Gestion et optimisation de l'architecture logistique de lacs de données

Marzieh Derakhshannia

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITE DE MONTPELLIER

En Informatique

École doctorale : Information, Structures, Systèmes

Unité de recherche : LIRMM

Gestion et optimisation de l'architecture logistique de lacs de données

Présentée par Marzieh Derakhshannia

Le 1er juin 2022

Sous la direction de Anne LAURENT, Arnaud MARTIN et Hicham HAJJ HASSAN

Devant le jury composé de

Anne Laurent, Professeur, LIRMM, Université de Montpellier	Directrice
Arnaud Martin, Maître de Conférences, OSU OREME - CEFE, Université de Montpellier	Co-encadrant
Hicham Hajj-Hassan, Chercheur, CNRS Liban	Co-encadrant
Jérôme Darmont, Professeur, ERIC, Université Lyon 2	Rapporteur
Sadok BEN YAHIA, Full Professeur, Tallinn University of Technology (TuT)	Rapporteur
François Bretagnolle, Professeur, UMR Biogéosciences, Université de Bourgogne	Examinateur
Marianne Huchard, Professeur, LIRMM, Université de Montpellier	Présidente
Cédrine Madera, Distinguished IT Architect, IBM	Invitée

UNIVERSITÉ DE MONTPELLIER

"J'adresse mes plus sincères remerciements à mes parents Nahid et Hamid, mon frère Mostafa et ma chère Professeure Anne Laurent, qui m'ont accompagnée, aidée, soutenue et encouragée tout au long de la réalisation de cette thèse...."

Résumé

Le monde numérique en constante évolution donne naissance au précieux concept "*data*" que l'on appelle l'or noir. Conformément à cette évolution, les systèmes de gestion de données, qui jouent des rôles importants dans la valorisation des données générées, deviennent un élément essentiel dans les systèmes d'information et pour les processus de prise de décision. Avec la révolution digitale, les données sont générées chaque seconde en gros volumes, par de multiples sources et dans différents formats. Il est communément admis que ces données brutes peuvent être exploitées pour extraire de la valeur. L'hétérogénéité des données sources se traduit par un besoin de systèmes intégrés pour stocker, traiter et analyser efficacement des données massives et éparpillées. Le phénomène de données massives, qui est connu sous le nom de mégadonnées, exige un système décisionnel avec une architecture souple qui stocke les données hétérogènes et soutient les caractéristiques principales de mégadonnées comme le volume, la variété, la vélocité, la vitesse et la véracité. Le lac de données, qui est un système de stockage centralisé, est une bonne réponse à ces problèmes posés afin accueillir à grande échelle des données brutes sous leurs formats natifs. Par rapport à cet objectif, nous formulons l'hypothèse que l'architecture et l'infrastructure du lac de données ont un impact significatif sur la rentabilité et la fonctionnalité du système global. À cet égard, la conception et la gestion de la structure du lac de données nécessitent des méthodes pratiques et innovantes afin de réaliser un référentiel centralisé intégré et optimal. En considérant la structure systémique du lac de données ainsi que l'architecture globale des systèmes, nous travaillons sur l'analogie d'une vision logistique (chaîne d'approvisionnement) et biologique pour répondre aux objectifs définis.

La chaîne d'approvisionnement est un bon exemple de systèmes logistiques où les participants hiérarchiques se coordonnent au sein d'un réseau intégré afin de préparer un produit ou de rendre des services aux consommateurs ciblés. La structure logistique ainsi que les stratégies de gestion de la chaîne d'approvisionnement pourraient être une source d'inspiration innovante pour concevoir, gérer et optimiser un système de gestion de données basé sur une vision logistique. Pour cette raison, la mise en œuvre de la méthode analogique entre structures systémiques clarifie dans quelle mesure on pourrait tirer parti des stratégies gestionnaires dérivées de la chaîne d'approvisionnement pour développer l'architecture et les performances du lac de données.

D'autre part, le nom de lac de données vient du système naturel qui correspond à un système écologique dans lequel certains éléments et certaines fonctions (êtres vivants, informatique ADN, énergie, nutrition, ...) pourraient être vues comme des données ou des processus de transformation des entrées en sortie (consommation, décomposition, évolutions,...). Cela constitue la base de notre approche par analogie pour rapprocher ces disciplines et réfléchir à l'optimisation ou le développement de la structure du système du lac de données.

Dans cette thèse, nous émettons l'hypothèse qu'il est possible de décrire un lac de données et ses fonctionnalités en le comparant à ces deux formes de structures : d'une part la vision logistique d'une chaîne d'approvisionnement et d'autre part la vision biologique. Sur la base de ces objectifs, nous déclinons nos contributions. Dans un premier temps, nous nous intéressons à plusieurs architectures de lacs de données et vérifions l'efficacité de ces architectures sur la performance de lac de données notamment par rapport à la gouvernance des données et la qualité de service. Dans un deuxième temps, nous introduisons la chaîne d'approvisionnement, gestion de la chaîne logistique et les méthodes qui sont utilisés fréquemment pour optimiser la chaîne d'approvisionnement. De plus, nous comparons tous les éléments de ce lac de données avec les systèmes logistiques et biologiques et nous nous concentrons sur leurs points similaires afin d'utiliser les méthodes interdisciplinaires pour optimiser le lac de données. Dans un troisième temps, nous proposons une nouvelle architecture pour les lacs de données basée sur la définition de chaîne d'approvisionnement grâce au processus évolutif de modélisation des structures des lacs de données. Nous terminons ce travail en optimisant l'architecture de lac de données proposée avec des stratégies de conception de réseau de chaîne d'approvisionnement et proposons des méthodes pour résoudre le modèle d'optimisation mathématique défini.

Mots-clés : Lac de données, Chaîne d'approvisionnement, Système biologique, Gestion de la chaîne d'approvisionnement, Architecture de lac de données, Gouvernance du lac de données.

Abstract

The digital world with constantly evolution gives rise to the precious concept, "*data*" that is known as the black gold. In accordance with this evolution, database management systems, which play an important role in data valuation, are becoming an essential element of information systems and decision-making processes. With respect to the digital revolution, data is generated every second in a huge volume, by multiple sources and with different formats. Despite the fact that managing large and dispersed data is a problematic issue , we could not neglect the precious value that could potentially be gained through raw data exploration. This heterogeneity translates into the need for an integrated system to efficiently store, process and analyze the huge amount of scattered data. The phenomenon of huge data, known as big data, requires a decision-making system with an appropriate architecture that stores the heterogeneous data and supports the main characteristics of the big data environment, such as the data volume, the veracity, velocity and veracity. The data lake, which is a centralized storage system, is a good answer to these arising problems to receive raw data on a large scale in their native formats. Concerning this goal, it is clear that the infrastructure and architecture of the data lake have a significant impact on the profitability and functionality of the overall system. In this regard, the design and management of the data lake structure requires practical and innovative methods in order to achieve an integrated and optimal centralized repository. By considering the systemic structure of the data lake as well as the global architecture of the systems, we work on the analogy of a logistics (supply chain) and biological vision to meet the defined objectives.

The supply chain is a good example of logistics systems where hierarchical participants are coordinated within an integrated network in order to prepare a product or render services to targeted consumers. The logistics structure as well as the supply chain management strategies could be an innovative source of inspiration to design, manage and optimize a data management system based on a logistics vision. For this reason, the implementation of the analog method between systematic structures clarifies to what extent one could take advantage of management strategies derived from the supply chain to develop the architecture and performance of the data lake.

On the other hand, the name of data lake comes from the natural system which corresponds to an ecological system in which certain elements and certain functions (living beings, DNA computing, energy, nutrition, ...) could be seen as data or input-to-output transformation processes (consumption, decomposition, evolutions, etc.). This forms the basis of our analogy approach to bringing these disciplines together and thinking about optimizing or developing the structure of the data lake system.

In this thesis, we hypothesize that it is possible to describe a data lake and its functionalities by comparing it to these two forms of structures: on the one hand the logistics vision of a supply chain and on the other hand the biological vision. First, we are interested in relying on several data lake architectures and verifying the effectiveness of these architectures on the performance of the data lake, in particular in relation to data governance and the quality of services. In a second step, we introduce the supply chain, supply chain management and the methods that are used frequently to optimize the supply chain. Moreover, we compare all the elements of this data lake with the logistic and biological systems and focus on their similar points in order to use the interdisciplinary methods to optimize the data lake. Thirdly, we propose a new architecture for data lake based on supply chain definition thanks to the evolutionary process of data lakes structure modeling. We finish this work by optimizing the proposed data lake architecture with supply chain network design strategies and propose the methods to solve the defined optimization mathematical model.

Keywords: Data lake, Supply chain, Supply chain management, Biological system, Data lake architecture, Data lake governance

Acknowledgements

Tout d'abord, je voudrais remercier les membres de mon jury d'avoir accepté de relire mes travaux. Mon sujet de thèse n'est pas classique, et je leur suis très reconnaissante d'avoir accepté de s'y plonger malgré ce caractère original et la combinaison des disciplines qui se croisent dans ce manuscrit.

Mes remerciements s'adressent également aux rapporteurs de thèse le professeur Sadok Ben Yahia et le professeur Jérôme Darmont qui m'ont fait l'honneur de bien vouloir étudier avec attention mon travail et pour m'avoir fourni une multitude de pistes de recherche pour les prochaines études.

Je voudrais également remercier tout particulièrement ma directrice de thèse la professeure Anne Laurent qui m'a dirigée tout au long de ces trois années de thèse. Elle a toujours été disponible, à l'écoute de mes nombreuses questions, et s'est toujours intéressée à l'avancée de mes travaux. Les nombreuses discussions que nous avons eues ainsi que ses conseils sont pour beaucoup dans le résultat final de ce travail. Sa capacité d'analyse et son enthousiasme m'ont montré que le monde de la recherche pouvait être un univers passionnant. Enfin, ses nombreuses relectures et corrections de cette thèse ont été très appréciables.

Je tiens à sincèrement remercier Docteur Arnaud Martin et Docteur Hicham-Hajj Hassan d'avoir co-encadré ce travail de thèse et ils ont souvent attiré mon attention sur certains problèmes de conception des sujets interdiciplinaires et j'ai eu de la chance de pouvoir travailler avec eux.

Je remercie également Dr Dickson Owuor qui m'a aidée pour la programmation des méthodes méta-heuristiques. Ses connaissances m'ont permis de progresser et ont répondu à plusieurs de mes préoccupations.

Table des matières

Table des figures

Liste des tableaux

Acronymes

GCL	**G**estion **C**haîne **L**ogistique
CPU	**C**entral **P**rocessing **U**nit
CL	**C**haîne **L**ogistique
ALLD	**A**rchitectur **L**ogistique de **L**ac de **D**onnées
GCVD	**G**estion du **C**ycle de **V**ie des **D**onnées
GECA	**G**estion **E**nvironnementale de la **C**haîne d'**A**pprovisionnement
ECV	**E**valuation de **C**ycle de **V**ie
IoT	**I**nternet **o**f **T**hings
ALLD	**A**rchitecture **L**ogistique du **L**ac de **D**onnées
GA	**G**enetic **A**lgorithm

Chapitre 1

Introduction

"Les données sont la nouvelle science. Le Big data détient les réponses"

– Patrick Paul Gelsinger

1.1 Introduction

Des quantités extraordinaires de données de formats variés sont générées en permanence dans des environnements de type « Big data ». Au niveau mondial plus de 60 zettaoctets de données ont été créés en 2020. D'après SAS [1], ***"Le big data s'adresse aux données qui sont énormément massives, rapides ou compliquées ce qui rend difficile voire impossible leur stockage et traitement avec des méthodes traditionnelles "***. La révolution technologique comme les objets connectés, Internet, la télé-médecine, les prévisions météorologiques, etc. augmentent le volume et l'hétérogénéité de données. En effet, les défis majeurs du big data ne sont pas seulement illustrés par la quantité extraordinaire des données massives, mais également représentés par le stockage, l'organisation et le traitement de ces données en grande quantité afin d'accéder aux informations qui soutiennent le processus de décision. De plus, la multiplication des types de données, la maintenance de données disparates et distribuées, la validité des informations manipulées, la facilité d'accès et de visualisation, les problèmes de sécurité, la qualité de service (comme le temps de réponse acceptable), et l'orchestration des données FAIR [Wilkinson et al., 2016] (données faciles à trouver, accessibles, interopérables et réutilisables), deviennent les grands défis pour toute personne concernée par les enjeux de données massives.

Depuis quelques années les données sont considérées comme l'or numérique pour les organisations et les entreprises qui en sont les propriétaires. De plus, stocker et tirer parti de cet atout précieux devient une préoccupation majeure pour ces organisations. En outre, la mise en place d'infrastructures efficaces permettant d'exploiter de manière efficiente cette typologie de données, représente un gage de compétitivité pour les entreprises les plus innovantes. Au cours de la dernière décennie, l'évolution des technologies de stockage et de traitement de données a joué un rôle important dans la gestion de données massive et l'atteinte des objectifs visant le développement des architectures opérationnelles pour sauvegarder et sécuriser les données importantes au sein d'une organisation. Par conséquent, de nombreuses organisations ont décidé d'améliorer leurs plates-formes de stockage de données afin de gérer des données massives qui sont produites chaque seconde par des sources diverses et variées [Llave, 2018]. Pour cette raison, une révolution est apparue dans les approches de gestion des données qui influencent les processus de maintenance et d'accès aux données hétérogènes ainsi que l'amélioration des systèmes d'information complexes.

En 1960, le terme "Base de données" a émergé. Dans les années 70, 80 et 90, les bases de données reposaient principalement sur des modèles relationnels et des langages tels que SQL. Les bases de données NoSQL, apparues plus récemment après les bases objet et relationnelles-objet, prennent en charge de manière plus aisée une grande quantité de données générées par plusieurs sources [Berg et al., 2012]. Récemment, l'informatique en nuage [2] est considérée comme un moyen novateur de stockage et de gestion des données qui innove et facilite la valorisation

[1] https://www.sas.com/en_us/insights/big-data/what-is-big-data.html
[2] Cloud Computing

des données, de manière rapide, économique, et sécurisée par rapport aux anciennes technologies [Mell and Grance, 2011]. L'infrastructure flexible et évolutive de l'environnement *Cloud* contient des machines virtuelles isolées qui permettent de partager leurs ressources sur un serveur physique avec un coût d'opération faible par rapport à l'ancienne infrastructure [Zhang et al., 2020]. Alors que nous pouvons observer l'émergence de plusieurs solutions qui ont rempli le gap technologique auparavant observé dans ce secteur, les organisations restent tout de même confrontées à des difficultés liés au choix et au lancement d'une infrastructure efficace et rentable pour le traitement et le stockage de données volumineuses. La solution choisie devra être adaptée aux besoins de l'organisation et fournir un environnement de travail complet, efficace et efficient.

En conséquence, les systèmes de bases de données centralisés tels que les entrepôts de données et ses sous-ensembles tels que les magasins (Datamarts) de données et, d'autre part, les systèmes de gestion de bases de données (SGBD) tels que PostgreSQL, sont mis en évidence afin de répondre aux besoins de stockage et de la manipulation de la quantité volumineuse de données structurées à l'ère du big data. Cependant, les limites des entrepôts de données classiques telles que le manque de compatibilité pour accueillir des données non structurées ou semi-structurées, une faible vitesse de chargement, sans oublier également les coûts d'investissements nécessaires au lancement d'une telle infrastructure, obligent les fournisseurs de services à trouver une solution plus optimisée et dynamique pour couvrir les lacunes et pallier les problèmes cités. De plus, de nombreux systèmes génèrent maintenant des données que l'on souhaite conserver sans savoir a priori à quoi et quand elles vont servir, ce qui constitue une très forte différence par rapport aux systèmes classiques opérationnels et décisionnels.

Sur la base de ces besoins, une nouvelle génération de systèmes de stockage de données est en cours de développement pour prendre en charge au moins des péta-octets de données dans leurs formats bruts provenant de sources dispersées. On parle de *Lac de données* qui est un système de stockage de données flexible et agile pour l'orchestration des données structurées et non-structurées d'une manière efficiente pour les organisations manipulant cette typologie de données. Un lac de données est un système de stockage centralisé qui a été créé pour stocker toutes les données difficiles à gérer avec les outils traditionnels. Un lac de données fonctionne en intégrant dans son écosystème une multitude de technologies, qui opèrent de manière harmonieuse et qui rendent possible la gestion des données massives [Fang, 2015, Lo Giudice et al., 2018, Gorelik, 2019].

Malgré les capacités du lac de données à intégrer des données hétérogènes, la sécurité et la qualité des données sont des problématiques à pallier. La gestion de ce système de stockage de données peut apporter des risques de submersion par les données de mauvaise qualité. Ce phénomène est appelé «*marécage de données*[3]». Pour éviter que le lac de données ne devienne un “marécage" de données inutiles, des stratégies et des règles sont mises en œuvre dans le déploiement pour contrôler la véracité et la validité des données hétérogènes. En se basant sur ces exigences, la gouvernance et le fonctionnement du lac de données sont deux priorités incontournables qui devront être prises en compte à travers une architecture organisée et systémique. Ces dernières années, plusieurs études ont été menées pour proposer une architecture capable de répondre aux problématiques de gouvernance des données et d'efficacité du fonctionnement de lac de données dans un environnement favorable et performant. Cependant, un équilibre entre la qualité et la sécurité de données d'une part et les coûts de lancement de l'architecture d'autre

[3]Data swamp

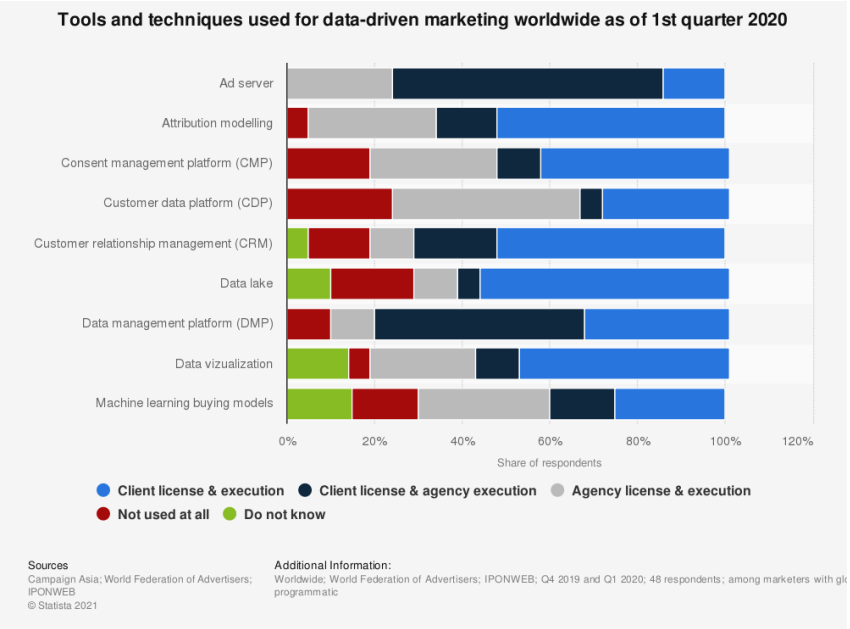


FIGURE 1.1 – Mise en évidence des outils et techniques utilisés pour la commercialisation basé sur les données dans le monde au 1er trimestre 2020

part, exige une architecture optimisée qui stocke, traite et analyse les données massives de la meilleure façon et au moindre coût possible.

Le nombre d'études sur la mise en œuvre et la commercialisation des lacs de données est en forte croissance durant ces dernières années. Les fournisseurs de services et le monde académique dans ce domaine s'efforcent de trouver les bonnes méthodes, stratégies, infrastructures et technologies pour déployer un lac de données rentable. La figure stastique 1.1 montre que le lac de données est devenu un outil basé sur les données connues au premier semestre de l'année 2020 [4]. En se basant sur ces informations, nous pouvons avancer que l'utilisation du lac de données en tant que système de gestion de l'information centralisée connaîtra sans doute une progression dans les grandes entreprises et organisations. C'est déjà le cas pour certaines d'entre elles.

L'origine de l'ontologie du lac de données est décrite avec le nom ***'Lac'***[5] proposé par James Dixon en 2010 qui est dérivé de la définition du lac naturel où tous les types de données comme les espèces vivantes sont accumulées dans un environnement centralisé. De plus, les performances du lac de données, sa structure mais aussi sa raison d'existence, nous rappellent la structure ***systémique*** qui contient des entrées, des sorties, et plusieurs éléments et composants intégrés vers un objectif global. Le lac de données affiche plusieurs caractéristiques qui permettent d'employer des stratégies synthétiques ou multidisciplinaires qui sont souvent utilisées dans d'autres structures systémiques. La vision systémique du lac de données solutionnera les problématiques importants liés à la fonctionnalité, aux performances, à l'optimisation et à la gestion du système centralisé de stockage de données massives.

