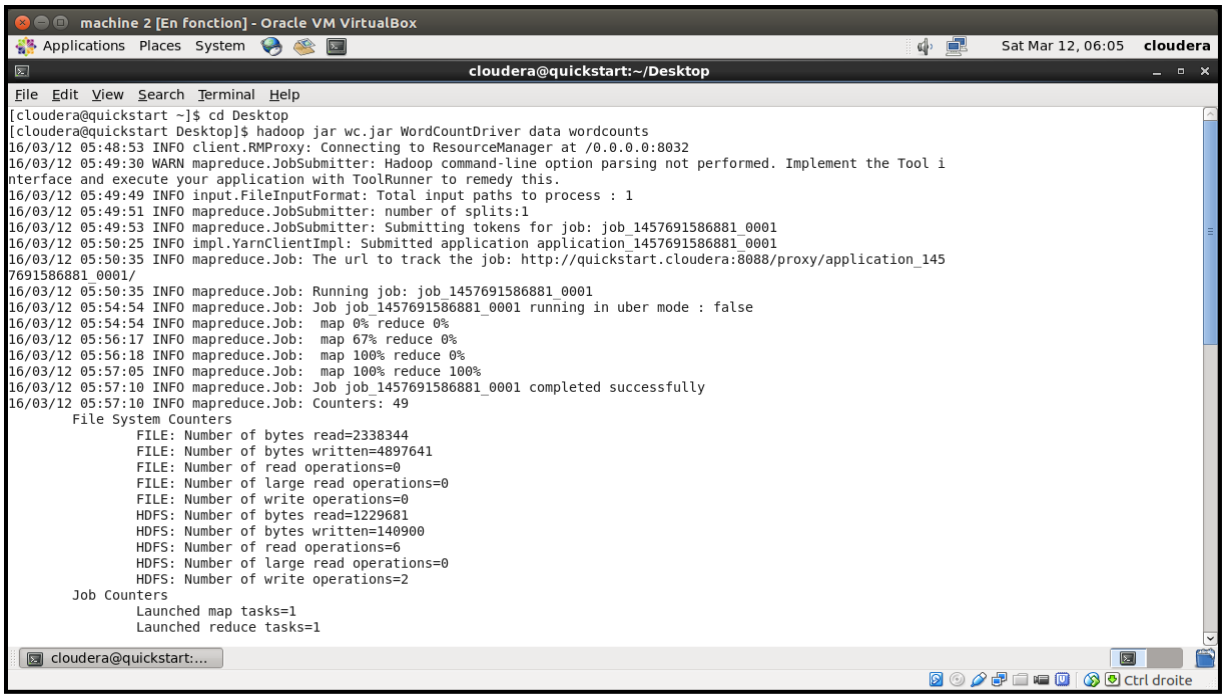
Exécution du Job:

Pour exécuter le programme WordCount, Procéder de la manière suivante:

- Ouvrez le Terminal
- Positionnez-vous au niveau du Bureau
- Lancer l'exécution de WordCountDriver.jar en précisant le répertoire en entrée (dans lequel se situe le fichier des œuvres complètes de Shakespeare) et le répertoire en sortie (dans lequel seront stockés les fichiers de résultats -un fichier par reducer, puis patientez...
- une fois le job terminé, affichez les résultats

Les commandes à utiliser sont détaillées ci-dessous:

CHAPITRE III : Installation et prise en main de Hadoop



La commande permettant de lancer WordCount s'analyse de la façon suivante :

- hadoop : la commande à executer est une commande Hadoop, pas Linux
- jar : la commande à executer est wc.jar

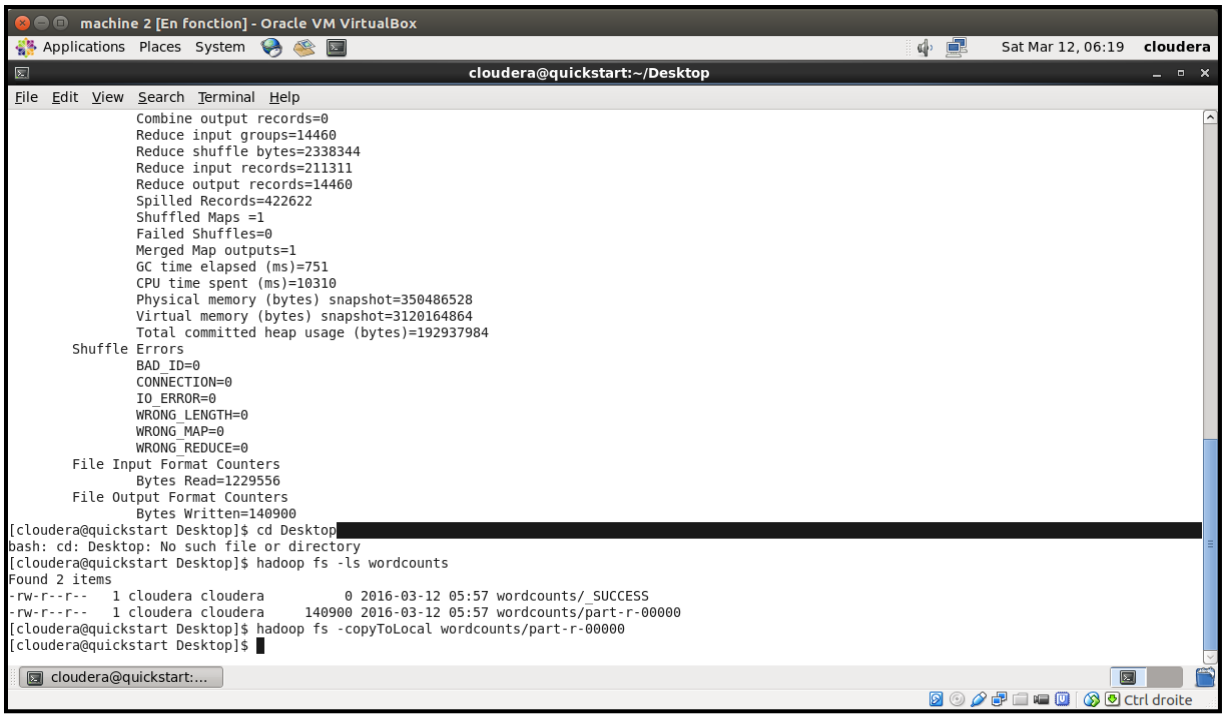
-WordCountDriver: nom de la classe à appeler pour lancer le job

-data: repertoire contenant les données en entrée

-wordcounts:repertoire contenant les resultats en sortie

Pour afficher les résultats, il faut :

- Noter le nom fichier contenant les résultats issus de reducers (un fichier par reducer, donc un seul fichier dans notre cas car, en mode local, un seul reducer est utilisé)
- Copier le fichier de Hadoop sur le bureau
- Visualiser le contenu du fichier a l'aide d'un éditeur de texte (nous avons opté pour `gedit`)



III.6.2.5. Récupération et analyse des résultats

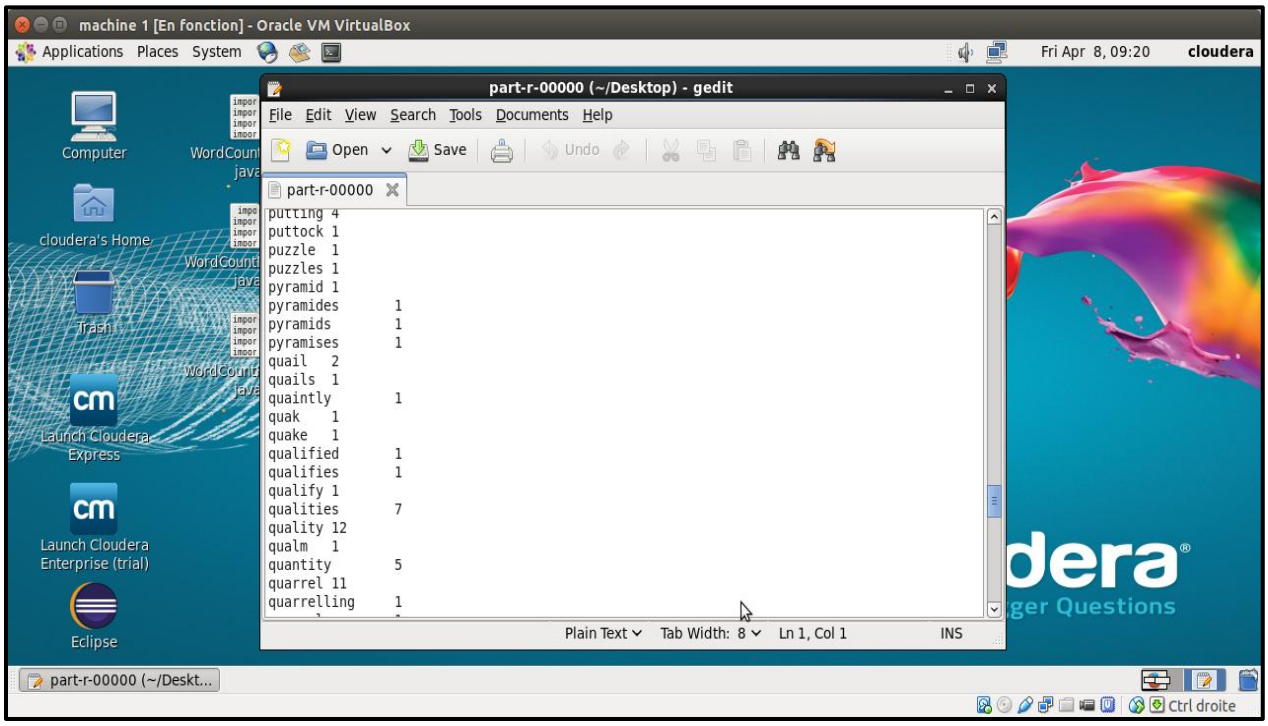


Figure III.19: les résultats de WordCount

III.7.Conclusion

Hadoop est un des sujets à la mode pour le moment, Big Data, architecture distribuée, inspirée de nombreux projets à succès dont Google, Hadoop fait maintenant partie des nombreux projets de la fondation Apache. Si vous devez travailler avec d'énormes masses de

données, et on ne parle pas ici de quelques gigas mais plutôt de petas, alors des projets comme Hadoop seront vos amis. HBase, MapReduce ou HDFS pour le stockage distribué, ces concepts sont la base de l'architecture. A ce jour, quelques grands noms utilisent Hadoop dont le plus connu, Facebook, mais aussi Yahoo ainsi que Microsoft.

Dans ce chapitre, nous avons illustré comment fonctionne Hadoop et détaillé les étapes de son installation afin que n'importe quel utilisateur, et quelques soit le domaine d'application, puisse se servir de ce mémoire comme un guide pour une installation et une utilisation réussies de Hadoop.