Pour cette raisons, nous avons pensé les deux systèmes efficaces, systèmes logistique et éco-

[4] https://www.statista.com/statistics/264163/tools-for-data-analysis-used-worldwide/
[5] https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/

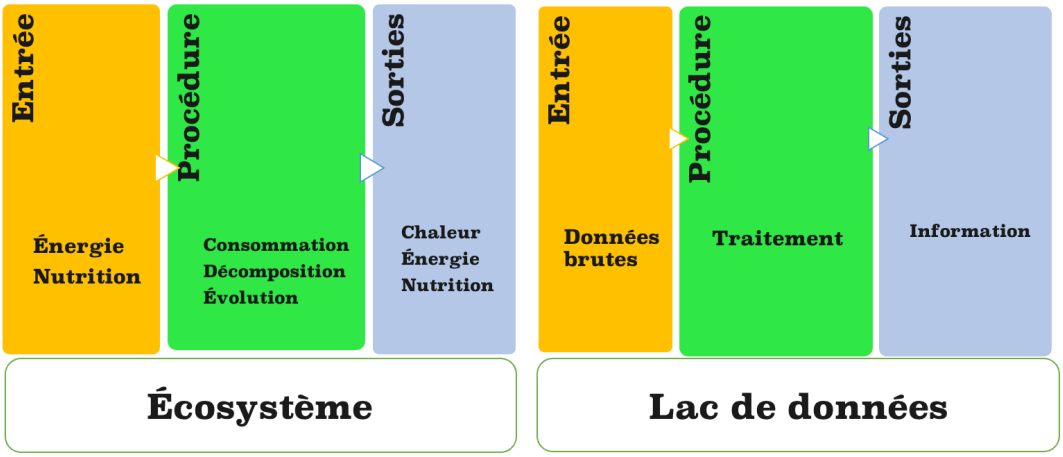


FIGURE 1.2 – Lac de données comme un écosystème

système, comme des sources d'inspiration de méthodes multidisciplinaires pour réfléchir à la structure et les performances des lacs de données. Dans cette perspective, nous avons étudié dans nos travaux deux systèmes associés à la gestion du lac de données : d'une part des méthodes bio-inspirées des systèmes naturels et d'autre part des cadres inspirés des systèmes logistiques.

Dans un premier temps, nous nous appuyons sur l'origine de nom du lac de données qui vient du système naturel [Miloslavskaya and Tolstoy, 2016, Khine and Wang, 2018].Les systèmes naturels et leur fonctionnalité pour gérer ou faire évoluer leurs composants sont normalement les sources de méthodes mimiques pour plusieurs domaines comme les domaines organisationnels, les domaines de l'ingénierie ou les domaines de l'informatique. Dans cette étude, nous parlons des similitudes entre la définition du lac de données avec l'écosystème où les entrées du système écologique (énergie, nutrition, ...) sont considérées comme des données, les processus de transformation des entrées en sortie (consommation, décomposition, évolutions,...) comme le traitement des données, et les sorties (chaleur, énergie et nutrition) comme les informations (données traitées) dans le lac de données [Tylianakis et al., 2008, Kane, 1997]. Sur la base de ces similitudes, on peut conclure que les stratégies de gestion utilisées par l'écosystème pourraient être une grande ressource de méthodes interdisciplinaires pour l'optimisation ou le développement de la structure du système du lac de données. La figure 1.2 décrit le lac de données en tant que système écologique.

Dans un deuxième temps, on parle du but de l'objectif du lac de données pour préparer et fournir les données en tant que produit précieux aux utilisateurs finaux. Par conséquent, on s'appuie sur la structure logistique du lac de données, qui est comparable à la chaîne d'approvisionnement. La chaîne d'approvisionnement est un système qui est utilisé par les organisations pour la fourniture et la logistique de produits ou de services aux clients et aux consommateurs [Chow and Heaver, 2007]. La pensée systémique, qui est la base de la définition de la chaîne d'approvisionnement, est mise en valeur pour l'étendre dans un domaine multidisciplinaire. Les outils et les stratégies utilisés pour optimiser la performance de la chaîne d'approvisionnement peuvent être également pratiques pour améliorer les opérations de tous les phénomènes qui fonctionnent comme un système, par exemple les systèmes de stockage de données. Par conséquent, nous allons nous concentrer sur les nouvelles méthodes et stratégies pour étudier l'architecture du

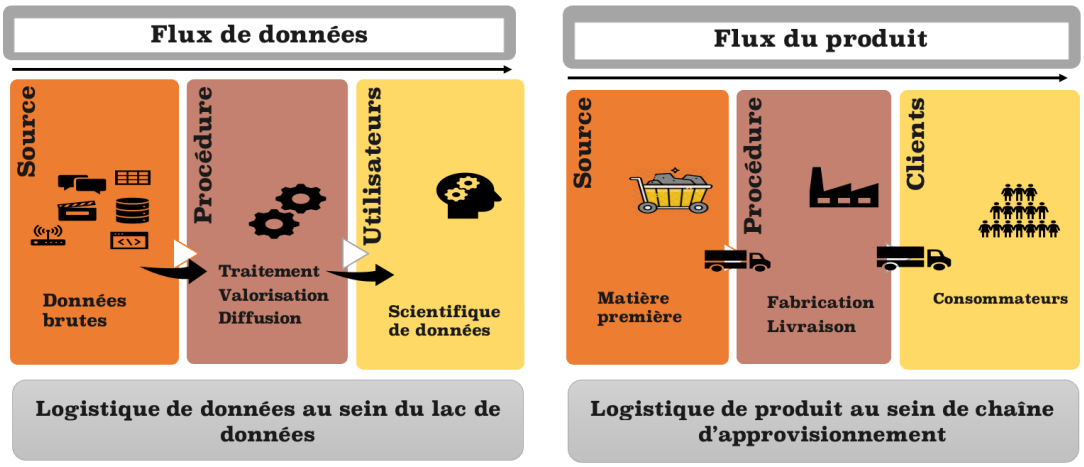


FIGURE 1.3 – Lac de données comme un système logistique

lac de données qui sont elles-mêmes basées sur les principes des systèmes logistiques. L'objectif de cette étude est de trouver un compromis entre la qualité, la sécurité et les performances du lac de données. Ces critères contribuent de manière significative à l'augmentation de la rentabilité d'un tel système de stockage. La figure 1.3 montre la représentation du lac de données en tant que système logistique.

1.2 Motivation et objectifs

Dans ce qui suit, nous expliquons pourquoi nous définissons la structure des lacs de données en fonction de structures systémiques comme système logistique et écosystème et quels sont les objectifs principaux de cette nouvelle approche dans la gestion des lacs de données.

1.2.1 Motivation

Comme nous l'avons indiqué, le lac de données est considéré comme un nouveau composant du système d'information qui a été créé pour stocker et sauvegarder des données massives sans schéma défini a priori. En effet, l'architecture de ce système de stockage de données pourrait influencer la performance et l'adaptation aux attentes des bénéficiaires. D'autre part, la qualité et la véracité des données stockées sont les autres enjeux qui nécessitent de nouveaux outils et stratégies pour la gestion du cycle de vie des données.

Jusqu'à aujourd'hui, les lacs de données ont été considérés par de nombreuses entreprises. Alors que différentes architectures, technologies, plate-formes, et les stratégies sont proposées pour augmenter la fonctionnalité et la rentabilité du lac de données, les recherches menées sur le lac de données du point de vue de la structure systémique pour l'optimisation des performances de ce système et la gestion des éléments intégrés sont naissantes. Ce sujet nécessite une vision systémique qui considère le lac de données comme une chaîne d'approvisionnement ou un système

écologique qui fournit, prépare et distribue des données en tant que produit ou espèces vivants pour les attends ou consommations finaux. La pensée logistique considère les données comme un produit commercial et énormément précieux qui coule dans les différents stages de système de stockage pendant son cycle de vie. D'autre part, la pensée biologique considère les données comme une espèce vivante qui naît et évolue selon des lois et des principes naturels tout au long de son existence. La définition d'un lac de données comme un système logistique tel qu'une chaîne d'approvisionnement ou un système naturel permet de fournir les stratégies et les disciplines managériales de ces systèmes pour la gestion du système complexe de stockage de données.

Nos travaux s'inscrivent dans la compréhension des évolutions de l'architecture du lac de données pour mieux comparer la fonctionnalité du lac de données et de ses composants sous différentes structures proposées et apporter un point de vue académique. Une architecture optimisée qui garantit la qualité et la véracité des données, protège la sécurité essentielle et augmente le niveau de service, représente un grand défi dans les domaines scientifiques et industriels. Notre travail propose des méthodes pluridisciplinaires basées sur l'analogie des systèmes logistiques et naturelles pour optimiser les performances des lac de données. De plus, cette étude pourrait déclencher des visions complémentaires pour des travaux futurs afin de mettre en évidence les avantages de la pensée analogiques dans les systèmes informatiques.

1.2.2 Objectifs

Les lacs de données deviennent le phénomène favorable pour les organisations qui envisagent des systèmes d'information adaptés aux attentes des décideurs organisationnels influents. Cependant les composants des lacs de données et l'architecture rentable et efficace pour lancer ce système de stockage centralisé sont des sujets problématiques qui n'ont pas été abordés ou résolus à certains points.

D'un point de vue académique, une structure efficace de lac de données devrait :

- Prendre en charge des données brutes provenant de plusieurs sources dans la phase de l'ingestion de données ;
- Garantir la qualité et la sécurité des données en mettant en place une gouvernance des données ;
- Réduire les coûts de mise en œuvre et de gestion des lacs de données ;
- Répondre de manière pertinente et en temps minimal aux demandes des utilisateurs ;
- Offrir un plateforme conviviale, agile et résiliente.

Un système logistique comme une chaîne d'approvisionnement doit être compatible avec les stratégies de coordination des participants, dans la mesure où elles augmentent la rentabilité du réseau de chaînes tout en minimisant les coûts totaux. Par conséquent, une conception résiliente et agile ainsi qu'une conception optimisée du réseau de la chaîne d'approvisionnement est un objectif indispensable pour les gestionnaires en amont et en aval. Une architecture équitable de chaîne d'approvisionnement pourrait influencer directement les relations entre les membres, les

nombres optimaux de composants et leurs stationnements, la résistance de la chaîne, les processus d'évaluation de cycle de vie de produit, la rentabilité et aussi la qualité de service des clients.

Sur la base de l'analogie des systèmes logistiques et des systèmes de stockage de données et en mettant l'accent sur les points communs, nous constatons que la résilience, la qualité de service, l'agilité, la sécurité et la gestion du cycle de vie des produits sont les défis stratégiques pour les deux systèmes. En effet, selon les objectifs définis et ainsi que les enjeux actuels des systèmes de stockage de données, une nouvelle architecture basée sur les nouvelles caractéristiques et disciplines est nécessaire afin de prendre des décisions à long terme **(stratégiques)** et à court terme **(tactiques et opérationnelles)** au sein des lacs de données.

D'autre part, la cohérence naturelle entre les composantes écologiques est le résultat de stratégies inhérentes du écosystème telles que l'évolution des espèces, la compétition, le parasitisme, le mutualisme et la prédation, qui tente d'établir un équilibre entre le taux d'entrées et de sorties et de faire évoluer le système global. Les stratégies naturelles ont également toujours été à l'origine de méthodes analogiques réussies dans des domaines interdisciplinaires. Pour cette raison, nous essayons de tirer parti des stratégies écologiques efficaces comme outils de gestion des systèmes de stockage de données.

Pour ce faire, notre objectif est donc :

Étudier les systèmes de stockage de données, en particulier le lac de données, analyser différentes architectures de lac de données et leurs performances en matière de gouvernance des données, d'efficacité et de qualité du service client, pallier les avantages et les inconvénients des anciennes architectures avec une nouvelle architecture logistique ainsi que proposer des méthodes et des stratégies interdisciplinaires de la gestion du cycle de vie des données.

Le travail présenté dans cette thèse reposait sur quatre pôles importants tels que :

- Comparer les architectures de lac de données avec les structures systémique connues ;
- Étudier le positionnement de ses composants essentiels et leurs objectifs ;
- Proposer les méthodes principales et mimétiques de gestion de données ;
- Définir les critères importants pour l'augmentation et l'optimisation des performances du lac de données dans le système d'information.

Ces objectifs fixés nous permettront de proposer une nouvelle architecture pour les lacs de données, concevoir cette architecture en tirant parti des bénéfices des structures de réseaux de la structure systémique notamment la chaîne d'approvisionnement, d'amorcer une approche conceptuelle du lac de données basée sur le système logistique, d'employer les stratégies et les outils de la gestion de la chaîne d'approvisionnement et écosystème pour l'optimisation, la gouvernance et la gestion de lac de données et aussi de mettre en place des méthodes issues des approches multidisciplinaires qui peuvent engendrer une implémentation physique efficace du lac de données.

1.3 Organisation du mémoire et contributions

Cette thèse est décomposée en huit chapitres.

Le chapitre 1 est consacré à l'introduction de nos travaux, motivations et objectifs.

Dans le Chapitre 2, nous étudions la définition de lacs de données, son architecture, ses composants et aussi ses fonctionnalités par rapport d'autres base de données. Par suite, nous suivons le cycle de la vie de données dans lac de données et marquons les points forts d'emploi de lac de données dans les organisation et les entreprises.

Le chapitre 3 présente en détails la chaîne d'approvisionnement comme un système logistique. Nous exposons les composants importants des réseaux de la chaîne d'approvisionnement, leurs objectifs et leurs tâches en coordonnant les participants de la chaîne. Ce chapitre a pour but de faire connaître de la chaîne d'approvisionnement, les processus de la prise de décisions au sein de la chaîne et aussi les outils d'optimisation de la performance.

Le Chapitre 4 est consacré à l'analogie du lac de données avec d'une part les systèmes logistiques en s'appuyant sur la chaîne d'approvisionnement et avec d'autre part le système biologique de lac naturel. Dans ce chapitre, nous démontrons les points communs entre les systèmes logistiques, les systèmes naturels et les systèmes de stockage de données et définissions la nouvelle architecture de lac de données en nous basant sur ces similarités. Dans ce chapitre les contributions sont :

- Comparaison les éléments importants entre les lacs de données avec la chaîne d'approvisionnement et système naturel ;
- Étude des opportunités de ces points communs vers les architectures mimétiques de lac de données ;
- Propositions les stratégies interdisciplinaires de la gestion pour la gestion et optimisation de lac de données.

Dans le chapitre 5, nous proposons une nouvelle architecture du lac de données logistique (ALLD) basée sur l'analogie du chapitre 4. Dans ce chapitre les contributions sont :

- Proposition d'une approche évolutive pour l'urbanisation de l'architecture de lac de données ;
- Étude de l'état de l'art de notre méthode de la modélisation ;
- Réalisation de quatre phases de modélisation de la structure du lac de données (conceptuelle, logique, technique et optimale).

Le chapitre 6 est consacré aux optimisations de lac de données. Nous étudions d'abord les enjeux relatifs aux coûts principaux de conception et d'implémentation d'un lac de données. Nous démontrerons les capacités de notre architecture en matière d'amélioration de fonctionnalité de

lac de donnée. En outre, nous proposons un modèle mathématique basé sur la nouvelle architecture d'un lac de données et les stratégies de prise de décision dans la chaîne d'approvisionnement afin d'optimiser le lac de données. Dans ce chapitre les contributions sont :

- Diagnostic des enjeux importants de l'architecture logistique de lac de données proposée au chapitre 5 par rapport des coûts et des bénéfices de la fonctionnalité du lac de données ;
- Introduction des stratégies conception de réseaux de chaîne d'approvisionnement optimisée comme une méthode mimétique d'optimisation du lac de données ;
- Modélisation du modèle mathématique du lac de données basé sur principes du problème de prise de décision hybride de conception de réseaux logistique comme localisation-allocation

Dans le chapitre 7, nous proposons la méthodologie de résolution du modèle mathématique d'optimisation du lac de données. Dans ce chapitre les contributions sont :

- Revue des méthodes de résolution des problèmes d'optimisation ;
- Réalisation d'un modèle exact pour résoudre le modèle mathématique du chapitre 6 ;
- Proposition de deux méthodes méta-heuristiques pour trouver les valeurs optimales du modèle d'optimisation proposé dans le chapitre 6.

Pour conclure, le Chapitre 8 est consacré aux conclusions, limites du travail et perspectives d'études futures.

Chapitre 2

Lac de données

"Pour atteindre la vérité, il faut une fois dans la vie se défaire de toutes les opinions qu'on a reçues, et reconstruire de nouveau tout le système de ses connaissances."

– René Descartes 1596-1650

Le lac de données est devenu un phénomène très étudié dans l'environnement du Big Data. A cet égard, l'exigence de ce système centralisé qui accueille tous les types de données, ses avantages par rapport aux autres bases de données, et la mise en place de sa structure optimale, sont débattues. Dans ce chapitre, nous nous appuierons sur la définition et la raison de création du lac de données ainsi que les études menées sur les architectures de ce système de stockage de données massives. Les enjeux suivants seront abordés dans ce chapitre :

- La définition du lac de données
- Lac de données versus entrepôt de données (les avantages et les inconvénients)
- L'architecture et l'environnement du lac de données
- Les enjeux du lac de données
- Implémentation du lac de données

2.1 La définition du lac de données

Le nom du lac de données est inspiré par le lac naturel [1] où tous les organismes et les espèces biologiques ou matériels non-biologiques sont autorisés à y entrer. De la même manière, le lac de données prépare un environnement pour livrer tous types de données (structurées et non-structurées) depuis diverses sources.

Selon [Fang, 2015] qui a été l'une des premières personnes à définir les lacs de données, le lac de données est la nouvelle génération des systèmes de stockage de données qui changerait les façons dont on gère les données massives. Il définit le lac de données comme suit :

" Un "lac de données" est une méthodologie chargée par un dépôt de données massives basé sur les technologies moins coûteuses qui améliore la capture, le raffinement, l'archivage et l'exploration de données brutes au sein des entreprises." [Fang, 2015]

En développant des systèmes de "Big data", le volume de données non-structurées produites par les entreprises, les objets connectés, les réseaux sociaux et les capteurs dédiés à récolter des informations spécifiques, a fortement augmenté. Le lac de données améliore les moyens de chargement, de stockage et de traitement des données non structurées par rapport aux anciens outils en mettant en œuvre des stratégies pour accueillir cette typologie de données ainsi que des données structurées. En effet, cette avancée technologique a permis de réduire le temps et les coûts liés à la préparation et au traitement de données. De plus, le lac de données en tant

[1] https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/

que composant des systèmes d'information supporte la procédure de prise de décisions à l'ère de l'explosion des données.

[Pradeep, 2015] a remarqué que le lac de données a une structure flexible qui lui procure un avantage important par rapport à d'autres entrepôts de données. Selon lui, un lac de données possède les caractéristiques suivantes :

- Le lac de données est un grand dépôt de données qui stocke une grande quantité de n'importe quel type de données en ses formats bruts, non pas pour les usages immédiats mais pour les demandes futures ;
- La fonctionnalité du lac de données pourrait augmenter grâce aux bases de données complémentaires comme les entrepôts de données ;
- Il soutient l'approche " enregistrer tout [2] pour accueillir les données massives " ;
- Le lac de données aide à créer des modèles de données flexibles et peut supporter le traitement par lots ;
- Dans le lac de données, les données sont accueillies comme elles sont, donc elles ne sont pas classifiées. Cette propriété permet une réduction significative des coûts, du temps de préparation, de la classification, du raffinement et de la transformation.

A la suite de l'émergence du *Big data*, les entreprises ont été confrontées à plusieurs défis tels que les moyens et les coûts de chargement, de stockage, de traitement et d'analyse d'énormes quantités de données, en transformant les données brutes en connaissances précieuses et exploitables pour les organisations. L'exigence en termes de données pour les entreprises modernes, souligne l'importance de déployer des infrastructures pour sauvegarder les données et les rendre accessibles, raisonnables, et visibles pour les utilisateurs. Pour cette raison, [Tomcy, 2017] s'est concentré sur les fonctionnalités du lac de données aux seins des entreprises. Il considère le lac de données comme un vaste référentiel centralisé qui collecte les données depuis plusieurs sources en formats natifs, afin d'en tirer des informations pertinentes pour l'entreprise grâce aux divers algorithmes d'analyse et d'apprentissage automatique.

Récemment, les lacs de données sont considérés comme un nouveau composant du système d'information afin d'améliorer les procédures de prise de décision pour les organisations et valoriser leur données. En accord avec la nouvelle architecture des systèmes d'information qui intègrent le lac de données, [Madera, 2018] indique que le lac de données est un système de stockage qui devient un composant complémentaire du système d'information et développe le système décisionnel. Les lacs de données sont présentés comme une *collection de données* générées afin de répondre aux besoins de l'exploration de l'ensemble de données non-transformées et sans schéma pré-défini, en vue de valoriser des données au sein des organisations. Compte tenu de l'ingestion de l'ensemble de données de formats non contraints dans un lac de données, les utilisateurs actuels des lacs de données sont principalement des experts en sciences des données. Même si de nombreux travaux soulignent l'importance de faire évoluer l'accès aux lacs par des non experts, les utilisateurs actuels font appel à cette technologie dans le but de mener des démarches efficaces et pertinentes pour découvrir acquérir de nouvelles connaissances.