CHAPITRE IV :Cas réel d'application du Big data

IV.1. Introduction
IV.2. Impact de l’analyse des météorologiques
IV.3. L'utilisation de Pig
IV.4. L'analyse des données météorologiques avec Pig
IV.5.Conclusion

CHAPITRE IV :Cas réel d'application du Big data

IV.1. Introduction

Pour le moment, nous n'avons pas eu accès à cette technologie de l'internet des objets qui produisent des données massives que l'on peut transformer en des informations et de connaissance. Néanmoins il y a sur internet de nombreuses sources de données pouvant servir à notre travail. En effet, de nombreuses données sont aujourd'hui mises à dispositions par des internautes, des états comme la France où les Etats Unis mettent aujourd'hui des données de divers domaines en ligne. L'administration Obama a ainsi lancé le site web www.data.gov, mine de données, services web et outils cartographiques permettant aux collectivités locales, entreprises et citoyens américains de s'informer et se prémunir des effets du changement climatique (inondations et sécheresses dans la phase pilote actuelle ; impacts sur la santé, les écosystèmes et les infrastructures énergétiques). La Maison Blanche a aussi enrôlé une foule d'organisations publiques et privées (dont les incontournables mastodontes Google, Microsoft, IBM ou Amazon) pour qu'elles mettent à disposition des chercheurs des capacités de calcul et de stockage informatique ou qu'elles développent des applications mettant en évidence les risques climatiques [45].

IV.2. Impact de l'analyse des météorologiques

Nous avons choisi de travailler sur un jeu de données météorologiques mises en ligne par Météo France. L'ensemble des données est accessible sur Git Hub.

Ce choix a été opéré car ces données ont impact réel sur divers domaines. Nous avons choisi de l'illustrer à travers deux domaines hautement importants à savoir : la climatologie et l'agriculture.

IV.2.1. Impact sur la climatologie

On peut se poses la question de savoir si le Big Data sauvera le climat ? La question peut paraître provocatrice mais elle mérite d'être posée. La masse astronomique et exponentielle de données numériques que nous produisons grâce aux énergivores «data centers» n'est-elle pas déjà responsable de 2% des émissions mondiales de CO2 ? Peut-être, mais selon les études, l'analyse de ces données climatologiques a un énorme potentiel [45].

Comme nous allons le montrer dans la suite de ce chapitre, les données météorologiques peuvent avoir un réel intérêt sur le domaine de la climatologie. L'analyse de ces données permettra par exemple, de participer à la production d'énergies renouvelables, d'anticiper des phénomènes en analysant et en croisant de nombreuses domaines désormais disponibles et de se prémunir de catastrophes comme les feux de forêts, les inondations etc.

IV.2.2. Impact sur l'agriculture

Un autre exemple est celui du projet InfoClim [46] qui met les données climatiques à la disposition des collectivités locales et contribue à l'adaptation des agriculteurs sénégalais aux changements climatiques. Les petits exploitants jouissent de nombreuses années d'expérience dans l'évaluation de l'impact des conditions climatiques, surtout la pluviométrie, sur leurs cultures. Mais à mesure que le climat change, ce savoir, souvent accumulé pendant toute une vie, pourrait ne plus être valable. Les agriculteurs vulnérables ont donc besoin de soutien pour

adapter ou affiner leurs pratiques. Pourtant, à mesure que la surveillance et la recherche climatique deviennent plus sophistiquées, l'écart entre la technologie et les communautés d'agriculteurs ne fait que se creuser, mais le projet InfoClim mis en œuvre au Sénégal contribue maintenant à combler cet écart.

Un projet comme InfoClim est une bonne illustration de la façon dont les décideurs locaux peuvent utiliser les données scientifiques pour intégrer les questions liées aux changements climatiques dans les plans de développement local. Ce projet a eu pour effet d'accroître la confiance des agriculteurs en renforçant leurs compétences et en les dotant d'outils pour traduire les données scientifiques et techniques en messages simples et compréhensibles. Les agriculteurs ont besoin de données précises sur les événements climatiques qui affectent leurs cultures [45].

Nous pouvons inous nspirer de ce projet pour l'agriculture en Algérie par exemple en collectant en croisant de nombreuses données pour aider les populations vulnérables, surtout les agriculteurs, à mieux adapter leurs semailles, labour et autres événements par rapport au climat actuel.

Un projet comme ça peut fournir aux communautés et aux agriculteurs locaux des informations précieuses comme des statistiques sur l'état des sols,. Il peut également les aider à partager leurs connaissances pour améliorer les pratiques culturales et assurer de meilleurs rendements. Il s'agit, entre autres, de changer les dates de semailles, d'utiliser des semences résistantes à la sécheresse, de diversifier les cultures et adopter des cultures pérennes, d'améliorer la gestion de l'eau et du sol, de lutter contre l'érosion des sols, de développer l'agroforesterie, d'intégrer les cultures, l'élevage et les arbres, et d'identifier des sources alternatives de revenus.

IV.3. L'utilisation de Pig

Le paradigme **MapReduce** est fondamentalement adapté pour paralléliser des traitements sur un volume important de données, ce qui a valu à **Hadoop** le succès que l'on sait. Mais il atteint ses limites dès qu'il s'agit d'implémenter des traitements plus complexes que le simple comptage de mots dans un corpus de textes, fussent-ils volumineux. L'écriture de fonctions *Map/Reduce* peut s'avérer être une opération très fastidieuse, et même répétitive puisque certains traitements comme les filtres sont très courants dès qu'il s'agit d'analyser des données. Mais surtout, il faut reconnaître que cette tâche n'est pas à la portée du premier consultant en analyse décisionnelle [44.5]

Les projets Hive et Pig, tous deux placés sous la bannière de la fondation Apache, apportent un modèle de développement de plus haut niveau, et donc beaucoup plus expressif et simple à appréhender, afin de démocratiser l'écriture de traitements MapReduce**.** Hive propose un modèle de programmation dérivé du langage SQL. Pig se rapproche plus d'un ETL où l'on part d'un ou plusieurs flux de données que l'on

transforme étape par étape jusqu'à atteindre le résultat souhaité. Les différentes étapes de la transformation sont exprimées dans un langage procédural (Pig Latin).

Hive aura la faveur de quiconque est familier du langage SQL, mais nous pensons que Pig est plus adapté à l'univers de l'informatique décisionnelle où dominent les représentations orientées flux de données couplées à un processus d'élaboration par étape. Comparée à Hive, la courbe d'apprentissage est certes plus importante mais l'effort en vaut la peine [44.5]. C'est précisément ce que nous tâcherons de démontrer dans ce qui suit à travers l'analyse de données météorologiques avec Pig**.**

IV.4. L'analyse des données météorologiques avec Pig

Nous avons utilisé deux jeux de données issus de Météo France. L'ensemble des données et exemples de code sont accessibles sur GitHub.

Le premier fichier de données utilisé (meteo.csv) contient les relevés météorologiques des stations au format CSV :

```
indicatif OMM;date;temperature;temps present;direction du
vent;force du vent;pression
07005;2012-11-13 18:00:00;101;10;120;41;102960
07005;2012-11-14 00:00:00;79;0;130;46;102950
```

L'indicatif OMM correspond à l'identifiant de station météorologique.

Le second fichier (stations.csv) contient l'ensemble des stations météorologiques en France métropolitaine, identifiées par l'indicatif OMM :

```
Id;Ville;indicatif OMM
1;Abbeville;7005
2;Lille-Lesquin;7015
```

Conçu pour ingérer à peu près tout et n'importe quoi (d'où son nom), Pig n'impose aucune contrainte sur le format des données à traiter. Néanmoins, il est très pratique d'indiquer au chargement la structure et le type des données, particulièrement lorsqu'elles sont au format CSV.

Nous avons utilisé l'interface Hue qui nous a permet d'éditer des scripts avec Pig.Nous avons téléchargé les fichiers des jeux de données dans HDFS

En mode local, les fichiers de données doivent être localisés dans le même répertoire que le répertoire d'exécution du programme Pig.

En mode pseudo-distribué ou **distribué** , les fichiers de données doivent d'abord être déposés sur le système de fichiers distribué HDFS.
Si vous disposez d'un environnement Hadoop distribué ou pseudo-distribué, les fichiers peuvent être copiés via les commandes suivantes :

```
hadoop fs -mkdir /Data
hadoop fs -put ./meteo.csv /Data/
```

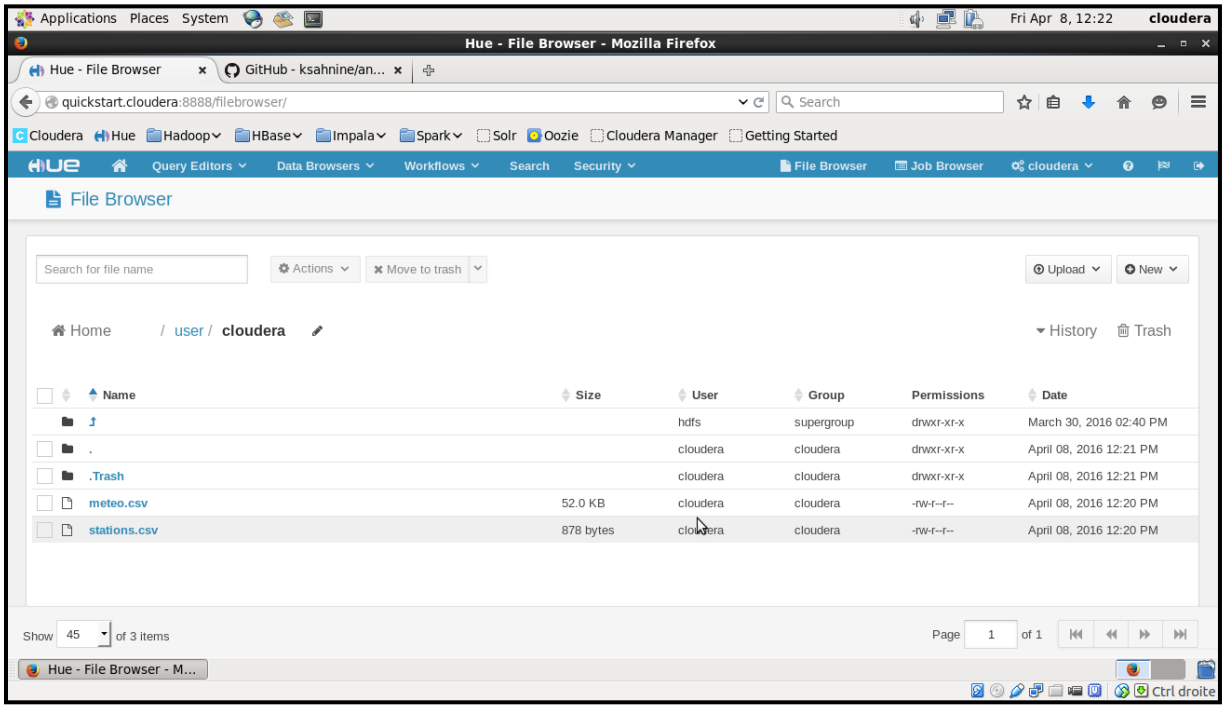


Figure IV.1 : chargement des données dans HDFS

IV.4.1. Premier script

Essayons de sortir la liste des 5 villes les plus froides de France le 15 Novembre 2012 à midi. Pour ce travail, nous avons utilisé deux sources de données. La première contient l'ensemble des relevés météorologiques, la seconde contient la liste des stations référencées en France métropolitaine.