[2] Save-all

En se basant sur cet état de connaissances, les caractéristiques d'un lac de données sont résumées comme suit :

- Un lac de données est un vaste zone de stockage de données brutes hétérogènes ;
- Un lac de données permet aux données d'être saisi en n'importe quel format (structurés, semi-structurés et non-structurés) depuis des sources variées (Les bases de données, Logs, données du capteur, Mobile APP, Cloud, objets connectés, Streaming data, ...) ;
- Lac de données stocke, traite, explore, et analyse les données avec des technologies à faibles coûts ;
- Un lac de données ne définit pas les schémas pré-établis pour accueillir les données mais il définit la structure des données au moment de leur utilisation qu'il est souvent appelé *Schema-on-read* ou *Late binding* [Fang, 2015] ;
- Les stratégies du lac de données peuvent combiner des approches de base de données SQL et NoSQL et des capacités de traitement analytique en ligne (OLAP) et de traitement de transaction en ligne (OLTP) [Miloslavskaya and Tolstoy, 2016] ;
- Le lac de données utilise l'index de sources de métadonnées afin d'augmenter et assurer la qualités de données ;
- Un lac de données est gouverné par les règles, et processus qui sont appelés les gouvernances de données. Cela permet de garantir la véracité et la sécurité de données ;
- La complexité du Lac de données l'a rendu accessible uniquement aux spécialistes en sciences des données.

En nous basant uniquement sur la définition du Lac de données et ses caractéristiques particulières, nous pouvons constater des lacunes et des questions importantes qui ne trouvent pas de réponses. Ces questions traitent des avantages du lac de données et ses capacités à stocker et traiter les données, par rapport à d'autres solutions comme les entrepôts de données. Par conséquent, Il nous semble important de bien préciser, au travers la section suivante les différences entre les lac de données et son complément, l'entrepôt de données, à l'égard de la gestion de données en quantité massives.

2.2 Lac de données et entrepôt de données

Avec l'avènement du concept du lac de données, on pensait que ce dernier pourrait remplacer l'entrepôt de données, parce qu'il ingère tous types de données et propose une plate-forme plus rentable en vue de découvrir des nouvelles pistes d'information à exploiter. Pourtant, les enjeux controversés qui sont liés à l'architecture et les objectifs du lac de données exigent un système de stockage complémentaire équilibrant les avantages et les inconvénients du lac de données. Dans cette section nous détaillerons les points comparatifs entre les lacs de données et l'entrepôt de données afin qu'on puisse distinguer ces deux systèmes de stockage de données et démontrer comment on pourrait tirer parti des points forts de chacun. Nous comparons les lacs de données

et les entrepôts de données en prenant en considération plusieurs critères qui nous permettront de mettre en exergue les différentes phases de cycle de vie de données dans les systèmes de stockage. Dans la figure 2.1 nous présentons une comparaison entre les deux systèmes de stockage de données.

La figure 2.1 résume l'analogie entre le Lac de données comme un système de stockage plutôt moderne et l'entrepôt de données comme un système de stockage plutôt accessible et adapté au marché. Cette comparaison adresse les phases importantes de cycle de vie de données dans les systèmes de stockage, tels que la phase d'ingestion, de stockage, du traitement et d'exploitation de données. Du point de vue de la phase *d'ingestion*, le lac de données permet à tous les types de données depuis les différentes sources d'entrer dans le lac sans aucune préparation. En opposition, les entrepôts de données n'autorisent que les données structurées avec des schémas prédéfinis depuis des sources opérationnelles.

Le chargement des données hétérogènes représente un risque pour le lac de données. En effet, le système pourrait se transformer en marécage de données ou être rempli d'une énorme quantité de données inutiles et non identifiables [Madera, 2018]. Par conséquent, la gestion des métadonnées et la gouvernance de données sont indispensables pour protéger la fiabilité du lac de données et la véracité de données stockées, tandis que l'entrepôt de données grâce à l'hébergement de données structurées, est moins risqué [Sawadogo et al., 2019a]. Par rapport aux méthodes de chargement de données, le lac de données profite de la stratégie de transformation de données au moment de l'utilisation qui s'appelle *late binding* en anglais. Par contre, l'entrepôt de données a besoin de transformer les schémas de données avant le chargement afin d'éviter de saisir des données incompatibles avec la structure définie [Fang, 2015].

Dans la phase de ***stockage***, les données sont stockées avec des coûts faibles à travers de lac de données par rapport à l'entrepôt de données et cette caractéristique est un grand avantage pour le lac de données. Dans le lac de données, selon les conditions requises, le système de stockage de données est construit par une variété d'outils de stockage comme Hadoop, NoSQL, et les bases de données relationnelles et cela permet d'accueillir n'importe quel type de donnée dans un environnement centralisé et unique, et extraire des informations plus rapidement que d'autres systèmes de stockage. Grâce à ces techniques, toutes les données sont ingérées dans un système intégré qui n'impose aucune limitation pour les requêtes demandées pour les utilisateurs et valorisent les facultés des données [LaPlante, 2016]. Cependant, les données dans les entrepôts de données sont plus contraintes par la modélisation et les systèmes de stockage pré-établis. En revanche, ces fortes restrictions garantissent une sécurité optimale du système de stockage de l'entrepôt de données, contrairement au lac de données qui pourrait être confronté au risque de submersion par les données ingérables.

Le *traitement* de données est effectué par les technologies de programmation avancées comme MapReduce ou Spark. Ces plateformes supportent tous les deux types de traitements tels que le traitement en temps réel [3] et le traitement par lots [4]. Par conséquent, le lac de données prépare un environnement favorable pour traiter en temps réel les données qui sont chargées depuis le web ou les réseaux sociaux. Par contre, l'architecture *schema-on- write* d'entrepôt de données encadre les données par les modélisations complexes et précises. Cette architecture exige un processus ETL (Extract, Transform, Load) afin de vérifier les structures et la sécurité de données et les

[3] Real-time processing
[4] Batch processing

Phase de données	Métriques	Lac de données	Entrepôts de données
Ingestion	**Sources de données**	**Tous**	**Externe / opérationnelle**
	Types de données	structurées / traitées / non structurées / semi-structurées / brutes	Structurées /Traitées
	Métadonnées	Oui	Optionel
	Gouvernance de données	Nécessite une approche axée sur les métadonnées	sécurité des données facile à contrôler
	Vitesse de chargement	Rapide	Lent
	Transformation	Transformé lorsque les données sont prêtes à être utilisé.	Transformées avant le chargement des données
	Méthode de chargement	*Late binding*	*Early binding*
Stockage	**Coûte de stokage**	À faible coût	Coût modéré
	Système de stockage	Hadoop,SQL,NoSQL,Relationelle	SQL/Relationelle
	Modelisation	A la volée	Etoil ou flacon
	Volume de données	Approche Store-All pour d'énormes volumes de données	volumes de données modérés
	Securité	En développement	Pleinement développé
	Durée de stockage	Toutes les données qu'il pourrait utiliser à l'avenir pour toujours	Durée déterminée
Traitement	**Type de traitement**	***Temps réel / Batch***	***Batch***
	Agilité	Haut	Bas
	Schema	Schema-On-Read	Schema-On-Write
	Temps de traitement	Court	Complex
	Traitement de donnees	ELT	ETL
Exploiement	**Requête**	Programmation	SQL
	Accès	Méthode de numérisation	Méthode de recherche
	Utilisateurs	Scientifiques des données	Professionnel des affaires
	Usage	Usages avancés d'analyse data (modélisation prédictive, IA & Machine Learning…)	Usage simple à avoir accès à des reportages et des métriques clés

FIGURE 2.1 – Lac de données et Entrepôts de données

mises en forme selon les standards définis. Dans le lac de données avec l'architecture *schema-on- Read*, la modélisation de données est définie en fonction des demandes d'utilisateurs au moment d'utilisation. De plus, ce schéma permet d'employer des processus ELT (Extract, Load, Transform) au lieu de ETL pour gérer les données pendant la phase de traitement.

Pendant la phase d'*exploitation*, pour extraire les données depuis le lac de données, on a besoin de requêtes plus avancées que les requêtes telles que SQL, puisque ces types de données sont brutes et n'ont pas une structure propre et facile à traiter par les utilisateurs non professionnels. Au contraire, ces architectures nécessitent l'intervention d'experts en sciences des données pour extraire des connaissances recherchées et tirer parti de leurs valeurs inhérentes.

A travers d'analogie entre les deux systèmes de stockage, les fonctionnalités et les bienfaits du lac de données sont facilement observées. Cependant, le lac de données n'est pas considéré comme un substitut aux entrepôts de données. Par conséquent, plusieurs recherches proposent des architectures hybrides pour les systèmes de stockage de données qui incluent les entrepôts de données comme un composant complémentaire. Par exemple [Sawadogo et al., 2019a] a indiqué deux types d'architectures combinant le lac de données et l'entrepôt dans lesquelles un lac de données est considéré comme un source de données pour entrepôt de données et d'autre part, un entrepôt de données est positionné comme un composant de lac de données.

Sur la base de la fonctionnalité générale du lac de données et de ses propriétés distinctives par rapport aux autres systèmes de gestion de données, il est nécessaire de comprendre comment l'infrastructure de ce système de stockage de données massives pourrait prendre en charge les processus concernant le cycle de vie des données. Dans la section suivante, nous prenons position sur l'évolution de l'architecture du lac de données selon les différents travaux menés qui nous encourageaient à proposer la nouvelle définition et concevoir une architecture optimisée pour les lacs de données.

2.3 Architecture de lacs de données

Selon la définition du lac de données, tous les types de données sont autorisés à être saisis dans le lac de données. En revanche, cette permission inconditionnelle pose un risque pour le lac de données de devenir un marécage de données ou lac unilatéral [5] qui accueille toutes les données sans possibilité d'en extraire des connaissances précieuses [Inmon, 2016]. Pour remédier à ces inconvénients, les études liées à la philosophie des lacs de données ont convergé sur des conceptions d'applications et d'architectures efficaces selon les différentes perspectives de ce système de gestion de données. Dans section étude, nous passerons en revue les architectures de lac de données les plus importantes proposées depuis la naissance du terme de lac de données. La typologie de l'architecture du lac de données pourrait être définie comme :

- Architecture basée sur la taxonomie et la maturité des données ;
- Architecture basée sur le type de traitement des données (en batch ou en temps réel) ;

[5] One-way data lake

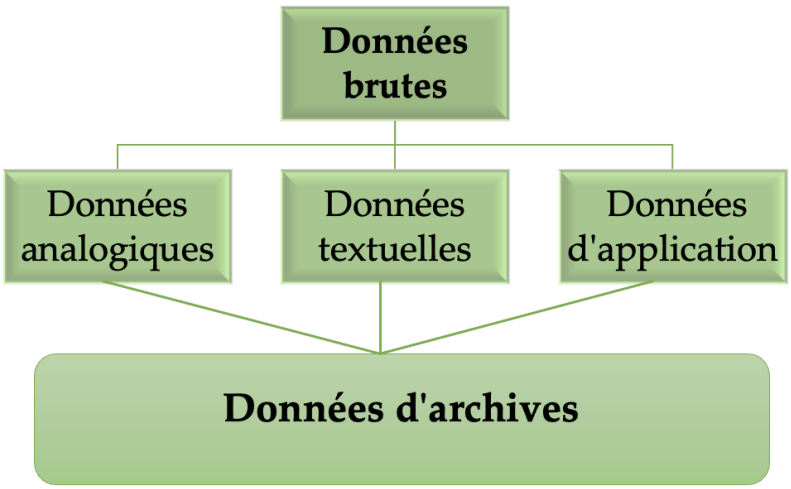


FIGURE 2.2 – Architecture d'Inmon basée sur la typologie de données

- Architecture basée sur le fonctionnement du lac de données pour gestion du cycle de vie des données (architecture en couches).

Dans ce qui suit, nous passerons en revue chronologiquement cette typologie de la structure des lacs de données dans les études réalisées.

L'architecture de référence d'Inmon est fondée sur la typologie de données qui catégorise les données en trois types principaux, tels que les données analogiques, les données d'application et les données textuelles. Selon cette classification, l'architecture d'Imnon consiste en des bassins [6] pour chaque type de données afin de faciliter le processus de traitement et d'analyse de données à travers de lac de données. La figure 2.2 présente l'architecture du lac de données proposée par Imnon.

L'architecture des bassins (Ponds) s'appuie sur le cycle de vie de données depuis l'entrée du lac de données jusqu'à la phase d'archivage. **Le bassin (Pond) de données brutes** accueille les données en forme native depuis des sources variées et rejette les données dans trois autres bassins selon les conditions définies. Les données dans ce bassin sont disponibles pour les professionnels en sciences de données pour les démarches analytiques. Cependant, les données brutes et non-conditionnées pourraient être inutiles pour les utilisateurs ordinaires à l'égard de connaissances sémantiques de données. L'architecture de référence d'Inmon propose une démarche de séparation de données structurées et non structurées dans des bassins (Ponds) particuliers selon leurs sources et leurs formats afin de rendre les données compréhensibles et significatives.

Le bassin (Pond) de données analogiques intègre des données qui sont produites par les machines de manière répétitive comme les outils de mesure ou les données d'un capteur. Par contre, les données non structurées et répétitives comme les données textuelles sont placées dans un bassin qui est assigné particulièrement pour ces types de données. Les données textuelles sont complexes à traiter sous leurs formes brutes. Pour cette raison, dans le bassin textuel les

[6] Pond

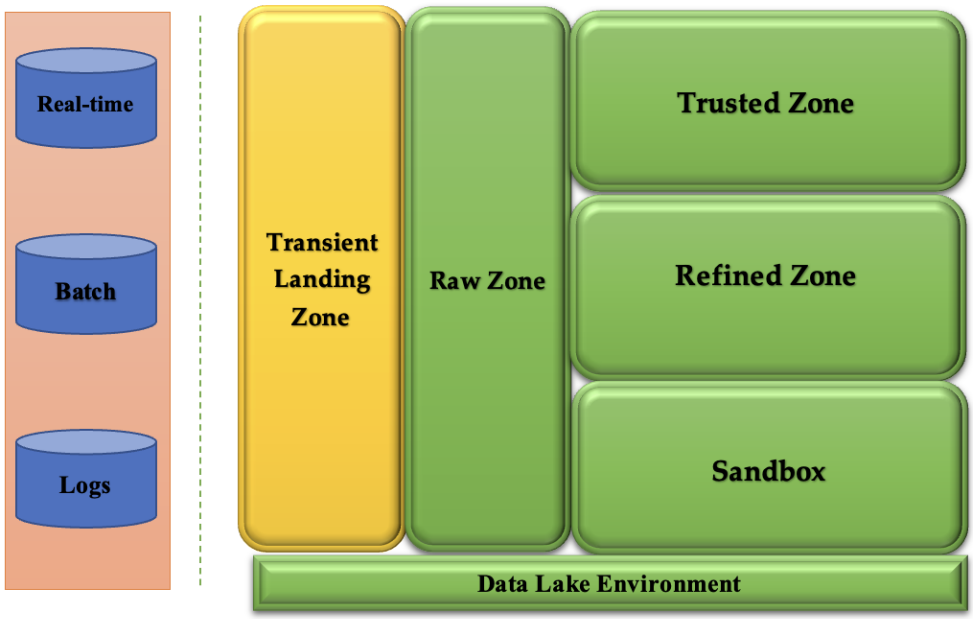


FIGURE 2.3 – Architecture de référence de Zaloni

données sont préparées pour être analysées à l'aide de processus de désambiguïsation de textes et contextes [Inmon, 2016, WH, 2019].

Dans le bassin (Pond) de données d'application, les données sont saisies depuis des applications qui génèrent des données de transactions ou d'autres types de données structurées qui sont gérées normalement par le système de gestion de base de données. Enfin, **le bassin d'archivage** reçoit des données analogiques, textuelles et d'applications afin d'archiver les données pertinentes pour les demandes ultérieures.

L'architecture de lac de données basée sur la typologie de données, est évoluée par la nouvelle plateforme de Zaloni [7]. Zaloni est une entreprise qui travaille sur le développement de structure et fonctionnalités de lac de données. L'architecture de référence de Zaloni est organisée en quatre blocs ou zones principales et une zone optionnelle qui suivent les données tout au long de son cycle de vie selon leurs niveaux de la qualité et de la maturité. Le figure 2.3 illustre les blocs consécutifs dans lesquelles chaque zone reçoit la sortie de bloc précédent comme entrée et envoie les données pour les blocs suivants.

Le premier bloc dans l'environnement d'architecture de Zaloni est *Transient landing zone* qui est une zone optionnelle et privée pour le stockage temporaire des données selon les attentes des organisations. Les données brutes en forme originale sont saisies immédiatement dans la *Raw zone* pour la préparation des données avant leur consommation. Dans cette zone, des processus ETL sont effectués, les métadonnées deviennent disponibles pour tous les utilisateurs, la sécurité et la qualité de données sont vérifiées et la gestion du cycle de vie de données est déployée selon les règles définies. Les données de cette zone partent à *Trusted zone* ou *Sandbox zone*. Les données

[7] https://www.zaloni.com/resources/webinars/four-zones-data-lake-architecture/

qui arrivent dans *Trusted zone* sont mises en application des standards et elles obtiennent des certifications afin d'être autorisées à accéder et consommer par les utilisateurs . Dans *Refined zone* les données sont regroupées en data marts qui sont des sous-ensembles d'entrepôts de données totales pour les requêtes particulières et interrogées normalement par SQL ou Hadoop SQL. Les données dans cette zone pourront être ingérées directement par *Raw zone*.

Pour finir, *Sandbox zone* accueille les données depuis toutes les d'autres zones telles que *Raw zone*, *Trusted zone* et *Refined* pour les processus de traitements, d'exploitations et d'explorations.

Afin de pallier les inconvénients des architectures en bassin (Pond) et en zone tels que le risque de perte de données et l'interruption du lignage des données, [Sawadogo and Darmont, 2021] révise une catégorie pour adresser l'architecture de lac de données en ce qui concerne de type de composants qui sont définis pour structurer le lac de données. Cette catégorie est définie comme *Fonctionnelle* $\times$ *maturité* architecture, qui est une plateforme hybride basée sur une architecture fonctionnelle et mature, et compense les lacunes et insuffisances de ces deux architectures par rapport à l'état dans lequel elles sont mises en œuvre indépendamment. Selon d'étude de [Sawadogo and Darmont, 2021] une architecture fonctionnelle contient les fonctions pour chaque phase de cycle de vie de données comme l'ingestion, le stockage, le traitement et l'accès, et d'autre part, l'architecture basée sur la maturité des données contient des composantes liées au niveau de raffinage de donnée qui est toutes pareilles des architectures de zones. Afin de renforcer des fonctionnalités de ces deux architectures, une structure hybride avec des composants qui demandent des fonctions de cycle de vie de donnée ainsi que de niveau de la purification est exigeante.

Le type de traitement de données réalisé en temps réel ou par lot, est une problématique dans le domaine de la conception de l'architecture du lac de données. D'une part, pour les données d'ingestion par lots, un système de traitement par lots est proposé, comme Hadoop [Barbierato et al., 2013]. D'autre part, pour les données en temps réel, le système qui prend en charge les données en flux est mis à disposition, comme Spark [Abadi et al., 2003]. En revanche, certaines architectures sont capables de mettre à disposition des traitements des données en temps réel ou réaliser le processus de données en lot et en temps réel dans le même temps comme les architectures Lambda et Kappa [Giebler. et al., 2018, Casado and Younas, 2015, Kreps, 2014]. En se basant sur cette exigence, l'architecture Lambda a émergé pour présenter un environnement hybride de traitements des données et prépare les moyens flexibles pour servir les grands ensembles de données en lot et en temps réel tout en réduisant le temps de réponse [Marz and Warren, 2013, Tomcy, 2017].

L'architecture Lambda proposée par [Marz and Warren, 2013] est une instruction ou modèle de base qui considère les systèmes de données comme une plate-forme qui est potentiellement résistantes aux panne de traitement de données historique (lot) et nouvellement généré dans le même environnement. Cette architecture est concentrée sur trois couches fondamentales telles que la couche de *Batch*, la couche de *Speed*, et la couche de *Serving* qui sont conçues exclusivement dans le but de traiter les données en lot et en temps réel de manière efficace.