CHAPITRE IV :Cas réel d'application du Big data

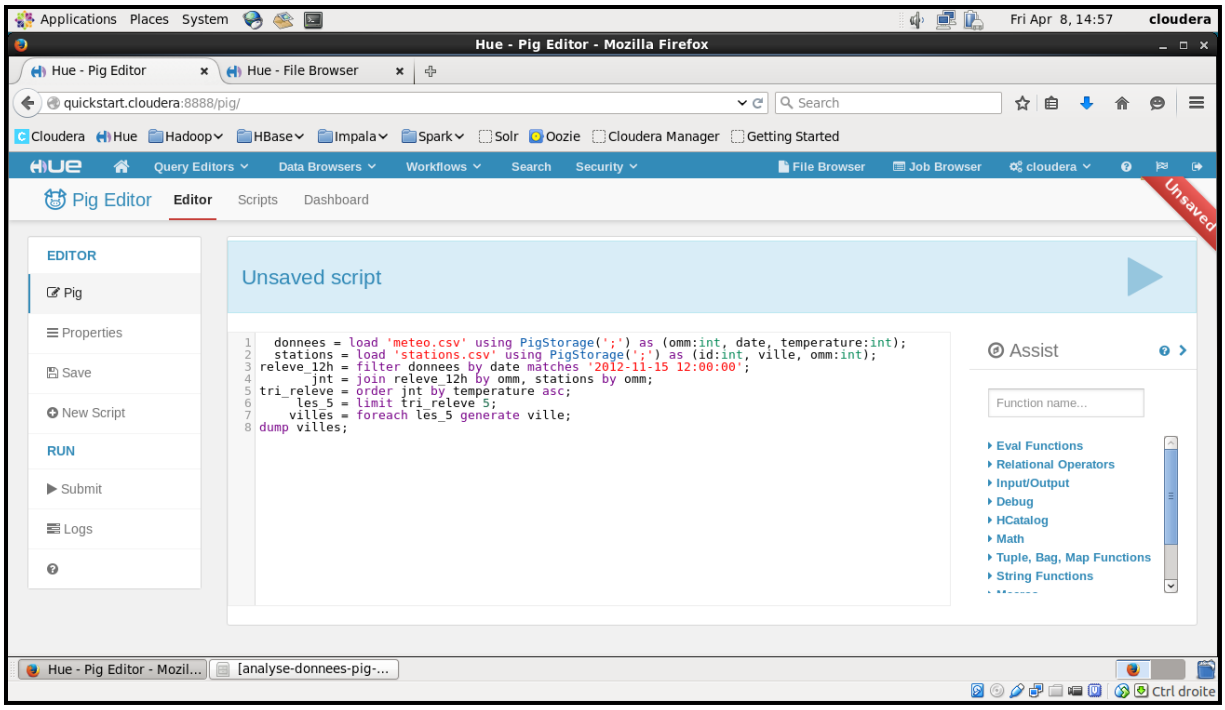


Figure IV.2 : script édité par Pig

➢ Charger les données

On notera les points suivants :

- utilisation de la clause using suivie du délimiteur de champ (PigStorage(';'))
- Pig ignore la première ligne d'en-tête du fichier CSV au chargement
- les fichiers de données sont automatiquement décompressés s'ils portent l'extension .bz2, .gz ou .lzo

```
donnees = load 'meteo.csv' using PigStorage(';') as (omm:int, date,
temperature:int);
  stations = load 'stations.csv' using PigStorage(';') as (id:int,
ville, omm:int);
```

➢ Trier les données

- Filtrons le flux de données pour ne retenir que les relevés effectués à midi précise via une expression régulière (clause matches) :

```
releve_12h = filter donnees by date matches '2012-11-15 12:00:00';
```

➢ Les jointures

- On filtre le relevé sur la date du 15 Novembre à midi et on effectue une jointure sur la liste des stations en croisant par indicatif OMM :

```
jnt = join releve_12h by omm, stations by omm;
```

➢ Trier les données

L'instruction order / by permet d'opérer un tri de manière assez similaire à son équivalent

SQL.

- on trie le résultat de la jointure par ordre croissant de température

```
tri_releve = order jnt by temperature asc;
```

- on se limite aux 5 premiers enregistrements

```
les_5 = limit tri_releve 5;
```

- on affiche le libellé des 5 villes les plus froides de France

```
villes = foreach les_5 generate ville;
```

- La commande dump permet d'afficher le contenu du flux de données dans la sortie standard.

```
dump villes;
```

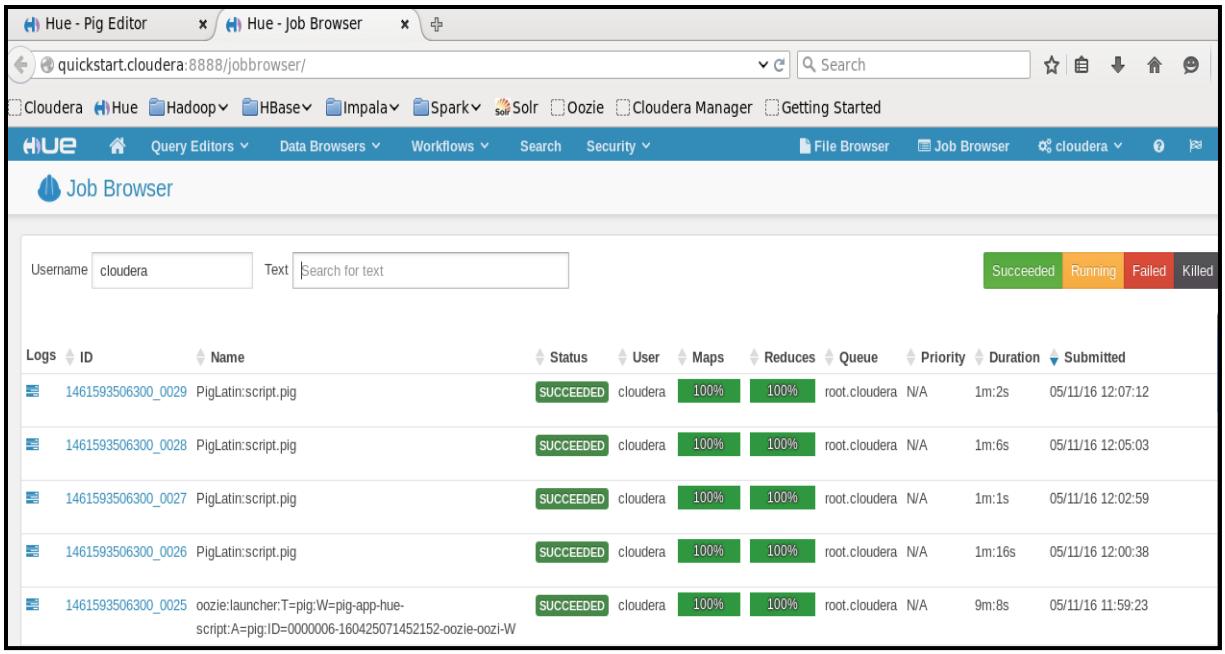


Figure IV.3 : L'état des jobs Map/Reduce

A l'exécution, on obtient le résultat suivant :

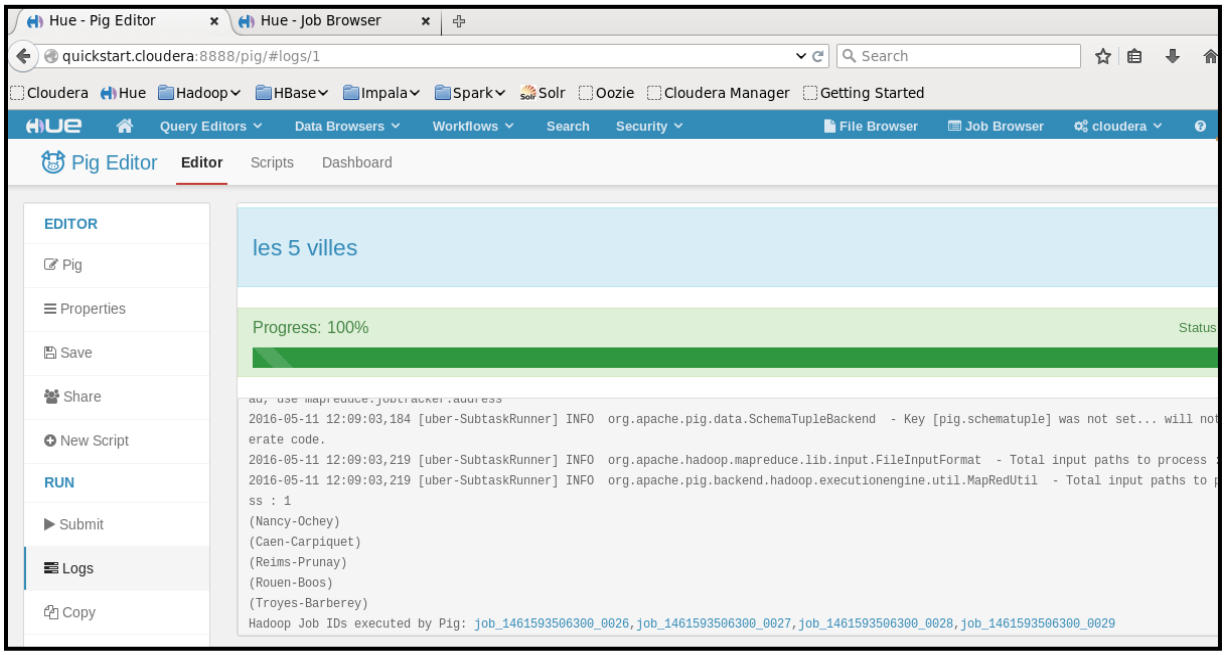


Figure IV.4: la liste des 5 villes les plus chaudes de France le 15 Novembre 2012 à midi.

Nancy-Ochey
Caen-Carpiquet
Reims-Prunay

Rouen-Boos
Troyes-Barbery

Nous pouvons avec le meme script savoir les villes plus chauds en remplaçant asc par desc. et voila le resultats:

Figure IV.5: la liste des 5 villes les plus chaudes de France le 15 Novembre 2012 à midi.

IV.4.2 Deuxième script

Calculer une moyenne : la fonction AVG

A titre d'exemple, calculons la température moyenne à midi à Reims (indicatif OMM 7072) à partir de l'échantillon de données.

```
  donnees = load 'meteo.csv' using PigStorage(';') as (omm:int, date,
temperature:int);
    reims = filter donnees by omm == 7072 and date matches '.* 12:00:00';
grp_reims = group reims by omm;
 temp_moy = foreach grp_reims generate AVG(reims.temperature)/10;
dump temp_moy;
```

voila le resultats :

Figure IV.6: Résultat de script de la température à Reims

La Moyenne de temperature = 6.871428571428571

IV.4.3. Analyse des potentiels résultats obtenus :

Les information pouvant êtres extraites à partir des données météorologiques brutes telles que la température, la force du vent ou encore la pluviométrie parmi lesquels les domaine de la climatologie, du développement durable ou encore de l'agriculture.