Selon la figure 2.4, [Tomcy, 2017] a établi une architecture du lac de données en considérant la couche *Lambda* qui est une couche supplémentaire par rapport d'autres structures définies. Cette couche fournit les traitements hybrides pour le lac de données. Comme toutes les autres architectures, la couche d'acquisition reçoit tous les types de données en faible latence et les

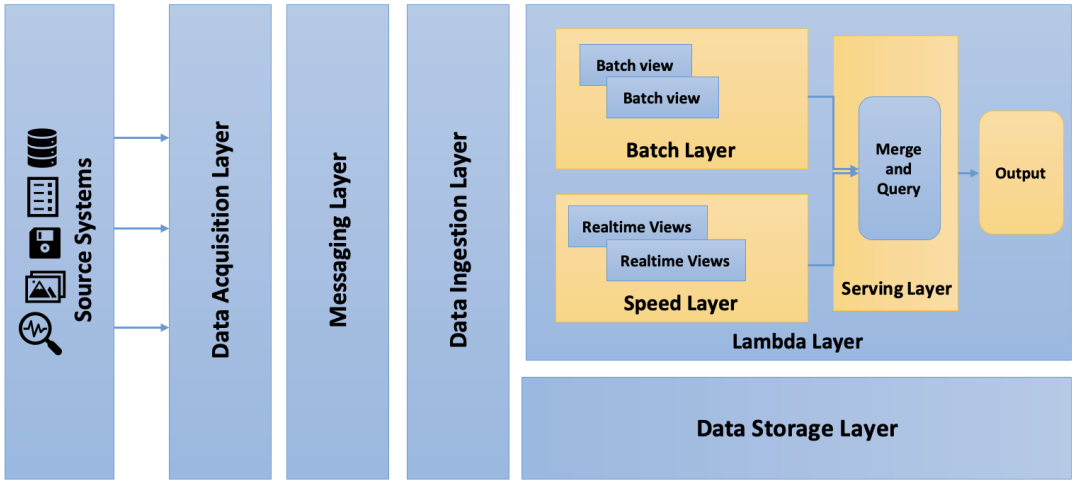


FIGURE 2.4 – Architecture de lac de données Lambda [Tomcy, 2017]

prépare pour la couche ciblée. La couche de message organise le middleware orienté message [8] pour lac de données en assurant la livraison des messages pour les utilisateurs et mettre en file d'attente et retirer les messages en écriture et en lecture. La couche d'ingestion est une couche rapide et hautement évolutive pour les requêtes à la demande. Cette partie ingère et transforme rapidement les données acquises en formats ciblés et garantit les propriétés des données pour les couches suivantes [Tomcy, 2017].

La couche particulière dans cette architecture est la couche *Lambda* qui contient trois couches complémentaires telles que la couche de *Batch*, la couche de *Speed*, et la couche de *Serving*. Cette couche avec tous ses composants sont responsables du traitement des données et de la répondre aux requêtes dès que possible et avec une faible latence. La couche de *Batch* est considérée pour le traitement des données qui est ingéré en lots. Dans cette partie, les données brutes et immuables sont converties en format échangeable selon les modèles de données définis ainsi qu'améliorer et rassurer la qualité de données. Par contre, la couche de *Speed* est préparée pour traitement des données modifiables en temps quasi réel. Pour cette raison la prise en charge d'opérations de traitement rapide et efficace est la caractéristique principale de cette partie qui devrait être prise en compte pour le lancement de cette couche. La troisième couche dans la couche *Lambda* est la couche de *Serving* qui fusionne des résultats des couches de *Batch* et de *Speed* dans une manière intégrée et organise des requêtes correspondants de ces deux couches pour les utilisateurs finaux. La dernière couche mais pas des moindres, est la couche de stockage de données qui accueille toutes les données en lot et en temps réel et prend en charge les opérations sériées et aléatoires [Marz and Warren, 2013, Tomcy, 2017].

L'architecture Lambda est une référence au fonctionnement hybride afin d'affronter l'utilisation des données massives dans l'environnement de Big data. Basée sur cette idée, une nouvelle architecture de traitement hybride est construite par [Giebler. et al., 2018] qui s'appelle BRAID. Selon [Giebler. et al., 2018], les objets connectés [9] exigent les outils combinés pour gérer les don-

[8] Message Oriented Middleware
[9] Internt of Things

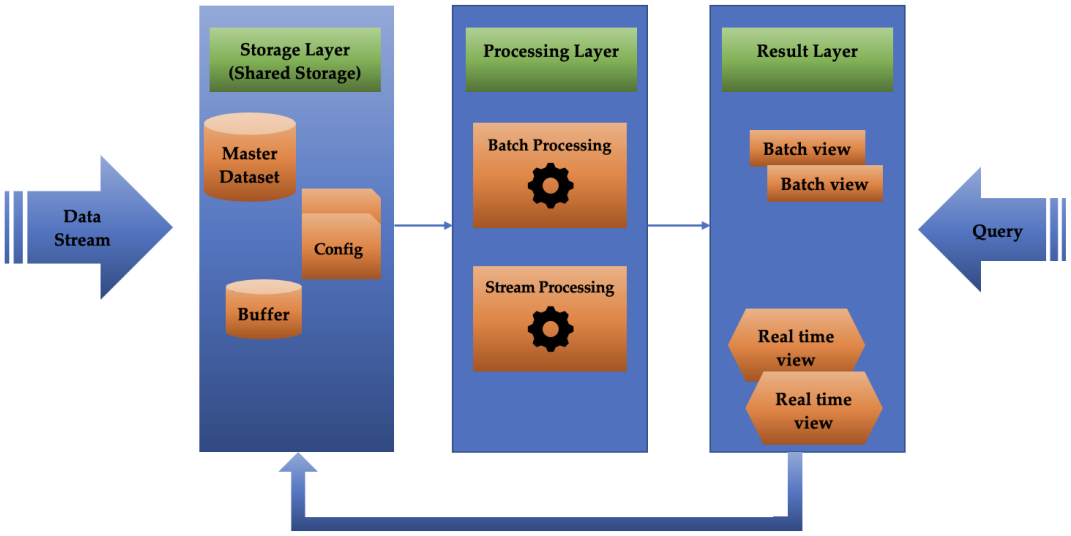


FIGURE 2.5 – Architecture de référence de BRAID [Giebler. et al., 2018]

nées historiques (qui contiennent les informations prédictives) aussi que données fluxionnelles (qui contiennent les informations perspectives). Dans cette architecture proposée, les données de base [10] sont traitées par moteur de traitement par lots et les données en flux sont traitées par moteur de traitement par temps réel afin d'obtenir les résultats d'analyse exhaustive.

Cette architecture est concentrée sur le traitement de données historique et fluxionnel avec un canal de communication entre deux couches (Batch et Temps réel) et contient de trois couches principlas. *La couche de stockage* de données maintient tous les flux de données (vers l'avant et vers l'arrière) afin de préparer les données pour la couche de traitement analytique. Il prend également en charge la réception de données traitées à partir de couches de résultats et de couches de traitement pour des utilisations supplémentaires.

La couche de traitement compose les branches de traitement en lot et en temps réel et les résultats de processus de ces branche sont stockés comme vues par lots et vues par flux dans la couche de résultats. Les résultats de traitement de données soit en lot soit en flux, sont gardée dans *la couche de résultats* et sont disponibles pour tous les requêtes (Batch ou Stream) avec une structure de métadonnées propres. Le point particulier de l'architecture de BRAID par rapport d'autres structures étudiées, est l'interconnexion entre les couches de résultats et de stockage qui assure combinaison automatique de résultats et les analyses exhaustive.

L'architecture hybride Lambda est devenue une infrastructure convenable pour lac de données qui ingère tous types de données depuis plusieurs sources. Les bienfaits de Lambda provoquent des facultés pour un lac de données exhaustive afin de mise en ouvre les traitements complets pour les données historiques et en flux. Ces bienfaits incluent :

- Traitement à faible latence les données de base (Master data) grâce à couche de *Batch* en créant vues par lots (Batch views) ;

[10] The Master Data

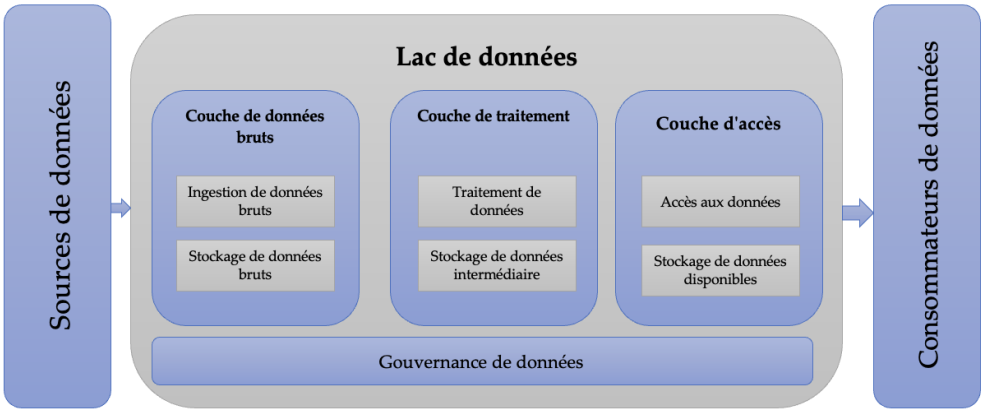


FIGURE 2.6 – Architecture de lac de données proposée par [Ravat and Zhao, 2019]

- Traitement des données récentes grâce à couche de *Speed* en stockant et mettant à jour les vues en temps réel des données ;
- Fusion des résultats de la couche de *Batch* et de la couche de *Speed* d'une manière composée et vite grâce à couche de *Serving* ;
- Calcul évolutif ;
- Interconnexion entre les couches différentes qui garanti les résultats analytiques ;
- Fonctionnalités hybrides élevées ;
- Haute flexibilité pour l'ingestion de données en formats bruts ;
- Recalcul qui permet de corriger la tolérance aux pannes.

Une large catégorie d'architecture de lac de données fait partie de l'architecture en couches qui contient les différents zones liés de la préparation des données dans les différentes phases de sa vie, par exemple couche d'ingestion, couche de stockage, couche de traitement et couche d'accès [LaPlante, 2016]. L'arrangement, le nombre et les types de ces couches sont variés dans les différentes études. Par exemle, [Ravat and Zhao, 2019] présente une architecture fonctionnelle de lac de données basée sur quatre couches importantes 2.6 :

- **La couche de données brutes** qui ingère et stocke les données brutes d'entrée ;
- **La couche de traitement** qui traite les données et les stocke de manière intermédiaire ;
- **La couche d'accès** qui mis en disposition les données traitées ainsi que un stockage de données disponibles ;
- **La couche de gouvernance** qui contrôle et vérifie la qualité des données dans toutes les autres couches.

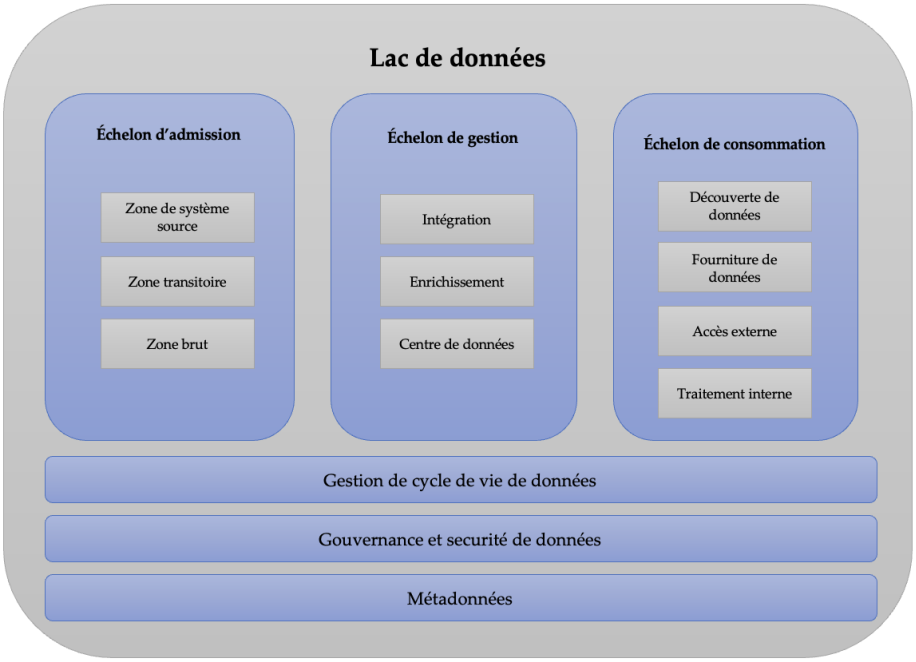


FIGURE 2.7 – Architecture de lac de données proposée par [Pradeep, 2015]

Une autre forme de pipeline de lac de données en couches est abordée par [Pradeep, 2015] dans lequel trois échelons d'opération des données et trois couches de gestion de la qualité des données sont conçues pour cette architecture. Les trois échelons d'opération sont classés en échelon d'admission [11], échelon de gestion [12], et échelon de consommation [13]. Les trois couches de gestion sont la couche de gestion du cycle de vie des données, la couche de gouvernance et de sécurité des données et la couche de métadonnées 2.7.

Une étude menée sur la conception d'une architecture efficace de lac de données est liée au travail de [Kachaoui and Belangour, 2020] qui propose un lac de données intelligent en s'appuyant sur la gestion optimale des métadonnées. Cet écosystème adresse un réservoir hybride qui contient de lac de données et entrepôts de données comme un composant complémentaire. Le figure 2.8 montre l'architecture étudiée qui est composée de des couches essentielles telles que : couche d'acquisition, couche d'exploration, couche sémantique et couche de perspicacité.

Dans *la couche d'acquisition*, les données de référence en format brut sont collectées et les métadonnées sont mises à jour afin de préparer les données pour la phase analytique. Dans *la couche d'exploration* les données sont regroupées selon leurs métadonnées et mises en disposition pour les processus analytiques afin de trouver les tendances pertinent et les liens logiques. *La couche sémantique* est la partie qui servit les utilisateurs finaux en permettant de générer les requêtes pour accéder les données formulaires et traitées selon leurs besoins. L'interprétation depuis des données massives est une tâche supérieure que les processus analytiques et exigent

[11] Intake tier
[12] Management Tier
[13] Consumption Tier

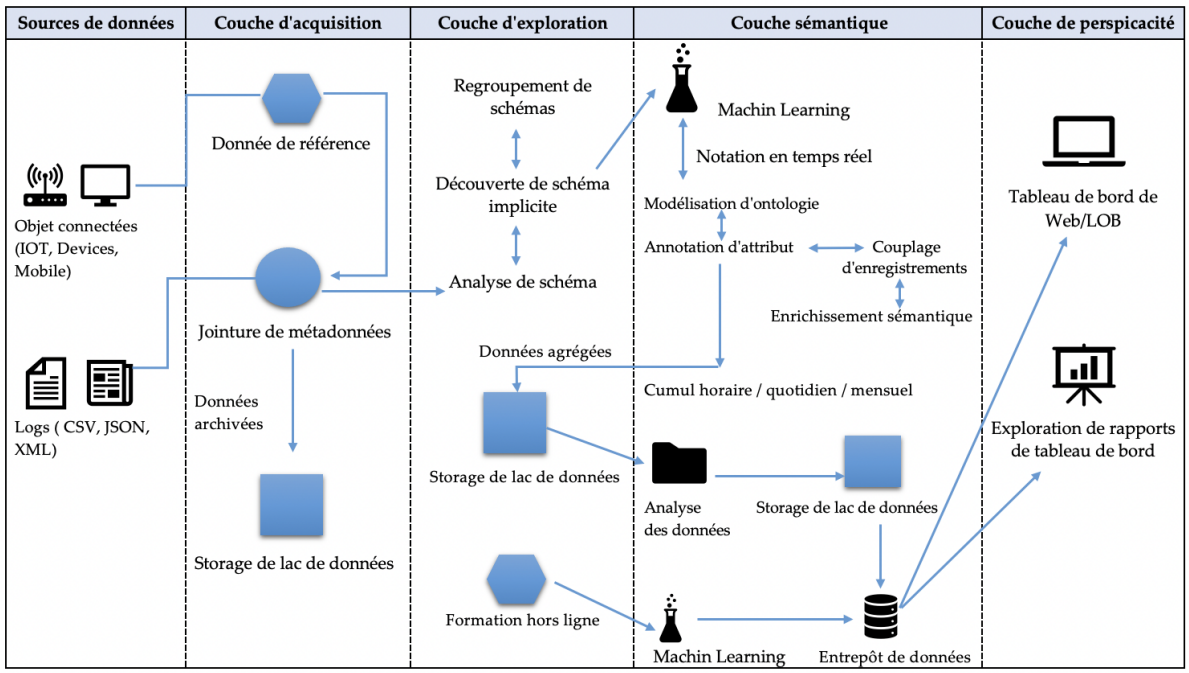


FIGURE 2.8 – Architecture de lac de données intelligent proposée par [Kachaoui and Belangour, 2020]

les outils avancés. *La couche de perspicacité* prépare les dispositions pour interpréter les données filtrées et exploiter les connaissances cachées qui sont difficiles à tirer parti par les utilisateurs communs [Kachaoui and Belangour, 2020].

Un travail récent sur l'architecture en couches s'intéresse à la proposition d'une infrastructure pour un projet de gestion et d'analyse de données archéologiques. Dans ce projet, neuf couches sont présentées liées au cycle de vie des données [Liu et al., 2021]. Selon la figure 2.9, ces couches sont dénommées comme suit :

- **La couche de source de données** qui fait référence à diverses sources de données d'entrée ;
- **La couche d'ingestion** qui prépare les outils pour ingérer les différentes formes de données et contrôler les métadonnées concernées ;
- **La couche de stockage** stocker et conserver un grand volume de données multi-structurées ;
- **La couche de distillation des données** qui a fourni les procédures pour nettoyer et purifier les données en double ou erronées ;
- **La couche de la connaissance** [14] qui transforme les données brutes en connaissances précieuses ;

[14] Insights layer

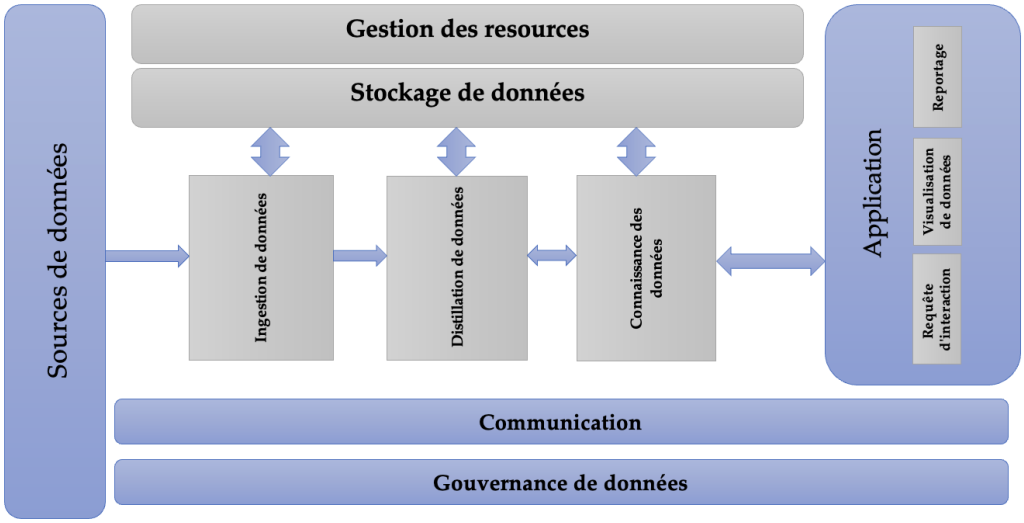


FIGURE 2.9 – Architecture de lac de données ArchaeoDAL proposée par [Liu et al., 2021]

- **La couche d'application de données** qui met en œuvre les outils d'exploration et d'utilisation des données par les utilisateurs ;
- **La couche de gouvernance** qui est responsable des contrôles de qualité et de sécurité des données ;
- **La couche de gestion de ressources** qui prépare les planifications de tâches pour les serveurs ;
- **La couche de communication** qui est un dispositif de communication des autres couches afin de garantir l'intégrité et la coordination des procédures dans le lac de données.

Le lac de données avec une architecture bien conçue est annoncé comme capable de couvrir toutes les attentes et exigences pour un système intégré de stockage de données. Cependant, il reste encore des enjeux essentiels qui devraient être considérés comme des éléments fondamentaux de l'architecture des lacs de données. En s'appuyant sur ces besoins, [Giebler et al., 2021] présente les principes pour concevoir une architecture exhaustive de lac de données dans laquelle *tous les aspects architecturaux nécessaires d'un lac de données et leurs interdépendances sont couverts* qu'elle est nommée DLAF [15]. Les principes de DALF est construit sur neuf aspects architecturaux pour déployer un lac de données complet, tels que l'infrastructure, le stockage de données, le flux de données, la modélisation des données, l'organisation de données, le processus de données, la gestion de métadonnées, la sécurité et la confidentialité des données, et la qualité de données. Selon cette méthodologie, il faut tenir en compte tous les neuf aspects pour implémenter une architecture exhaustive de lac de donnée d'une manière régulière.

Parmi les études menées en architecture de lac de données, la définition de [Madera, 2018] analyse la nouvelle position des lacs de données au sein des systèmes d'information par rapport aux attentes des organisations pour les procédures de la prise de décision efficace. Dans

[15] Data Lake Architecture Framework

cette étude, les lacs de données sont considérés comme un composant auxiliaire dans le système d'informations qui positionne en parallèle des systèmes décisionnels. En outre, [Madera, 2018] a indiqué le terme *démarche d'urbanisation* qui est proposé par [Servigne, 2008] pour la conception de l'architecture d'un lac de données qui contient quatre phases principales basés sur les besoins des organisations concernées, telles que *l'architecture métier, l'architecture fonctionnelle, l'architecture applicative, l'architecture technique* respectivement. Dans ses travaux, les lacs de données dans le système d'informations sont comparés avec des systèmes décisionnels selon les caractéristiques de systèmes stockages comme :

- L'acquisition de données ;
- Le catalogage de métadonnées ;
- Le stockage de données ;
- L'exploitation de données ou la fouilles de données ;
- La gouvernance de données.

Selon la figure 2.10, cette architecture proposée pour un système d'information qui contient le lac de donnée comme un élément complémentaire, est concentrée sur la l'aspect gouvernance des données afin d'assurer la qualité et la sécurité de données au sein des systèmes de pilotage d'une organisation. La gestion de données est une problématique dans ce lac de données qui a été moins prise en compte dans d'autres travaux à propos de l'évolution du concept de lac de données.

En conséquence, la meilleure mise en œuvre de l'infrastructure du lac de données conformément à tous les critères à prendre en compte pour la concevoir, dépend de plusieurs facteurs et des enjeux principaux qui rendent parfois difficile la réalisation d'un lac de données en accord avec toutes les attentes des organisations. Dans la section qui suit, nous nous appuierons sur ces questions et problèmes critiques pour savoir comment implémenter un lac de données efficace tout en respectant les limitations potentielles.

2.4 Les enjeux de lac de données

Le lac de données en tant que système de gestion de données est confronté avec les nombreuses finalités de gestion dans le cycle de la vie de données. L'architecture inédite et hybride de ce système de stockage de données permet d'accueillir les données brutes de toute une variété de sources et les traiter en temps réel et par lot. Pourtant, le principe d'accepter des données de tous types de formats risque d'altérer la qualité et la propreté des informations et met en évidence l'importance des processus de cataloguage, l'évaluation de la qualité et la gouvernance et la gestion de données tout au long du cycle de la vie de données. Pour cette raison, le besoin de concevoir et d'implémenter un lac de données fiable et utilisable engendre des enjeux considérables par rapport de véracité de données. Afin de concevoir une architecture de référence de lac de données fiable et valoriser des données en haute qualité, les enjeux stratégiques doivent être pris en compte, comme nous le présentons dans la suite.

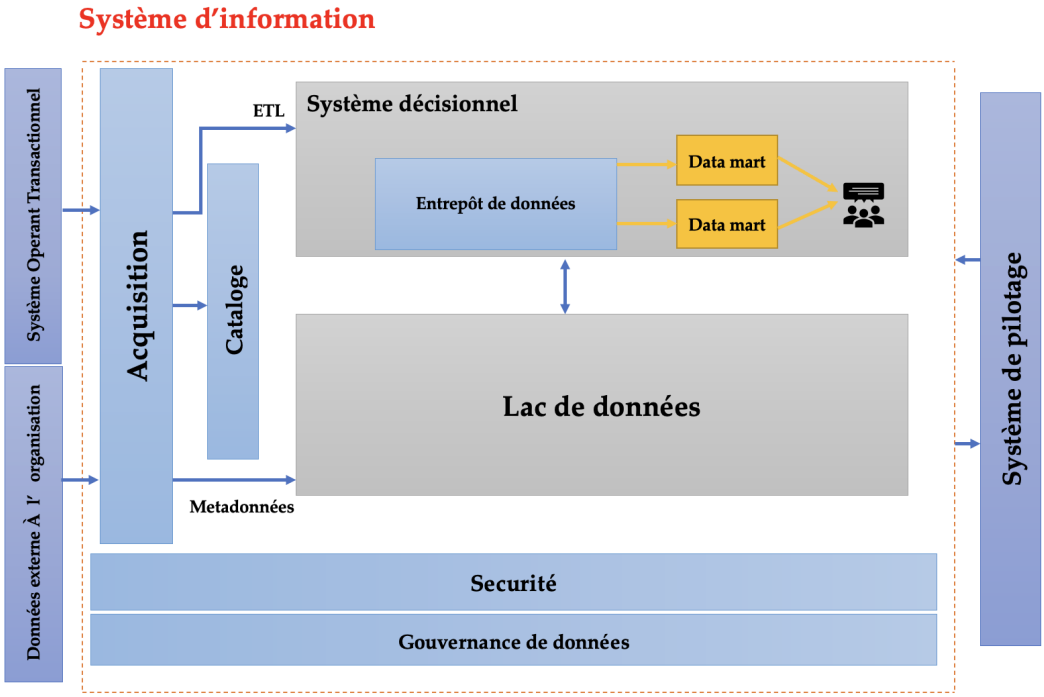


FIGURE 2.10 – Position du lac de données dans le système d'information, proposée par [Madera, 2018]

2.4.1 Métadonnées

La métadonnée est l'information structurée concernant l'origine et les ressources des données, elle comprend notamment les informations techniques sur la façon de générer la donnée, sa maintenance, et la réutilisation de données [Foulonneau and Riley, 2014]. La métadonnée rend les données lisibles, accessibles, et traçables d'une manière compatible afin que les machines ou utilisateurs soient capables de d'interagir à travers de ces informations pratiques qui révèlent les caractéristiques des données. En sachant que les données sont considérées comme un produit dans l'environnement commercial de big data, la métadonnée est un élément d'information qui décrit les ressources de données, comme par exemple les informations jointes sur un emballage de produit, qui incluent des informations à propos des ingrédients et de façon dont le produit est fabriqué, transporté, livré, et conservé [L, 2013]. De la même façon, la métadonnée -selon la faculté de droit d'Harvard [16]- est une *empreinte digitale* [17] de données qui est embarqué les informations liées des types de ressources, la mode de stockage, le traitement et l'analyses de données. Une structure adaptée de métadonnées contient les éléments principaux qui décrivent les ressources en détail et facilitent l'identification des créateurs et propriétaires de ressources, sujet de ressources, l'information technique, et le droit pour l'utilisation de données [L, 2013].

La structure des métadonnées est catégorisée dans plusieurs formats selon les attentes des utilisateurs ou propriétés de logiciels qui échangent ou dérivent ces métadonnées. Les métadonnées

[16] https://hls.harvard.edu/dept/its/what-is-metadata/
[17] Fingerprint

FIGURE 2.11 – Taxonomie des critères pour choisir un standard de métadonnées

peuvent être classées en cinq formats, *descriptif*,*administratif*, *accès ou utilisation*, *conservateur*, et *structurel* [Hillmann et al., 2008]. De plus, pour chaque format et dans chaque domaine, certains standards sont définis afin de déterminer le nombre et le type d'éléments fondamentaux pour créer des métadonnées logiques et lisibles. En revanche, la prise de décision pour choisir la bonne structure de métadonnées est une prise de décision multicritères qui influent directement sur la gestion de métadonnées. Pour cette raison, certains aspects devraient être considérés lors de la sélection de standard approprié afin de mettre un équilibre et faire le compromis entre les coûts de la génération de métadonnées et la capacité d'interopérabilité. La figure 2.11 montre la taxonomie des critères qui normalement tient compte du standard pour la métadonnée[18].

On parvient à l'interopérabilité de métadonnées à travers quatre couches importantes qui adressent toutes les enjeux pour définir et établir des métadonnées telles que les protocoles, format de représentation, les schémas, et sémantique des concepts [Duval, 2001]. L'un des standards les plus connus pour mettre en forme des métadonnées lisibles par une machine est Dublin Core [19] qui définit les schémas adaptés pour les métadonnées dans tous les domaines. En outre, par rapport des formats communs pour représenter les métadonnées échangeables, on pourrait nommer XML, RDF et pour un exemple de Web HTML, comme des langages équitables à la mise en oeuvre de structure de métadonnées.

Au-delà de la structure de métadonnées, la mise en place des systèmes gestionnaires de métadonnées est une issue problématique dans l'environnement de big data. Par ailleurs, les données multi-structures et multiformats exigent l'existence d'un programme de gestion qui contrôle l'efficacité du système de métadonnées depuis les ressources et facilitent les traçabilités

[18] http://www.ukoln.ac.uk/qa-focus/documents/briefings/briefing-63/html/
[19] https://dublincore.org/specifications/

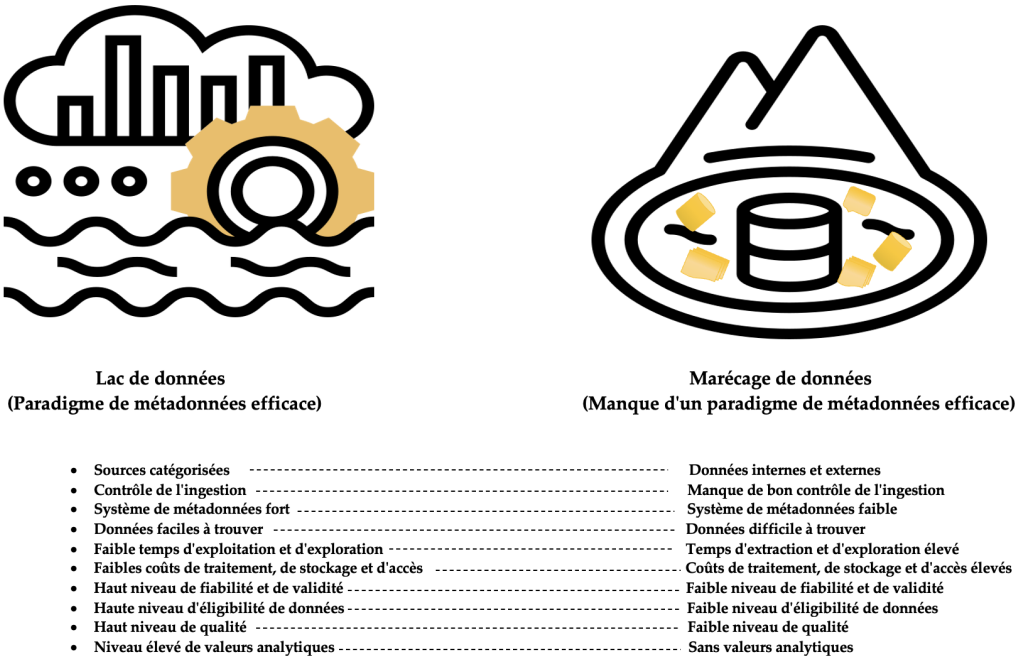


FIGURE 2.12 – Lac de données et Marécage de données

de données tout au long de son cycle de la vie. En revanche, un système inapproprié de gestion de métadonnée pourrait influencer la véracité et qualité d'informations qui sont dérivées à travers de analyse de données et risque des systèmes de gestion de données de se remplir avec énorme quantité de données inutiles. Le phénomène dans laquelle les systèmes de stockage ou les systèmes de gestion de données sont remplis de données non organisées, inutiles, introuvables, inaccessibles et avec le mauvais système de métadonnées qui n'ont aucune valeur pour traiter et analyser, qui s'appelle *marécages de donnés* [20]. Le marécage de données est considéré comme un risque critique pour les systèmes de stockage de données hétérogènes qui menace la qualité et l'utilité des données où le lac de données s'est transformé en un réservoir ingérable et inexploitable [Giebler et al., 2021, Munshi and Mohamed, 2018]. Selon la figure 2.12, le lac de données unilatéral est un phénomène indésirable qui augmente le temps et les coûts de traitement, d'analyse et d'exploitation des données inefficacement et doit être contrôlé et géré avec les meilleures stratégies de gouvernance et de gestion des données.[Inmon, 2016].

Par conséquent, afin de mettre en œuvre les systèmes de métadonnées propres pour les systèmes stockage de données, les méthodes efficaces sont indispensables. Les objectifs principaux des stratégies de gestion de métadonnées et des données sont de vérifier l'éligibilité des données et d'évaluer leur viabilité pendant leur présence dans le lac de données afin d'éviter le phénomène de marécages de donnés. Pour cette raison, [Vemuganti, 2013] propose un cadre de gestion des métadonnées pour l'analyse des données massives qui évalue la validité de métadonnées pendant son cycle de la vie. Dans ce cadre 2.13, les niveaux de gestion de métadonnées sont considérés comme les niveaux de préparation de produits dans un système logistique qui contient la découverte de métadonnées, la collection de métadonnées, la gouvernance de métadonnées, le stockage

[20] Data swamp

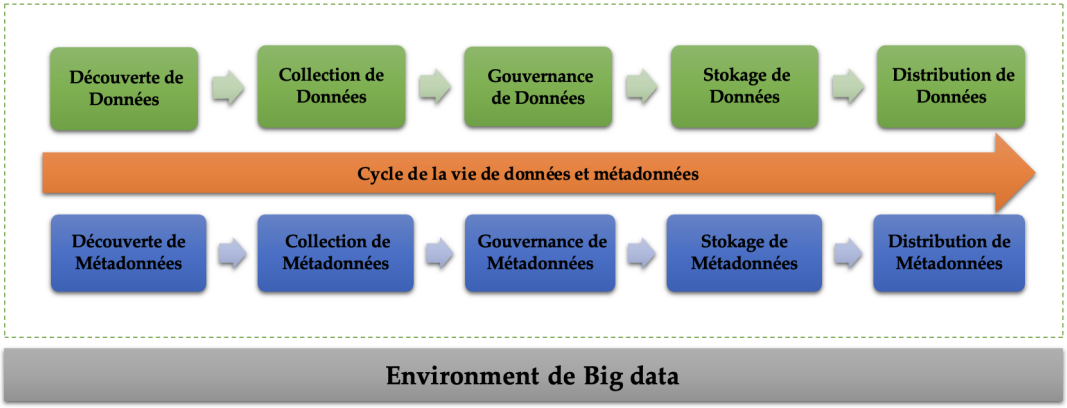


FIGURE 2.13 – Cadre de gestion des métadonnées [Vemuganti, 2013]

de métadonnées, et la distribution de métadonnées.

Selon le cadre défini dans la figure 2.13, la gestion de métadonnées est considérée en parallèle et d'une manière inséparable avec la gestion de données tout au long de chaîne de valorisation de données qui assure la surveillance de traçabilité et la validité de données dans l'environnement du big data. En ce qui concerne l'importance de gestion de métadonnées dans le lac de données qui est une plateforme accueillant de données brutes, [Sawadogo et al., 2019b] introduit un modèle de métadonnées basé sur le graphique pour la gestion de métadonnées en lac de données qui s'appelle MEDAL [21]. MEDAL est un modèle générique et complet qui offre d'ensemble de fonctionnalités et critère pour l'implémentation un système de métadonnées raisonnables, telles que la valorisation d'aspect sémantique de données, l'optimisation de la requête de donnée par indexation, l'intégration entre les jeux de données pertinentes, la multi-représentation des mêmes données et, la conservation de l'historique de l'évolution des données. MEDAL s'est positionné comme référence par rapport aux autres concurrents à définir un système intégré de gestion de métadonnées.

Compte tenu de l'importance du système de métadonnées pour l'interopérabilité des données, le terme catalogage des données est devenu un nouvel enjeu dans le domaine de l'indexation et de l'identification des données. *Oracle*[22] definit le catalogue de données comme suit :

" Le catalogue de données est un outil qui utilise des métadonnées pour aider les procédures de la gestion de données. Il facilite également la collection, l'organisation, l'accède et l'enrichissement des métadonnées en prenant en charge de gouvernance de données. "

La métadonnée est un outil descriptif d'ontologie de données qui décrit l'origine, le contenu et les caractéristiques techniques de données grâce aux éléments prédéfinis selon les standards spéciaux. Toutefois, la vitesse d'accessibilité aux données propres et sécurisées pour professionnel de données, met en exergue l'existence de catalogue de données. Le catalogue de données est un levier pour gestion de métadonnées qui contient des informations pratiques à propos des métadonnées et jeux de données disponibles. De plus, cet inventaire des données [23] rend les données

[21] Metadata Model for Data Lakes
[22] https://www.oracle.com/fr/big-data/what-is-a-data-catalog/
[23] https://www.ibm.com/cloud/learn/data-catalog

plus détectables et consultables pour les utilisateurs et améliore les démarches de découverte des données pertinentes et la gouvernance de données.

Après avoir énoncé que le système de gestion de métadonnées a un impact manifeste sur la rendement et l'efficacité du système du stockage de données massives, un cadre convenable de métadonnées devrait engendrer les avantages pour les systèmes d'information telles que :

- Le standard multidisciplinaire exhaustif pour la documentation l'identification, la gestion technique, le traitement, la conservation, l'exploitation, l'analyse et l'interopérabilité de données ;
- Les disciplines qui assurent la viabilité et la véracité de données ;
- Les méthodes qui facilitent la découverte de données et d'informations pertinentes par les utilisateurs ;
- Les standards pour authentification et certification de données ;
- Les sémantiques modèles qui recouvrent hétérogénéités sémantiques à chaque niveau d'agrégation d'informations ;
- Les facultés pour représenter les contenus, le contexte, et la structure de données [Baca, 2016] ;
- Les outils pour révéler les relations dedans et entre les données et informations pertinentes ;
- Les modèles d'ontologie qui prennent en charge les sémantiques exploitables par machine ;
- La garantie de rétention d'intégrité entre les données actuelles et archivées ainsi que les enregistrements historiques ;
- Les régulations sécurisées pour protection de données sensibles ;
- L'infrastructure compatible pour les processus de Mapping.

2.4.2 Gouvernance de données

La gouvernance des données d'un lac de données est le sujet le plus problématique et discuté dans les domaines de gestion des systèmes d'information. L'échange des données à l'intérieur et à l'extérieur de l'organisation ainsi que l'exactitude de démarches d'extraction des savoir-faire, exigeant les protocoles appropriés pour la gestion de la sécurité des données. Toutefois, le lac de données conçu comme un réceptacle de données multi-structurées, entraîne de nombreux avantages par rapport à un système de stockage de données traditionnel tel que les entrepôts de données.

La flexibilité du lac de données pour accueillir toutes les formes de données brutes et la faculté pour supprimer les processus de normalisation et la préparation de données, sont les avantages plus importantes pour un système stockage de données massives. Pourtant, ces propriétés augmentent les risques de redondance de données et menacent la sécurité et la confidentialité de lac de données. Le saturation le lac de données avec des données inutiles et de mauvaise qualité

conduit à un marécage de données qui transforme le lac de données en une boîte de données sans valeur [Madera, 2018]. Ces inconvénients du lac de données provoquent les reproches et critiques du lac de données en comparaison avec d'autres systèmes de stockage normalisés. Pour cette raison, la gouvernance de données est devenue un élément important dans la conception de l'architecture de lac de données afin d'avérer la fiabilité de données pendant le cycle de vie. L'implémentation de cette gouvernance de données est indispensable au cas où les organisations, qui travaillent dans les domaines de valorisation de données, accentuent la sécurité de fonctionnement des aspects sémantiques afin de tirer parties de données pertinentes et hautes qualité.

La gouvernance de données est définie comme toutes les stratégies, les régulations et les principes pour garantir la sécurité des systèmes informations et gouverner les données qui viennent des ressources éparpillées. Le terme gouvernance de données est corrélé avec la gestion de donnée et la gestion de l'information d'entreprise [24]. Pour cette raison, [Ladley, 2012] a défini la gouvernance de données en ce qui concerne le corps de la connaissance en gestion des données [25] comme ensemble de procédures d'autorité, de contrôles et d'administration de données en conformant des politiques définis sur la gestion des actifs de données.

Selon [Ladley, 2012], il faut distinguer la gestion de données et les processus qui valident la qualité de gestion de données. La gestion de données est l'ensemble des programmes et politiques qui gère et contrôle la valeur de données en horizon temporel spécifié alors que la gouvernance de données est un projet permanent qui contient d'ensemble de principes pour vérifier et affirmer l'exactitude de la mise en œuvre de la gestion des données. [Ladley, 2012] a aussi utilisé le métaphore de chaîne d'approvisionnement pour démontrer les relation entre la gestion de données, la gestion de l'information d'entreprise et la gouvernance de données. Selon cette métaphore, la gestion de l'information d'entreprise est considérée telle que la gestion de la chaîne d'approvisionnement qui fournit une philosophie générale pour orienter les membres vers des objectifs globaux, la gestion des données est considérée telle que la gestion de l'inventaire qui contrôle et surveille les actifs de données, et la gouvernance de données est considérée tels que les standards et règles qui vérifient la qualité de l'implémentation des données en gérant le coût et la véracité de manipulation des données. La figure 2.14 montre cette métaphore qui présent les données comme l'actif d'entreprise et précise les priorités pour la démarche de gestion de données.

D'une façon générale, le terme de " gouvernance " indique tous les démarches, les règles et les disciplines pour diriger et gouverner les ensembles des activités décisionnelles d'une communauté particulière. Tandis que, la gestion d'un système et la gouvernance de système sont distingués aux plusieurs mesures par rapport ses fonctionnalités, ses stratégies, ses responsabilités, et ses buts, ils sont forcément complémentaires afin de piloter un système d'une manière intégrée. Selon les définitions au-dessus et la table 2.1, les démarches gestionnaires correspondent aux toutes les stratégies et les décisions pour organiser les entités des systèmes vers les objectifs stratégiques ou tactiques, alors que les procédures de gouvernance font partie comme des éléments indispensables de la gestion du système et sont considérés tels que des contrôleurs qui vérifient la mise en œuvre des stratégies gestionnaires selon les règles et les buts prédéfinis.