Ces informations peuvent, en effet, aider les agriculteurs à s'adapter aux changements climatiques qui boulversent grandement leurs connaissances ancestrales, en améliorant par exemple l'irrigation ou encore en leur permettant de choisir les cultures les plus adaptées à leurs conditions climatiques. Ils pourront ainsi savoir quels legumes ou quels fruits ils vont planter selon les conditions météorologiques et climatiques des differentes regions.

Nous pouvons même avec ce jeu de données apporter des informations nécessaires au développement d'énergies propres et renouvelables. En effet, on peut identifier les villes ou les régions il y a le plus de vent afin de planter un champs d'éolienne, ou encore de connaître le taux d'ensoleillement le plus élevé et placer des dispositif de production d'énergie solaire tels que des panneaux photovoltaiques comme cela est déjà faits dans de nombreux pays (Espagne, Qatar, etc.).

IV.5. Conclusion :

Il existe plusieurs langages et outils pour produire de l'information à partir de données stockées dans un cluster Hadoop. Java est, historiquement, le langage de programmation de

Hadoop. En effet, Hadoop a été écrit en Java et, à partir de Java, un programmeur a accès à l'ensemble de fonctionnalités de Hadoop.
Des alternatives à Java comme les scripts ont cependant été développées
Nous avons utilisé Pig qui est un outil de traitement de données qui fait partie de la suite Hadoop et qui permet l'écriture de scripts qui sont exécutés sur l'infrastructure Hadoop sans être obligé de passer par l'écriture de tâche en Java via le framework MapReduce.

CONCLUSION GENERALE

Dans ce travail nous avons présenté l'internet des objets qui pose une problématique de données , alors nous nous intéressons en premier lieu aux données capturées par ces objets, en se basant sur les techniques de transformation de ces données brutes en des informations, dans le but de prendre des décisions

Dans le premier chapitre, nous avons vu le rôle et l'apport de l'Internet des Objets dans les différents domaines (smart villes, grid, transport, domotique ...). Mais nous nous sommes retrouvés face à une problématique dans la gestion des données que l'Internet des Objets génère ; des données volumineuses, hétérogènes et qui sont produites en permanence. Les données sont la nouvelle ressource brute à exploiter. Notre objectif est l'extraction de connaissances à partir de ces données qui seront utilisées dans les processus de décision.

Le deuxième chapitre, nous nous sommes penchés sur l'étude d'un nouveau domaine celui du Big data parce que nous avons vu que les techniques actuelles de traitement des données sont limitées et inadaptées face aux données massives et de différentes natures. La mise en œuvre d'une solution de gestion des Big Data nécessite le choix : d'une méthode de stockage, d'une technologie d'exploitation et des outils d'analyse de données pour optimiser les temps de traitement sur des bases de données volumineuses

Dans le troisième chapitre, nous avons examiner Hadoop, le Framework open source qui s'appuie sur MapReduce et GFS pour le traitement analytique et distribué des données massives, on a compris qu'il n'est qu'une introduction sur toute l'architecture Hadoop et sur ses composants, on a détaillé en effet ce qu'est MapReduce, HDFS, comment gérer le matériel efficacement pour ce type d'architecture,. Après vous en avoir expliqué les bases de Hadoop, on a expliquer comment l'installer Hadoop sur une station de travail dans un mode local et un mode pseudo-distribué et nous avons expliqué théoriquement l'exemple de WordCount basé sur l'algorithme de MapReduce puis nous avons l'implémenter dans Hadoop .

Dans le quatrième chapitre, nous avons nous passé par l'algorithme MapReduce qui a valut à Hadoop le succès que l'on sait. parce que cet algorithme atteint ses limites dès qu'il s'agit d'implémenter des traitements plus complexes que le simple comptage de mots. L'écriture de fonctions Map/Reduce peut s'avérer être une opération très fastidieuse, et même répétitive puisque certains traitements comme les filtres sont très courants, c'est précisément ce que nous avons démontrer avec l'outil Pig et le script pour extraire des informations a partir des données brutes.

BIBIOGRAPHIE

[2] Corp Agency, guide du Big data l'annuaire de référence à destination des utilisateurs, guide, Paris, 2015 / 2016, 180 pages.

[3] Gilles Babinet et Al, Big data et objets connectés faire de la France un champion de la révolution numérique, article, Institut Montaigne, Avril 2015, 228 pages.

[5] Daniel Kofman, L'internet de demain Nouveaux enjeux-nouvelles problématiques, article, Institut Mines-Télécom, Juin 2015, 2 pages.

[7] Stéphane Lohier, IoT, ESIPE-MLV France, 75 pages.

[8] Manar Jaradata et Al, The Internet of Energy: Smart Sensor Networks and Big Data Management for Smart Grid, memoire, Procedia Computer Science, 2015, .p 592-597.

[9] Alicia Asín ET David Gascón, 50 Sensor Applications for a Smarter World, article, Libelium. Espagne, May 2015, 41 pages.

[15] A Zaslavsky et Al, Sensing as a Service and Big Data, article, Research School of Computer Science. Australie, 2 Janvier 2013, 8 pages

[16] Émilie Baro, Vers une définition des Big data en santé basée sur la littérature, thèse de doctorat, Université Lille 2 droit et santé Faculté de médecine Henri Warembourg, 11 mai 2015 à 16h, 69 pages.

[17] Jean-Louis Monino, Big data open data et valorisation des données, article, Réseau de Recherche sur l'Innovation. France, 2015, 17 pages.

[18] Amrane Abdesalam, Big Data Concepts et Cas d'utilisation, Rapport, CERIST centre de recherche sur l'information scientifique et technique, 2015, 12 pages.