Par conséquent, la gouvernance des systèmes d'information est un enjeu considérable dans les domaines de la gestion de données massives de points de vue informatique et juridique. Le lac de données comme un entité importante de systèmes d'information, est envisagé aux risques de

[24] Enterprise Information Management
[25] Data management Body of Knowledge (DMBOK)

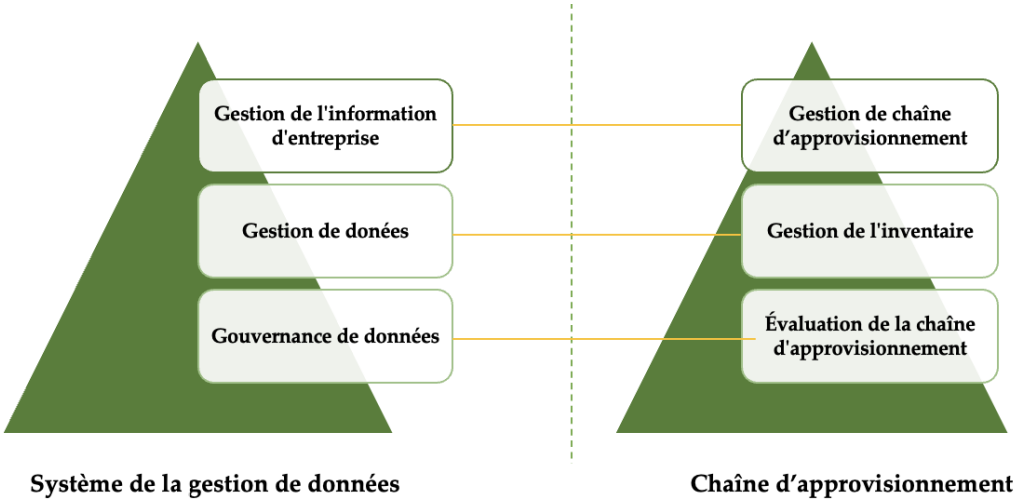


FIGURE 2.14 – Analogie entre le système de gestion de données et la chaîne d'approvisionnement [Ladley, 2012]

Définition de la gouvernance de données	Source
La gouvernance de données fait partie de la gestion de données et contient les processus qui assurent que les données importantes d'entreprise sont gérées d'une manière appropriée	[Paschalidi, 2015]
La gouvernance de données tant que la partie importante de la gestion de la qualité des données adresse le cadre spécifique pour les droits de décision et les responsabilités dans l'utilisation des données.	[Wende, 2007]
Un système de droits de décision et de responsabilités pour les processus liés à l'information, exécuté selon des modèles convenus qui décrivent qui peut prendre quelles actions avec quelles informations, et quand, dans quelles circonstances, et en utilisant quelles méthodes.	[Thomas, 2006]
La gouvernance des données fait référence à l'exercice de l'autorité et du contrôle sur la gestion des données afin d'augmenter la valeur des données et de minimiser les coûts et les risques liés aux données	[Abraham et al., 2019]

TABLE 2.1 – Les définitions de la gouvernance de données

manque de la confidentialité et la qualités des données, notamment pour les données sensibles. La mise en œuvre d'un cadre juridique pour contrôler l'exactitude des données pourrait d'une part superviser la gestion du cycle de vie des données et minimiser les risques et les coûts liés à la conservation des données ingérables d'autre part. Selon IBM la gouvernance de données est la stratégie globale qui permet d'assurer l'intégrité et la sécurité de données dans les entreprise. IBM met en évidence l'exigence de la gouvernance de données dans l'environnement de Big Data comme suite

" Face à l'accumulation exponentielle de nouvelles données, concevoir une architecture de données devient une problématique importante pour les entreprises afin de gérer les données, les intégrer et les rendre disponibles dans toute l'entreprise. Cette intégration des données a un impact inéluctable sur les flux de travail et la prise de décision des différentes équipes. Par conséquent,

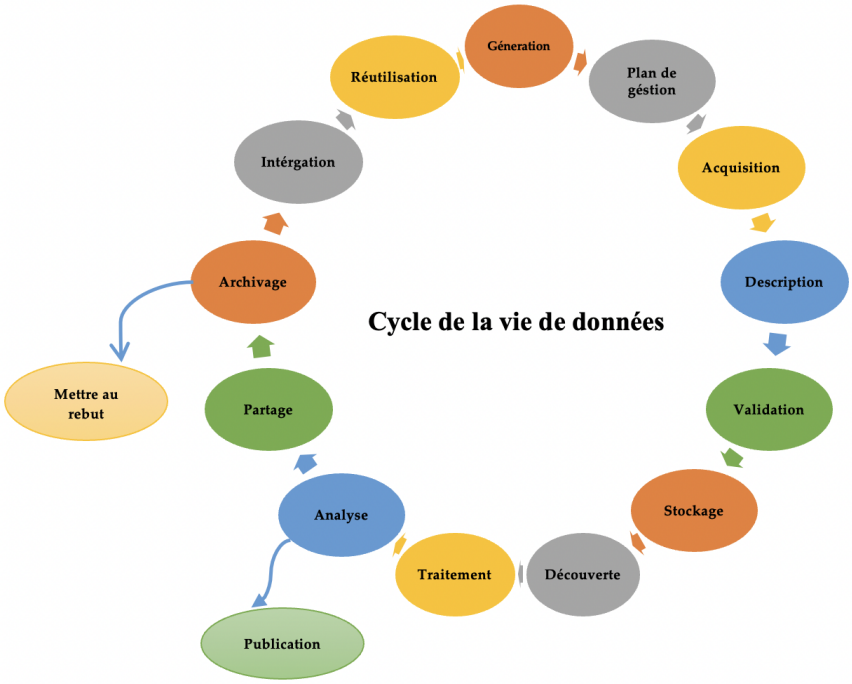


FIGURE 2.15 – Cycle de la vie de données dans les systèmes d'information

la gouvernance des données est définie comme un élément essentiel à la stratégie globale d'une organisation en matière de gestion des données [26]*."*

La gestion du cycle de vie de données est un levier pratique dans le domaine de l'intégration des données qui surveille la donnée tout au long de piste de vie de données à partir du moment où ils sont générés et les saisis dans le système jusqu'à ils sont archivés ou mis au rebut du cycle de la vie. Le cycle de la vie de données consistent tous les phases importantes liées d'ontologies de données dans les systèmes d'information qui est résumé dans la figure 2.15.

La première étape du cycle de vie des données commence par la production de données à partir de sources distribuées. Les données sont ensuite mises en correspondance avec un plan de gestion de données qui identifie les méthodes d'obtention, de stockage, de modification, de réutilisation, d'accès et d'analyse des données. Dans les étapes de description et validation, le plan de gestion de métadonnées joue un rôle important de sorte que justifie la preservation de données saisi dans les systèmes de stockage. Dans les étapes de description et de validation, le programme de gestion des métadonnées joue un rôle important pour justifier la conservation des données saisies dans les systèmes de stockage. Ensuite les données sont découvertes, traités, analyses, et publiées selon les attentes et les requêtes des utilisateurs en respectant les règles prédéfinies pour l'accessibilité et la manipulation de données. L'étape décisive de cycle de vie de données est le compromis pour garder les données valables et mettre au rebut les données inutiles ou ingérables qui est dirigée par les principes de gouvernance de données. En revanche, la gouvernance de

[26] https://www.ibm.com

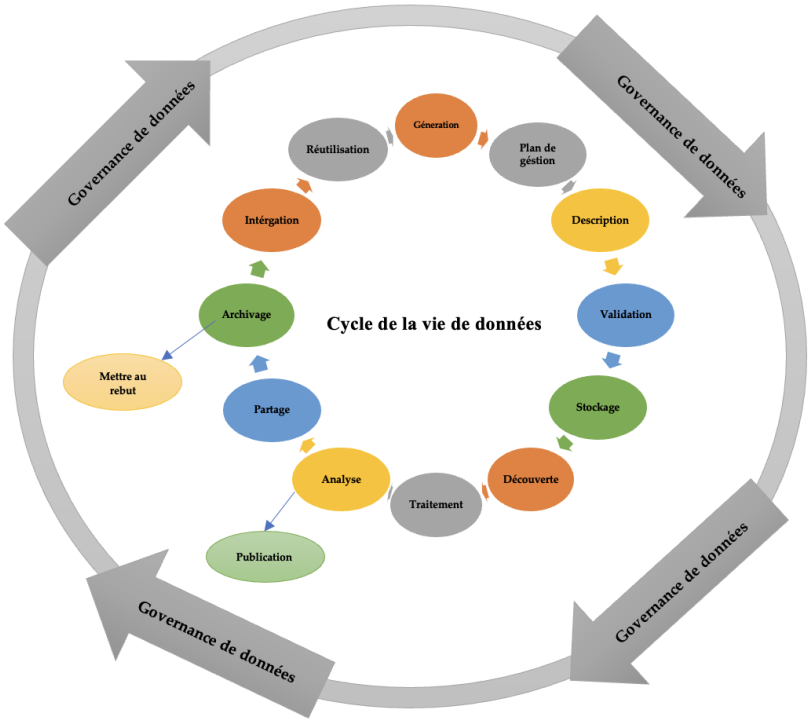


FIGURE 2.16 – Cycle de la vie de données dans les systèmes d'information en présence de la gouvernance de données

données devrait couvrir et inspecter toutes les étapes de cycle de la vie de données afin que rassurent l'intégrité et la sécurité de données tout au long de sa vie et garantie la trouvabilité, l'accessibilité, l'interopérabilité, et la réutilisabilité de données (les données FAIR [27]) au sein des systèmes d'information. Par conséquent, la gouvernance des données a des responsabilités importantes dans la gestion du cycle de vie des données qui devrait être effectuée tout au long de ce cycle afin d'éviter la redondance de données inutiles et sans valeurs dans chaque étape.

En rapport à l'importance de la gouvernance de données dans la gestion de données, on pourrait modifier la figure 2.15 en mettant l'accent sur l'existence de principes de la gouvernance de données tout au long du cycle de vie de données comme suit :

Le cadre de la gouvernance de données emploie plusieurs éléments tels que les instruments d'évaluation, les politiques, les métriques, les mesures, et les technologies pour effectuer la vérification équitable dans les systèmes d'information. En outre, une mise en œuvre efficace de la gouvernance de la données pourrait engendrer les avantages précieux pour les organisations. Ces avantages sont dressés comme ci-dessous :

- Vérifier la mise en œuvre de programme de la gestion des données ;
- Améliorer la fiabilité des données et les avantages concurrentiels pour les entreprises ;
- Réduire et détecter les risques de processus frauduleux ;

[27] https://www.go-fair.org/fair-principles/

- Réduire le risque de marécage de données ;
- Assurer la qualité, la sécurité, l'autorité, l'intégrité ; l'accessibilité, la cohérence et l'interfaçabilité de données ;
- Surveiller la mise en place la protection de données notamment pour les données sensibles et confidentielles ;
- Gérer la valeur et les coûts de données avec compromis entre les risques et les avantages [Saed et al., 2018].

Étant donné que l'importance de la gouvernance de données est indispensable pour les systèmes d'information et ses ensembles, l'implémentation de cadre de contrôle de données dans les lacs de données devient une issue problématique dans le domaine de recherches des systèmes de gestion de données massifs. Pour cette raison, le nombre d'études proposant de nouveaux outils et stratégies de gouvernance des données est remarquablement augmenté. En conséquence, une gérance complète pour une gestion et une gouvernance efficace des données est proposée par [Plotkin, 2020] afin de mise en place une ligne directrice des métriques et les mesures pratiques. D'autre part, [Paschalidi, 2015] a indiqué la gouvernance de données comme un enjeu décisif en l'environnement de données massives et a développé les cadres conceptuels de la gouvernance de données afin d'éviter un lac de données être transformée en marécage de données. En outre, [Madera and Laurent, 2016] a mis accent sur l'évolution de systèmes d'aide à décision sous l'influence de la gouvernance de données et ont prouvé que lac de données est un projet de la gouvernance de données plutôt qu'un système d'analyse de données. En complétant les travaux effectués sur l'exigence de la mise en place de gouvernance de données, dans les chapitres suivants nous allons nous appuyer sur les méthodes interdisciplinaires qui sont dérivées depuis des analogies de systèmes connus, telles que systèmes logistiques ou les systèmes de lacs naturels, afin de présenter un cadre inédit pour gouverner les données.

2.4.3 La gravité des données

Le phénomène "gravité des données" est inventé en 2010 grâce aux travaux de McCrory [28] dans l'environnement de Big data. Selon d'analogie de McCrory, les données sont considérées comme les planètes qui possèdent la gravité et la puissance de cette gravité à la relation directe avec leur densité. La gravité de données est définie par analogie à la gravité des planètes : à mesure que le volume de données est accumulé, sa gravité sera augmentée. À la suite de l'accroissement de volume de données massives dans les systèmes d'informations actuels, la gravité de données impose la force aux puissance de traitement et services qui s'approchent d'environnement de ces données denses. Mccrory a indiqué que *la latence* et *le débit* sont deux agents principaux qui jouent les rôles importants dans l'accentuation d'attraction gravitationnelle entre les données, les services et la puissance de traitement. En revanche, la gravité de données a un impact fort sur les possibilités de déplacement de données : à mesure que le volume de données est accumulé, la gravite de données rend difficile les déplacements de données d'un système à un autre en matière du temps, de logistique et de coût [Walker and Alrehamy, 2015].

[28] https://datagravitas.com/2010/12/07/data-gravity-in-the-clouds/

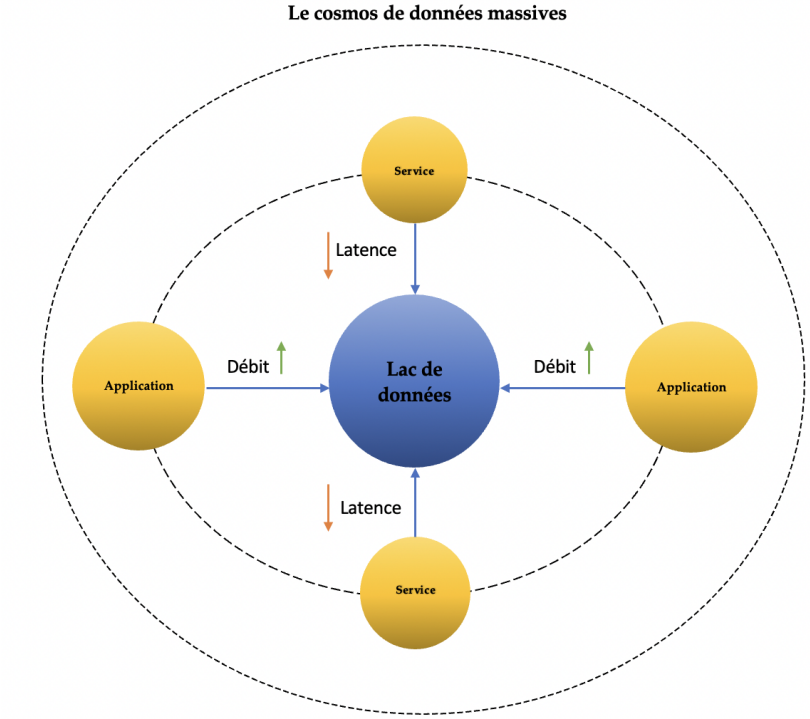


FIGURE 2.17 – La gravité de données

La croissance exponentielle de données mise en évidence les enjeux problématiques liés aux volumes de données dans les lacs de données tels que le stockage, l'analyse et la plus importante, le déplacement de données volumineuses. En effet, la gravité de données est un sujet plus discutable, du point de vue du mapping et de la migration de données, pour les systèmes de stockage de données élevées qui saisissent énormément de données. En nous basant sur cette vue, un lac de données est identifié comme une planète [29] dans le cosmos de données massives qui attire les autres corps de données(corps céleste en physique) tel que les services et les applications. Au fur et à mesure que la gravité s'élargit, de plus grands volumes de corps de données avec une vitesse plus élevée seront récupérés par lac de données. Un lac de données avec une grande gravité est un référentiel centralisé attrayant pour les organisations car il contient une variété de données brutes alors qu'il est compliqué d'immigrer les données vers d'autres systèmes.

La formule d'indice de gravité des données de référence de *Digital Realty Market Intelligente and Analytics en septembre 2020* [30] qui est basée sur l'equation de McCrory, indique les facteurs influents de la gravité de données qui sont formulées comme :

$$\frac{(DM \times DA \times BW)}{L^2} \tag{2.1}$$

[29] https://www.talend.com/resources/what-is-data-gravity/

[30] https://go2.digitalrealty.com/rs/087-YZJ-646/images/Report_Digital_Realty_2009Data_Gravity_Index_Report.pdf?_ga=2.79908436.2107548876.1601170016-2036818952.1571770991

où, DM est pris comme la masse de données (les données stockées ou accumulées), DA est défini comme l'activité de données, BW est la bande passante, et L est considéré la latence qui a l'effet inverse sur l'intensité de la sévérité (une latence plus élevée représente une gravité plus faible).

Le phénomène de gravité de données sont moins mené par les recherches scientifiques dans les domaines de données massives, tandis que l'impact de données hétérogènes et volumineux sur la fonctionnalité de systèmes intérieurs, comme lac de données, et l'influence de ces données sur les environnements extérieurs, comme les autres systèmes de gestion de données, sont peine à négliger. Par contre, [Laurent et al., 2020, Madera and Laurent, 2016] ont adressé l'impact inévitable de la gravité de données sur l'architecture de lac de données qui est un système de stockage de données massives. Ils ont indiqué que la sensibilité de données, liées aux données personnelles, ainsi que le coût de duplication des sources de données sont d'autres agents importants qui pourront influencer la gravité de données plus que l'agent de volume. Étant donné que la gravité des données et l'infrastructure du lac de données ont une relation réciproque, la conception de l'architecture des données devient plus importante afin de tirer parti de ce phénomène bouleversant. Une architecture optimisée de lac de données pourrait fédérer et organiser les données hétérogènes d'une manière équitable avec un compromise entre les avantages et les inconvénients de la gravité de données pour le développement de la fonctionnalité des systèmes gestion de données massives.

2.4.4 Utilisateurs

Les lacs de données adressent des données extrêmement massives et sous des formats bruts qui posent la question de manipulation par des utilisateurs non avertis. En général, le lac de données est un système de gestion de données avancé, hybride et compliqué qui est forcément dédié aux utilisateurs professionnels tels que les scientifiques de données dont les compétences leur permettent de traiter et exploiter les données natives. De plus, cette complexité d'accès a remis en question la popularité de lac de données en tant que système de gestion de données pratique pour les organisations traditionnelles et l'a rendu inaccessible en tant que système de stockage de données personnelles. Pour toutes ces raisons, plusieurs architectures et modèles sont proposés pour le lac de données afin d'identifier, d'indexer, de surveiller et de gérer les données de manière compréhensible par tous types d'utilisateurs et de rendre le lac de données plus efficace et commode pour les données personnelles.

En raison de ce besoin, [Walker and Alrehamy, 2015] a défini un modèle pour le lac de données qui renforce l'importance de la gestion des métadonnées dans le lac de données, ce qui permet au lac de données de devenir un environnement approprié pour les données personnelles. Selon [Walker and Alrehamy, 2015] quatre responsabilités (utilisateurs) principales sont classées pour servir un lac de données personnelles telles que ***un fournisseur de données*** qui génère, prépare et transforme les données selon des schémas interrogeables, ***un consommateur de données*** qui exploite les données avec des outils d'analyse de manière professionnelle, ***un fournisseur de stockage de données*** qui prépare les dispositions pour la mise en œuvre du système de métadonnées pour les données stockées, et ***un utilisateur personnel*** qui possède des données ainsi qui génère et consomme des données de manière non-avertie.

Afin de globaliser le lac de données pour divers utilisateurs et en sachant que la gravité des données rend les systèmes de stockage de données énormes plus réactifs, plusieurs mesures devront être envisagées en ce qui concerne la convivialité du tel système stockage de données :

- La démocratisation de l'accès pour plusieurs niveaux de consommateurs de données [Madera, 2018] ;
- Les fragments de données personnelles selon la maturité et la structure de données ;
- La gestion de la gravité de donnees afin de profiter des avantages de données massives pour les démarches analytiques multiobjectifs ;
- La mise en oeuvre les dispositions pour augmenter la comptabilité des services d'API dans l'infrastructure de lac de données ;
- La fourniture des moyens professionnels liés de sémantiques de données qui facilitent l'interprétation et la convivialité de données tous les types des utilisateurs ;
- La plateforme compatible et securisée à traitement facile des requêtes (L'interface interrogeable facile) ;
- L'amélioration de la gestion de métadonnées conciliables ;
- La mise en place de services pour la protection des données sensibles et la confidentialité des données personnelles.