[19] MNN Mother Nature Network, Big Data = Big Wins for the Environment, article, UPS United Parcel Service of America, 2013, 1 page.

[20] UPS 2014 Corporate Sustainability Report, Committed to More, 29 Juin 2015, guide, UPS United Parcel Service of America, 142 pages.

[22] Jean-Louis Ermine et Al, une chaîne de valeur de la connaissance, article, Management international, 7 Fevrier 2016 04 :29, p. 29-40.

[24] GFII Groupement Français de l'industrie de l'information, Big Data : exploiter de grands volumes de données : quels enjeux pour les acteurs du marché de l'information et de la connaissance ?, article, Maison de l'Europe, Paris 3 juillet 2012, 48 pages.

[25] Luc Bretones et Al, Big data l'accélération d'innovation, Livre blanc, l'institut G9+, décembre 2014, 122 pages.

[26] Rudi Bruchez, NoSQL et le Big Data Comprendre et mettre en œuvre 2 éd, 22 rue des Grands Augustins, 75006 Paris, 2015,63 pages.

[27] Laurent Jolia-Ferrier, Big Data concepts et mise en œuvre de Hadoop, livre, Fevrier 2014, editions-eni France, 207 pages

[37] Corp Agency, guide du Big data l'annuaire de référence à destination des utilisateurs, guide, Paris, 2014 / 2015,120 pages

[38] Arthur Haimovici, EBG - L'Encyclopédie des Big Data 2016, guide,2016, France 202 pages
[39] Orange, acteur de l'Internet des objets et du Big Data, article, Novembre 2015, 17 pages
[40] EBG, internet-des-objets-30-projets-concrets-livre-blanc, , guide,2016, France 146 pages

[41] Barb Edson Directrice générale Microsoft Corp, Créer l'Internet de vos objets,article, 2014 Microsoft Corporation

WEBOGRAPHIE

[1] ZDNet, http://www.zdnet.com/article/ten-examples-of-iot-and-big-data-working-well-together/, 05/12/2015.
[4] IB, http://www.ib-formation.fr/catalogue/nbs-details/catref/universib-gouvernance-informatique-etat-de-lart-etat-de-lart/ref/sem76/etat-de-lart-de-linternet-des-objets-connectes, 08/12/2015.
[6] OpenEdition Books, http://books.openedition.org/editionsmsh/84, 12/12/2015.
[10] Smart Grids-CRE, http://www.smartgrids-cre.fr/index.php?p=smartcities-caracteristiques, 20/01/2016.
[11] Smart Grids-CRE, http://www.smartgrids-cre.fr/index.php?p=smartcities-masdar , 20/01/2016.
[12] Boursorama, http://www.boursorama.com/actualites/songdo-en-coree-du-sud-exemple-parfait-de-ville-intelligente-5d9716f9ce85f1260a0d81c55ab5191c, 24/01/2016
[13] Ubiant, http://www.ubiant.com/hemis-ubiant/, 26/01/2016.
[14] CityzenSciences, http://www.cityzensciences.fr/ , 26/01/2016.
[21] objetconnecte.com, http://www.objetconnecte.com/4-organisations-combinaison-bigdata-iot-2306/, 05 /12/2015.
[23] JDN, http://www.journaldunet.com/solutions/analytics/big-data/, 12/02/2016.

[28] JDN, http://www.journaldunet.com/developpeur/outils/les-solutions-du-big data/principe-de-fonctionnement-de-mapreduce.shtml , 12/02/2016.
[29] JDN, http://www.journaldunet.com/developpeur/outils/les-solutions-du-big-data/big-data-des-solutions-open-source.shtml, 12/02/2016.
[30] Quora, https://www.quora.com/Whats-the-difference-between-big-data-and-cloud-computing, 04/05/2016.
[31] IDATE, http://www.idate.org/fr/Actualites/Cloud-Big-Data_739.html, 04/05/2016
[32] JDN, http://www.journaldunet.com/solutions/saas-logiciel/comparatif-4-distributions-hadoop/,04/05/2016
[33] Le magit, http://www.lemagit.fr/definition/Hadoop-Cluster, 18/03/2016.

[34]open souce guide, http://www.open-source-guide.com/Solutions/Developpement-et-couches-intermediaires/Big-data, 10/04/2016

[35] orange, http://www.orange-business.com/fr/blogs/cloud-computing/transformation/quand-les-big-data-deviennent-intelligents, 12/04/2016

[36] orange, http://www.orange-business.com/fr/blogs/usages-dentreprise/open-data/big-data-faites-parler-les-donnees-issues-de-l-internet-des-objets

[42]blog Cloudera, http://blog.cloudera.com/blog/2014/01/how-to-create-a-simple-hadoop-cluster-with-virtualbox/ , 15/03/2016
[43] codersvoice, http://www.codersvoice.com/a/webbase/install/10/082014/139.html, 26/03/2016
[44] Inovia Blog, http://blog.inovia-conseil.fr/?cat=27, 22/03/2016
[45] liberation, http://www.liberation.fr/futurs/2014/09/21/climat-il-pleut-des-data_1105465, 05/04/2016
[46] scidev, http://www.scidev.net/afrique-sub-saharienne/cultures/article-de-fond/de-l-utilit-des-donn-es-climatiques-pour-les-agriculteurs-s-n-galais.html, 03/03/2016
[47] pearltrees, http://www.pearltrees.com/u/57594082-differentes-distributions, 25/04/2016
[48] mtaterre, http://www.mtaterre.fr/dossier-mois/archives/chap/683/L-energie-eolienne-aujourd-hui-en-France, 14/04/2016
[49] Inovia Blog, http://blog.inovia-conseil.fr/?p=130, 22/03/2016