2.5 Implémentation de lac de données

Dans les sections précédentes, nous avons abordé l'exigence et les fonctionnalités du lac de données comme un système complexé d'accueillir et de la gestion de données massives. Les problèmes abordés dans les sections précédentes sont des problèmes liés à la gestion des lacs de données. Pourtant, les problèmes correspondants de mise en œuvre physique et technique restent encore des défis importants à relever. Étant donné que le lac de données occupe tous les processus liés au cycle de la vie des données, un environnement (plate-forme) hybride qui favorise tous les dispositifs nécessaires et fait fonctionner simultanément plusieurs applications sont primordiales afin de concevoir une architecture multitâches et complète. Pour atteindre cet objectif, plusieurs outils sont proposés pour le lancement d'un écosystème favorable qui soit capable de fournir les techniques diverses afin de répondre les attends informatiques. Parmi les plates-formes disponibles de l'implémentation du lac de données, Hadoop et ses distributions connues telle ques **Cloudera**, **Hortonworks**, **MapR** et **Apache Hadoop**, sont plus remarquables dans l'environnement de Big data qui offrent les outils pratiques et complets pour la gestion de données tout au long de cycle de la vie [White, 2012]. Hadoop est un cadre répandu de big data qui est innové grâce au développement d'un projet de moteur de recherche web open source. L'environnement d'Hadoop contient de plusieurs technologies qui misent en oeuvre des fonctionnalités appropriées concernant de la valorisation de données massives. Parmi les technologies existantes, **HDFS** (Hadoop distributed file system) qui organise de stockage de données et **MapReduce** qui realise les processus de l'analyse et traitement, sont les composants plus importants et célèbres de l'environnement d'Hadoop.

Suite à la croissance des taux d'emplois des lacs de données dans les organisations en tant qu'élément important du système d'aide à la décision, la sélection de technologies appropriées et pratiques est une action décisive pour la mise en œuvre réussie des lacs de données. Pour cette raison, la concurrence pour offrir des plateformes abordables et efficaces qui répondent à toutes les attentes des utilisateurs s'est accrue ces dernières années. En revanche, la prise des décisions pour choisir des bons technologies d'infrastructure de lac de données, est une décision multicritère qui devrait être effectuée par le compromise entre les avantages et les inconvénients de chaque application.

Comme nous l'avons expliqué, la variété de l'architecture des lacs de données permet aux organisations de choisir une infrastructure adaptée en fonction de leurs besoins. Cependant, afin d'opter les technologies efficaces pour lancer l'architecture choisie, il est préférable de mettre en évidence les caractéristiques suivantes :

- La capacité de gestion d'ingestion de données depuis des sources externes ;
- La capacité d'assurer l'intégration et la lignée des systèmes de données ;
- La prédisposition de minimiser les coûts initiaux, avals et amonts de lac de données ainsi que déduire le temps de service aux utilisateurs ;
- La facilité de déploiement et d'emploi l'infrastructure prédéfinie ;
- L'applicabilité du système de métadonnées et l'indexation ;
- La faculté de mise en oeuvre des dispositions de gestion de cycle de vie de données ;
- L'efficacité, la fiabilité, la gérabilité, et l'évolutivité des applications utilisées ;
- Les facultés pour la surveillance et la gouvernance de données ;
- L'interface de communication des clients ;
- La haute compatibilité des APIs associées ;
- L'aptitude à garantir la sécurité et la qualité de données ;
- L'infrastructure appropriée pour prendre en charge la gravité de données.

Généralement, la configuration technique du lac de données est une problématique qui dépend de la fonctionnalité, l'application et le type de l'architecture de lac. Par conséquent, l'architecture de lac de données est dressée dans la taxinomie ci-dessous selon le but d'utilisation et les infrastructures applicables pour le stockage et le traitement de données [Zagan and Danubianu, 2021] :

- **Lac de données On-premise :** Le lac de données On-premise est un lac de données traditionnel qui occupe toutes les dispositions matérielles et logicielles de l'infrastructure du lac de données. La prise en charge complète de la configuration et de l'administration de l'architecture du lac de données entraîne des problèmes tels que la perte de la valorisation de données,le manque de spécialistes, et des coûts d'investissement élevés. **Apache Hadoop** et ses sous-projets sont des technologies compatibles et peu coûteux pour lancer ce type d'architecture de lac de données.

- **Lac de données en Cloud :** Selon la définition de [Mell and Grance, 2011], *" l'informatique en nuage*[31] *est un modèle permettant un accès au réseau convenable, à la demande, et omniprésent à un bassin commun de ressources informatiques configurables tels que les réseaux, les serveurs, les stockages, les applications, et les services qui pourrait être rapidement provisionné avec un effort de gestion ou une interaction du fournisseur de services minimal."* Le lac de données en Cloud est une nouvelle génération innovante de ce dépôt de données qui est en train d'évoluer grâce au développement technologique et de devenir une architecture adaptée aux besoins des grandes entreprises. En se basant sur la définition du modèle Cloud, le lac de données en Cloud apporte un environnement commun, rentable, convivial, flexible, évolutif, avec une capacité de stockage élevée. Cette solution permet ainsi de réduire les silos de données, les contraintes inhérentes à leurs accès, aux risques de leur perte, et les coûts d'investissement, grâce aux outils d'analyse déployés à grande échelle et les stratégies de gestion applicables. On pourrait citer *Amazon Web Services (AWS)*, *Google's Cloud Platform (GCP)*, and *Microsoft's Azure*, *IBM bleu cloud* et les versions open-sources tels que comme *OpenStack*, *OpenNebula*, *Eucalyptus*, *Nimbus* comme les fournisseur connus pour déployer un lac de données en Cloud [Sefraoui et al., 2012].

- **Lac de données Hybride :** Bien que les deux architectures on-premise et Cloud comportent des avantages comme des inconvénients, la prise de décision quant au choix de l'architecture appropriée (on-permise ou Cloud) dépend de l'infrastructure et des possibilités potentielles de l'organisation. Par contre, on pourrait imaginer la mise en œuvre d'une architecture hybride qui offre un contrôle de gouvernance et de sécurité de données sensibles plus strict en utilisant une plate-forme sur site (on-premise) et en fournissant en même temps un environnement évolutif et agile pour le stockage et le traitement des données grâce aux plates-formes cloud[32].

- **Lac de données en Multi-Cloud :** Le lac de données Multi-Cloud contient au moins deux fournisseurs de services Cloud qui tirent parti des capacités et des applicabilités de différentes infrastructures Cloud et minimisent les risques et les problèmes associés à une seule technologie cloud. Afin de renforcer l'efficacité de cette architecture, les fournisseurs de services cloud les plus compatibles tels que AWS, Azure, Google, IBM, ou Oracle pourront été associés pour installer une architecture de lac de données Multi-Cloud.

Bien qu'il existe plusieurs architectures pour lancer un lac de données propre, on ne pourrait pas lister les critères exacts d'une architecture idéale. Une plate-forme efficace ou optimale est adoptée en fonction des objectifs et des exigences de chaque organisation pour le déploiement d'un système de gestion des données. D'autre part, les marchés technologiques dans le domaine de la mise en œuvre de lacs de données rentables se développent et les entreprises dynamisent des produits innovants et parfois optimisés pour faire appel aux intérêts des organisations. Pour cette raison, nous adoptons une architecture générale telle que **Lambda** (une architecture hybride) qui couvre toutes les phases inhérentes au cycle de vie des données afin d'introduire et de promouvoir les technologies existantes pour réaliser un lac de données.

L'architecture du lac de données contient généralement les zones et les couches décrites dans les sections suivantes.

[31] Cloud Computing
[32] https://www.zaloni.com/resources/blog/data-lake-cloud-hybrid-on-premises/

2.5.1 La Zone Ingestion

La zone d'ingestion, ou la zone de collecte, est la zone la plus proche des sources de données et joue un rôle important dans la capture et l'intégrité des données. Les technologies qui sont souvent utilisées pour implémenter cette couche doivent avoir une compatibilité et une flexibilité suffisantes pour saisir tous les types de données à partir de toutes les ressources ciblées. Selon l'architecture technique choisie (On-premise, Cloud, Hybride ou Multi-Cloud), les types de données envisagées (structurées, non structurées et semi-structurées), et les infrastructures des sources de données (IOT, la base de données relationnelle, réseaux sociaux, données de capteurs...), il existe beaucoup de solutions. Il faut alors arbitrer entre leurs avantages et inconvénients pour choisir les technologies d'ingestion.

Les technologies suivantes sont souvent utilisées selon les infrastructures définies :

- **Spring XD :** Un service open-source pouvant être exécuté dans un cluster Apache Spark ou Hadoop pour ingérer des données en temps réel ou par lots (Batch)
- **Apache Flume :** Flume est une plate-forme d'ingestion d'Apache pour transporter de gros volumes de données de fichier de logs qui sont générés à partir des serveurs de web, vers l'environnement Hadoop. Apache Flume ingère les données massives en temps réel, en diffusion en continu, et en micro-lots au système de stockage centralisé.
- **Apache Kafka :** Un logiciel open source de messagerie d'Apache qui ingère et centralise les données de diffusion en continu d'une manière fiable et rapid.
- **Fluentd :** Un systeme open-source de la collection de donnees en flux.
- **Spark streaming :** Un extention de une extension de l'API Spark pour mise à l'échelle les données de diffusion en continu et ingéré les données depuis plusieurs sources.
- **Amazon Kinesis :** Un service d'ingestion évolutif et rapide pour capturer des données en temps réel. Amazon Kinesis offre les outils et les capacités pour préparer les données avant de les charger, ce qui permet une ingestion plus économique et sécurisée par rapport à d'autres technologies.
- **Azure Event Hubs :** Un service d'ingestion de données en temps réel ouvert et évolutif qui prend en charge de gros volumes de données par seconde en provenance de diverses sources.
- **Apache Sqoop :** Un outil efficace connu pour importer des données structurées à partir des bases de données relationnelles vers l'écosystème Hadoop.
- **WebHDFS :** Une plate-forme pour capturer des données structurées et non structurées dans l'environnement Hadoop.
- **Datastage :** Une plate-forme évolutive d'ingestion d'IBM qui saisit tous les types de données en temps réel dans des environnements hybrides ou multi-cloud

2.5.2 La Zone Stockage

Le stockage et la centralisation des données sont les objectifs principaux de lac de données qui devront être réalisés de manière agile, sécurisée et optimisée. Les coûts de maintenance des données associés aux services de support matériel et logiciel et le risque de perturbation des données nécessitent le choix de technologies de stockage durables pour protéger la fiabilité et la disponibilité des données. Les services suivants sont les technologies les plus courantes dans l'environnement Big Data pour le stockage de données :

- **HDFS (Hadoop Distributed File System) :** Une solution pour stocker les nombreux jeux de données dans un environnement avec capacité limitée est le regroupement de données sur les réseaux de machines qui est effectué grâce aux systèmes de fichiers distribués [White, 2012]. HDFS est un système de fichiers distribué de Hadoop qui gère le stockage des pétaoctets de données massifs de diffusion en continu avec du matériel à faible coût. HDFS est la technologie la plus connue, la plus utile et la plus rentable pour la zone de stockage d'architecture de lac de données.
- **HBase :** HBase est une base de données open source orientée colonnes qui prend en charge de très grandes tables avec des milliards de lignes et de colonnes et permet un accès à faible latence pour lire et écrire des données en temps réel.
- **MapR-DB :** MapR-DB est une base de données NoSql de MapR qui prend en charge une variété d'applications pour analyser des données en temps réel à grande échelle en réduisant les délais d'accès.
- **Amazon S3 :** Un service Web pour le stockage d'objets sur internet proposé par Amazon et suggéré pour configurer le stockage de zone de l'architecture de lac de données en Cloud.

2.5.3 La Zone Traitement

La taxonomie des technologies de traitement des données est établie selon les méthodologies utilisées telles que ETL ou ELT ou selon les types d'intégration de données telles que la saisie en batch, en temps réel, et en quasi-réel. En fonction de l'architecture envisagée pour le lac de données, on peut choisir la technologie pertinente ou une combinaison de plusieurs services. Généralement, les services suivants sont populaires pour la phase de traitement des données :

- **MapReduce :** MapReduce est un modèle de programmation de traitement de données qui effectue les calculs parallèles ou distribués pour les données en lots avec deux fonctions importantes map et Reduce.
- **Spark :** Spark est une plateforme open source d'Apache qui traite les données en temps réel. Spark a été remplacé par MapReduce dans plusieurs architectures de lac de données car il analyse les données de manière plus minimisée et plus rapide avec un calcul à faible latence.
- **Apache Hive :** Apache Hive est une technologie d'entrepôt de données distribuée open-source qui permet de trouver et de manipuler des pétaoctets de données provenant d'eco-syetem Hadoop avec des requêtes en langage SQL.

- **Apache Storm :** Apache Storm est un système open source et évolutif pour le traitement de diffusion en continu en temps réel qui calcule les données à une vitesse supérieure à celle d'Apache Spark.

2.5.4 La Zone Accès et Visualisation

La zone d'accès et de visualisation est une étape décisive dans la valorisation des données. Mettre l'accent sur le choix des technologies et des outils qui facilitent la procédure de l'exploitation et l'exploration des données, est un enjeux considérable. Pour les entreprises et les organisations qui souhaitent tirer parti de leurs atouts de données pour les processus et activités de prise de décision, la fiabilité, la vitesse, la précision, l'exactitude et le coût des technologies choisies pour l'infrastructure du lac de données, sont les facteurs importants à souligner. Pour cette raison, les marchés des outils de visualisation et de découverte de données sont saturés de variétés de produits dont les valeurs concurrentielles sont la convivialité et la haute qualité de service pour les utilisateurs. Les outils suivants sont normalement employés dans la zone de la visualisation et l'analyse de données.

- **Qlik :** Un service pour la fouille de données, l'analyse, et la visualisation qui est basée sur les techniques d'intelligence artificielle.
- **Radoop :** Un outil convivial de l'intelligence d'entreprise qui analyse et visualiser les grandes quantités de données sur les clusters d'Hadoop d'une manière simple grâce aux procède flexibles.
- **Azure PowerBI :** Une plate-forme analytique qui analyse à grande échelle des données en temps réel ou par lots avec des outils d'intelligence artificielle avancés à grande vitesse.
- **AWS Athena :** Un service de requêtes de SQL simple pour la visualisation et l'accès aux données dans le service cloud d'Amazon S3.

2.6 Résumé

Dans ce chapitre, nous avons vu que l'évolution des systèmes de gestion des données entraîne une infrastructure flexible et agile pour la valorisation des données appelée 'Lac de données'. Le lac de données, tel qu'un référentiel centralisé de données brutes, prépare un environnement dynamique et équitable pour l'exploitation et l'exploration des données. Par conséquent, l'architecte de cette structure influence les performances et le choix des plate-forme de mise en œuvre. Un lac de données efficace est obtenu grâce à une infrastructure optimisée ainsi que des technologies équitables pour le déployer d'une manière harmonique.

Chapitre 3

Chaîne d'approvisionnement et ses stratégies gestionnaires

"La chaîne d'approvisionnement est comme la nature, elle est tout autour de nous"

– Dave Waters

3.1 Introduction

Le terme "système'" est un terme omniprésent. Selon la terminologie générale de systèmes qui est présentée par *International conceal on systems engineering* [1]

Un système est un arrangement de parties ou d'éléments qui, ensemble, présentent un comportement ou une signification que les constituants individuels n'ont pas.

Un système est une réunion d'ensembles de composants qui se coordonnent intégralement pour atteindre un ou plusieurs objectifs globaux garantissant l'existence du système. on pourrait trouver les exemples pertinents dans tous les domaines disciplinaires tels que, l'écosystème (un système naturel), le système nerveux (un système biologique), le système solaire (un système planétaire), le système électrique (système d'ingénierie), une chaîne d'approvisionnement (un système logistique ou gestionnaire), et notamment le système d'information (un système informatique). Dans le chapitre précédent, nous avons indiqué que lac de données est un système stockage et gestion de données qui pourrait jouer un rôle important dans la valorisation de données massives. Ensuite, nous avons mis en évidence l'exigence de l'architecture de ce système informatique qui influence la fonctionnalité et la valeur concurrentielle. Pour cette raison, nous visons à proposer une nouvelle démarche en nous basant sur les outils gestionnaires des systèmes logistiques pour optimiser et augmenter la performance des lac de données. Par conséquent, la chaîne d'approvisionnement est proposée comme un système logistique connu qui est une infrastructure susceptible d'imiter les outils et stratégies de gestion pour gérer l'architecture du lac de données.

Dans ce chapitre, nous allons aborder la définition de chaîne d'approvisionnement et ses stratégies pratiques pour la gestion du réseau logistique. Par conséquent, nous nous appuierons sur les enjeux suivants pour atteindre nos objectifs :

- La définition de chaîne d'approvisionnement ;
- La gestion de chaîne d'approvisionnement ;
- La Conception de réseau de chaîne d'approvisionnement.

[1] https://www.incose.org/about-systems-engineering/system-and-se-definition/general-system-definition

3.2 Chaîne d'approvisionnement

Après la révolution industrielle du 19ème siècle, la manière d'harmoniser l'offre et la demande allait beaucoup changer. L'approvisionnement par la production locale se transforme en un réseau global de plusieurs entités pour répondre rapidement aux besoins croissants des consommateurs. Les entreprises isolées deviennent obligées de coopérer avec d'autres partenaires de manière collaborative pour couvrir les attentes du marché et développer leur rentabilité. Depuis 1980, les définitions appropriées de la chaîne d'approvisionnement ont vu le jour où de nombreuses entreprises et organisations ont constaté que la chaîne d'entités intégrées est plutôt efficace qu'une entreprise décentralisée [Lummus and Vokurka, 1999].

La définition de la chaîne d'approvisionnement est enveloppée dans son nom, cela signifie qu'une chaîne de nombreux fournisseurs, fabricants, distributeurs et détaillants s'engage ensemble pour fournir un produit ou un service aux clients. Selon [Martin, 1998], la chaîne d'approvisionnement se compose de plusieurs acteurs en amont et en aval dans lesquels chaque acteur correspond à un processus ou une responsabilité spécifique afin d'ajouter de la valeur en termes de produit ou de service et d'atteindre les objectifs globaux prédéfinis. De plus, la chaîne d'approvisionnement est définie comme un système intégré qui transforme les matières premières en marchandises de valeur grâce à des membres hiérarchiques qui travaillent ensemble à plusieurs niveaux [Beamon, 1998, Janvier-James, 2012].

Sur la base des nombreuses définitions, il a été constaté que la base de la chaîne d'approvisionnement est la gestion de la logistique. En effet, la chaîne d'approvisionnement contient un flux logistique dominant tout au long du cycle de vie des produits qui sont élaborés à partir d'entités cohérentes afin de répondre aux attentes des clients [Panayides, 2006, Ayers, 2000]. Par conséquent, deux termes importants, liés à nos travaux actuels, se sont manifestés dans cette définition, *le système logistique* et *le cycle de vie du produit*. L'état des flux de produits au sein des structures de la chaîne et la qualité de la performance logistique déployée sont des indicateurs remarquables qui évaluent l'efficacité de la chaîne d'approvisionnement. Ces indicateurs seront mis en évidence dans l'estimation de la performance du lac de données en tant que système de stockage de données où nous définirons le lac de données comme une chaîne d'approvisionnement à la différence que les données son assimilées comme les produit et le cycle de vie du produit donne sa place au cycle de vie des données. En règle générale, une chaîne d'approvisionnement organisée est construite à partir de deux flux principaux, le flux vers l'avant qui englobe le flux de produits des fournisseurs aux consommateurs et le flux vers l'arrière qui contient le flux d'informations des clients ou des marchés [Min and Zhou, 2002]. Une chaîne d'approvisionnement boucle fermée [2] qui contient a la fois des deux flux vers l'avant et vers l'arrière, ajoute une importance aux rétroactions de clients afin d'augmenter la qualité des services ou produits ainsi que la gestion logistique et cycle de la vie des matériaux et des marchandises.

Contrairement à une définition simple d'une chaîne d'approvisionnement, la mise en œuvre et la conception d'un réseau propre nécessitent des prises de décision complexes aux niveaux stratégique, tactique et pratique. Établir la bonne coordination entre les partenariats de la chaîne afin d'augmenter la rentabilité ainsi que d'optimiser la fonctionnalité et de minimiser les risques et les coûts pertinents, nécessitent des stratégies de gestion globales pour la chaîne d'approvisionnement. Les principaux enjeux dans le domaine de la chaîne d'approvisionnement sont d'une part

[2] Closed-loop supply chain

de choisir les stratégies appropriées pour gérer les procédures globales et locales et d'autre part de concevoir une architecture optimisée capable de mettre en œuvre les stratégies prédéfinies. Dans les chapitres suivants, nous aborderons en détail ces deux enjeux importants qui constituent les fondements de l'analogie entre la chaîne d'approvisionnement et le lac de données et les outils que nous utiliserons pour optimiser le système de stockage de données.

3.3 Gestion de chaîne d'approvisionnement

Une chaîne d'approvisionnement optimisée et rentable n'est pas le résultat des efforts d'une seule entreprise ou d'un seul participant, mais le résultat de la coopération de tous les partenaires sous la direction de stratégies planifiées. En effet, tous les systèmes intégrés qu'ils soient naturels, biologiques ou logistiques ont besoin d'une instruction organisée afin que mise en oeuvre les régulations pour unifier tous les entités vers les objectives global de systèmes. On parle de la gestion de chaîne d'approvisionnement [3]. Selon *Council of Supply Chain Management Professionals (CSCMP)* :

"La gestion de la chaîne d'approvisionnement contient une série de stratégies clés qu'elle doit exécuter de manière efficace et opportune afin de répondre aux besoins des clients. [4] ***"***

Concernant la définition ci-dessus, la gestion de la chaîne d'approvisionnement est un programme complet pour gérer les relations intra et inter-entreprises afin de réaliser une synergie inter-réseau pour intégrer les membres au sein de la chaîne d'approvisionnement [Lambert and Cooper, 2000]. En outre, [Mentzer et al., 2001] a proposé une taxonomie de la définition de la gestion de la chaîne d'approvisionnement qui se divise en trois grandes catégories selon les études menées dans ce domaine :

- ***La gestion de la chaîne d'approvisionnement en tant que philosophie de la gestion :*** dans cette définition, le réseau de la chaîne d'approvisionnement est considéré comme un tout unifié qui doit être synchronisé grâce aux stratégies orientées ;
- ***La gestion de la chaîne d'approvisionnement en tant qu'ensembles de réglementations pour mettre en œuvre une philosophie de gestion :*** dans cette définition, la gestion de la chaîne d'approvisionnement contient des réglementations définitives telles que l'intégration des membres et des processus, le partage mutuel des informations, des risques et des profits, la détermination d'objectifs communs et l'organisation d'efforts coopératifs afin d'atteindre la satisfaction du client ;
- ***La gestion de la chaîne d'approvisionnement en tant qu'ensembles des processus de gestion :*** dans cette définition, la gestion de la chaîne d'approvisionnement est définie comme un ensemble de processus structurés et orientés pour ajouter ou créer des valeurs pour la chaîne d'approvisionnement aux travers de la fabrication de produits particuliers ou d'amélioration des services requis pour les clients ciblés.

[3] Supply chain management(SCM)
[4] https://cscmp.org

Sur la base de ces concepts, la gestion de la chaîne d'approvisionnement est définie comme un code général qui contient toutes les décisions stratégiques, tactiques et pratiques liées à la gestion des flux de produits. Cette instructions exhaustive orient tous les processus intra et inter-réseau tels que l'emplacement d'installation, la sélection des approvisionnements, le contrôle des inventaires, la planification de la distribution, la planification des itinéraires et des transports et la gestion des relations clientèles, sur la base des principes fondamentaux pour réduire les coûts supplémentaires et augmenter la rentabilité du réseau d'une manière coopérative. En 2007, [Anderson et al., 2007] a défini les sept principes importants pour la gestion de la chaîne d'approvisionnement qui devront être pris en compte pour parvenir à une chaîne productive :

- La segmentation des clients en fonction de leurs attentes ;
- L'arrangement le réseau logistique en fonction de la segmentation client ;
- L'attention aux signaux du marché et l'allocation des ressources en fonction de la demande des clients ;
- La réponse agile à la variété de la demande des clients ;
- La gestion de la sélection et de l'externalisation des fournisseurs pour réduire les coûts d'approvisionnement et de maintenance ;
- L'élaboration d'une stratégie technologique et la surveillance les flux de produits, de services et d'informations ;
- L'évaluation de la performance de la chaîne d'approvisionnement avec les métriques appropriées.

Tous les principes mentionnés ci-dessus peuvent être pris en compte pour obtenir un système de réseau intégré et flexible dans tous les domaines. Par conséquent, la gestion de la chaîne d'approvisionnement est à la fois un cadre complet pour les systèmes de motifs commerciaux et un code général inspirant pour toutes les structures qui ont des comportements systémiques. Par exemple, un système d'information qui est un système connu dans le domaine informatique, s'il était considéré comme une infrastructure systémique de fourniture d'informations ou de données, pourrait obéir à ces principes généraux. Pour aborder le sujet, nous pouvons indiquer que dans le système de gestion des données, plusieurs des principes mentionnés sont mis en évidence dans des sens similaires, tels que la segmentation des données selon des besoins des utilisateurs, l'agencement des structures et des logiciels pour des réponses agiles à des demandes spécifiques, la couverture de la variété des demandes, les choix stratégiques pour mettre en œuvre l'infrastructure économisée avec des technologies appropriées, la surveillance et la gestion du cycle de vie des données, l'évaluation des performances du système, et l'attention aux signaux de sécurité pour la protection des données.

En effet, pour mettre en œuvre les sept principes indiqués à travers les instructions du contexte de la gestion de la chaîne d'approvisionnement, nous avons besoin de stratégies à long terme et à court terme. Par exemple, pour la surveillance du flux de produits, des stratégies d'inspection de l'état des produits sont normalement appliqués, appelés gestion du cycle de vie [5]. Dans

[5]Lifecycle Management

le contexte de l'attention portée aux signaux du marché, la chaîne d'approvisionnement en boucle fermée est considérée comme une structure efficace pour gérer le flux vers l'arrière d'informations ainsi que pour mettre en œuvre des processus de recyclage ou d'élimination des produits. De plus, afin d'actualiser la chaîne d'approvisionnement par rapport aux attentes des clients d'un point de vue social, écologique et concurrentiel, des stratégies de gouvernance de la chaîne sont déployées pour atteindre une structure plus adaptée aux critères de marchés tels que la chaîne d'approvisionnement verte. En général, les stratégies de la gestion de la chaîne d'approvisionnement mobilise toutes les méthodes, modèles et principes pour lancer une chaîne systémique intégrée qui pourrait créer de la valeur pour tous les partenaires [Janvier-James, 2012]. Ces principes est orienté vers des chaînes d'approvisionnement plus agiles, résilientes, vertes et durables dans le cadre de stratégies mondiales.

Dans la section suivante, nous détaillerons les stratégies de gestion de la chaîne d'approvisionnement avec toutes ses règles, normes et réglementations au sein des chaînes d'approvisionnement qui engendrent les chaînes d'approvisionnement sous les différentes considérations comme classées ci-dessous. Par conséquent, nous puissions tirer parti de ces mécanismes pour établir les cadres d'imitation pour la gestion de la vie des données et la gouvernance des lacs de données dont nous discuterons dans le chapitre 5 ou optimiser la structure de ce système de stockage de données, ce qui sera discuté dans le chapitre 6.

3.3.1 Chaîne d'approvisionnement verte

Définition : De nos jours, les préoccupations environnementales sont devenues des enjeux concurrentiels pour les chaînes d'approvisionnement qui s'occupent une grande proportion des marchés. La chaîne d'approvisionnement verte [6] est définie comme un système logistique écoresponsable qui exécute des considérations environnementales dans toutes les activités et décisions de la chaîne d'approvisionnement du début à la fin du cycle de vie du produit - [Min and Kim, 2012, Vachon, 2007, Beamon, 1998]. Les stratégies verts englobent toutes les stratégies qui renforcent les normes écologiques pour toutes les actions au sein des réseaux telles que :

- Choix du fournisseur ;
- Concevoir des produits ou services ;
- Procédures de fabrication ;
- Emballage et stockage ;
- Transport et distribution ;
- Service de client (avant et après vente) ;
- Processus de recyclage et d'élimination [Min and Kim, 2012].

[6] Green Supply Chain

Objectifs : En général, les principaux objectifs de la promotion des chaînes d'approvisionnement vertes sont d'établir une structure rentable et économique qui met en évidence les enjeux écologiques en même temps. Selon [Vachon, 2007, Min and Kim, 2012, Beamon, 1999, Li et al., 2019] on mobilise les membres inter et intra organisationnels pour achever :

- Les collaborations vertes entre les entités de la chaîne d'approvisionnement ;
- La réduction, l'élimination ou la minimisation les effets destructeurs grâce aux actions environnementales de la chaîne d'approvisionnement ;
- Le développement des performances écologiques des organisations ;
- Les entreprises durables dans le cadre de considérations environnementales et économiques ;
- Le déclenchement de la mondialisation des activités vertes à l'intérieur et à l'extérieur de l'organisation ;
- La minimisation des coûts liée de gaspillage et la maximisation des rentabilités ;
- Les valeurs compétitives grâce aux compétences écologiques ;
- L'augmentation de la qualité des produits selon les normes définies et l'empêchement le flux de produits non standard.

Méthodes : La mise en place de chaînes d'approvisionnement vertes est une approche bi-objectifs, cela signifie que les stratégies environnementales doivent être appliquées en parallèle ou en complément des stratégies de gestion organisationnelle, pour atteindre les objectifs définis. Pour cette raison, plusieurs stratégies et normes sont étudiées afin de créer ou d'évaluer les états verts de la chaîne d'approvisionnement qui sont élaborés ci-dessous :

- La mise place de mécanisme de logistique inverse et des dispositions pour la structure de flux a l'arrière des information et des produit dans la chaîne d'approvisionnement ;
- Actualiser les savoir-faire environnementaux des membres de la chaîne d'approvisionnement avec les programmes éducatifs et les partage d'informations ;
- Concevoir les contrats de coordination verte entre les partenaire afin de les obliger à applique des mesures et protocoles écologiques au sein de leurs entreprises ;
- Promouvoir des plans de fabrications vertes en respectant les normes définies pour les produit écologiques ;
- Fonder*les collaborations environnementales* [7] entre les membres des organisations et les clients ainsi que *les surveillances environnementals* [8] pour les activités de la chaîne d'approvisionnement ;
- Déterminer les réglementation, politiques et les normes ; tels que ISO 14001 ; pour gouverner la chaîne et maîtriser les organisations contre les impacts environnementaux destructifs ;

[7] Environmental collaboration
[8] Environmental monitoring

- Définir les certificats ou attestations pour évaluer les performances environnementales des membres en amont et en aval ;
- Promouvoir les technologies et les techniques pour mobiliser tous les processus et activités des chaînes vers les principes écologiques.

3.3.2 Chaîne d'approvisionnement en boucle fermée

Définition : La chaîne d'approvisionnement en boucle fermée contient tous les processus de logistique inverse [9] et les informations pertinentes pour la récupération des produits telles que le réassemblage, la réutilisation, la remise à neuf, le recyclage, l'élimination et la phase de post-utilisation ainsi que la logistique en aval afin de concevoir un système logistique complet et de promouvoir la création de valeurs [Guide Jr and Van Wassenhove, 2009, Asif et al., 2012, Gaur et al., 2017].

Objectifs : L'objectif principal de l'établissement d'une chaîne d'approvisionnement en boucle fermée est la sensibilisation environnementale des organisations et des consommateurs en même temps. En revanche, les responsabilités des entreprises dans la mobilisation des mécanismes de récupération des valeurs des produits et informations retournés sont plus évidentes. Avec l'importance croissante de la conscience environnementale, la logistique inverse a été impliquée dans les systèmes d'approvisionnement afin de :

- Éliminer les effets des services ou produits nocifs et destructifs ;
- Accroître l'intérêt des entreprises et des consommateurs à prêter attention aux enjeux verts ainsi qu'aux enjeux économiques ;
- Économiser les réseaux logistiques grâce aux minimisations des coûts et des gaspillages ;
- Capitaliser en réalisant des bénéfices grâce à des récupérations de produits inutiles.

Méthodes : Les stratégies souvent utilisées pour la gestion de la chaîne d'approvisionnement en boucle fermée se concentrent sur la conception de réseaux avec des canaux de logistique inverse ainsi que sur les législations pour la mise en œuvre de stratégies prédéfinies [Gaur et al., 2017]. Les programmes étudiés dans les cadres de la chaîne d'approvisionnement en boucle fermée sont classés ci-dessous :

- Établir et concevoir les canaux inverses pour récupérer les produits ;
- Les contrats de coordination [10] pour aligner les incitations des partenaires qui participent à la promotion des systèmes de logistique inverse et des chaînes fermées ;
- Gestion des retours de produits et des opérations de fabrication ;

[9] Reverse Logistics
[10] Coordination Contract

- Protocoles et réglementations pour les tarifs d'élimination, les interdictions d'élimination, les restrictions sur le transport des produits retournés, la prévention des déchets et le contrôle des effets destructeurs ;
- Les plans pour surveiller et inspecter les procédures d'élimination, de collecte, de stockage, de transport, de recyclage et d'élimination par les organisations ou les consommateurs ;
- Modèles pour optimiser la structure du réseau tout en minimisant les coûts et en maximisant la rentabilité [Asif et al., 2012, Fu and Meng, 2020].

3.3.3 Chaîne d'approvisionnement résiliant

Définition : Etant donné qu'un système est constitué de plusieurs sous-ensembles qui travaillent de manière coopérative et s'influencent les uns les autres, il existe un risque que les maillons de connexion de cet élément de réseau, et bien sûr l'ensemble du système, soient confrontés à des événements perturbateurs ou fluctuants [Rezapour et al., 2017]. La résilience d'un système se définit comme une aptitude qui préserve le système contre des perturbations internes ou externes ou à renouveler assez rapidement les systèmes en état normal en cas de perturbations. Une chaîne d'approvisionnement résiliente [11] est un système durable qui est stable contre les catastrophes naturelles, les perturbations industrielles ou une combinaison de tous les autres facteurs. Il est également capable de réduire les probabilités d'interruption et leurs conséquences de ces perturbations et se qualifie pour remettre rapidement la chaîne dans son état initial [Falasca et al., 2008, Lee et al., 2014].

Objectifs : En bref, l'objectif principal de la mise en place d'un système résilient est de faire attention à tous les incidents perturbateurs afin de pouvoir réagir le plus rapidement possible et d'avoir les facultés de repérer la structure endommagée en réduisant les vulnérabilités du réseau et le temps de récupération

Méthodes :

La résilience d'un système logistique en tant que chaîne d'approvisionnement a été considérée comme un enjeux important, en particulier dans l'environnement fluctuant actuel [Falasca et al., 2008]. Par conséquent, les topologies des stratégies à mener pour mettre en œuvre une infrastructure résiliente sont :

- Concevoir une structure agile pour réagir rapidement aux changements imprévus ou destructeurs ;
- Définir des politiques pour la gestion des risques afin d'atténuer les conséquences indésirables ;
- Choisir des fournisseurs alternatifs (multi-fournisseurs ou multi-approvisionnements) ;
- La prévention des inventaires de sécurité [12](stockage de sécurité) est supplémentaire ainsi que les capacités de stockage et les installations de sauvegarde ;

[11] Resilient Supply Chain
[12] Back-up Stock

- Éliminer la complicité du réseau en réduisant les nœuds inutiles ;
- Concevoir des réseaux de chaîne d'approvisionnement optimisés en réduisant le coût total pour atteindre le nombre optimal d'installations et les allouer aux clients pour éviter de perdre du temps à répondre à des demandes imprévues.

3.3.4 Chaîne d'approvisionnement allégée

Définition : La minimisation des coûts et la réduction des processus sans valeur est l'objectif initial pour toutes les organisations qui travaillent dans un environnement concurrentiel. La chaîne d'approvisionnement allégée [13] et une bonne solution pour répondre à ces besoins de façon efficace et économique. La chaîne d'approvisionnement allégée est un système intégré qui réduit ou élimine tous les processus et activités qui ne valent rien ou n'ajoutent pas de valeur à la chaîne [Drohomeretski et al., 2012, Myerson, 2012]. En effet, les principes des systèmes allégée sont de réduire ou de minimiser tous les coûts inutiles liés à tous les processus d'approvisionnement, de fabrication, de transport et de stockage du produit tandis que les qualités de service et de satisfaction client sont augmentées de manière productive.

Objectives : L'objectif fondamental de la chaîne d'approvisionnement allégée est de minimiser les coûts globaux dans l'ensemble du réseau logistique, mais la réalisation de cet objectif principal dépend de la réalisation d'autres objectifs tels que :

- Aligner les attentes des consommateurs en réduisant les délais de livraison et la qualité du produit ou du service ;
- Réduire ou éliminer les affaires supplémentaires en maximisant la valeur ajoutée pour les clients ;
- Atteindre une qualité de service élevée en mettant l'accent sur la réduction des déchets au point d'origine et dans le processus de fabrication ;
- Diminuer tous les actifs excessifs tels que les stocks, les machines, les transports inutiles et les ressources humaines.

Méthodes : La typologie des méthodes de la gestion de chaînes d'approvisionnement allégée est [Czarnecka et al., 2017, Martínez-Jurado and Moyano-Fuentes, 2014, Drohomeretski et al., 2012]:

- Le système de fabrication immédiate [14] pour réduire les coûts de stockage excessifs ;
- La sensibilisation des participants aux outils/techniques économisée ;
- Élimination des sources de perte de temps ou de capital et développement de ressources efficaces ;
- Réduction de niveau des inventaires ;

[13] Lean Supply Chain
[14] Just-in-Time System

- La Mesure de la performance de toutes les activités et l'évaluation de la relation avec les clients et les fournisseurs ;
- Cartographie des flux [15] de la chaîne afin d'identifier et d'analyser les demandes des clients [Czarnecka et al., 2017] ;
- Amélioration continue des produits et des flux d'informations.

3.3.5 Chaîne d'approvisionnement agile

Définition : L'agilité est une propriété concurrentielle qui met l'accent sur la flexibilité et la rapidité de la chaîne d'approvisionnement face aux demandes fluctuantes des clients. En effet, une chaîne d'approvisionnement agile [16] est un réseau capable de réagir rapidement, précisément et efficacement aux demandes personnalisées des consommateurs avec des dispositions physiques et managériales [Power et al., 2001, Yusuf et al., 2004].

Objectives :

L'intention essentielle d'un système agile est la satisfaction du client concernant la demande personnalisée avec un minimum d'attente (délai minimum). La satisfaction des client est atteinte avec les objectifs suivants :

- Répondre aux demandes personnalisées des clients de manière efficace, flexible, rapide et avec une qualité de service élevé ;
- Ajouter des valeurs pour la chaîne avec l'enrichissement des clients avant les concurrents ;
- Sensibiliser le système au changement de demandes de clientes ;
- Mobiliser les facultés ou les technologies pour des structures adaptables en optimisant les performances pertinentes.

Méthodes : Les stratégies qui sont impliquées dans la chaîne d'approvisionnement pour atteindre les vertus de l'agilité, mettent généralement l'accent sur l'importance de la flexibilité et de la vitesse du système pour répondre aux besoins des consommateurs [Ming et al., 2007, Ambe, 2009, Van Hoek et al., 2001]. Pour cette raison, les méthodes sont classés comme suit :

- Equiper les techniques pour identifier et détecter les changements ou les personnalisations des attentes des clients ;
- Planifier les programmations et les calendriers pour les fabrications immédiates ;
- Intégrer la structure du réseau avec des contrats de coordination équitables ;
- Promouvoir la gestion de l'externalisation et les fournisseurs alternatifs ;

[15] Value stream mapping
[16] Agile Supply Chain

- Globaliser d'agilité dans tous les processus intérieur et extérieur de la chaine telle que ressources agiles, fabrication agile, approvisionnement agile, et transportation agile.

Nous avons révisé les stratégies notables dans le domaine de la gestion des approvisionnements qui sont largement utilisées pour l'optimisation et l'intégration de ce système logistique connu. Tandis que chaque stratégie mobilise des instructions particuliers pour gérer le réseau d'approvisionnement, les organisations et les entreprises emploient des stratégies uniques ou hybrides (combinaison de plusieurs stratégies) selon leurs objectifs primordiaux afin de favoriser un système optimal et créer de la valeur pour les partenaires. De plus, au-delà des mécanismes de gestion des processus et des relations au sein de la chaîne d'approvisionnement, le fondement de la structure physique de la chaîne tels que la localisation et le nombre d'installations et d'établissements, les modalités d'attribution des services aux demandes, et le routage des transports, ont des impacts indispensables sur les fonctionnalités des systèmes logistiques. Pour cette raison, dans les sections suivantes, nous nous appuyons sur l'un des problèmes les plus importants dans la gestion des chaînes d'approvisionnement qui adresse les solutions pour concevoir une infrastructure rentable et équitable basée sur des modèles mathématiques et optimales.

3.4 Conception de réseau de chaîne d'approvisionnement

L'urbanisation de l'architecture de la chaîne d'approvisionnement est un sujet discuté dans les milieux académiques. De nombreuses études sont menées pour proposer une structure la plus optimisée et la plus favorable qui englobe toutes les dispositions relatives aux aménagements physiques et au stratégie managériale dans la mobilisation des services pour les clients finaux. De nos jours, la complexité des structures des grands systèmes logistiques avec les densités de nœuds (partenaires) et de liaisons, rend difficile la conception d'une infrastructure productive. Pour cette raison, une méthodologie appropriée relative à la prise de décision multicritères est exigeant afin d'urbaniser une architecture optimisée et d'augmenter le rendement du système. La figure 3.1 montre un réseau d'une chaîne d'approvisionnement et les coûts pertinents de chaque niveau.

La conception du réseau de la chaîne d'approvisionnement [17] contient toutes les décisions et la modularité liées aux déterminations de la structure intégrée et cohérente de la chaîne d'approvisionnement qui influencent les coûts et les performances globaux du réseau [Farahani et al., 2014]. De plus, selon [Ballou, 2001] la conception du réseau est un élément important de la capitalisation des investissements qui constitue la fondation des opérations et des processus de la chaîne d'approvisionnement.

Étant donné que l'architecture de la chaîne d'approvisionnement a mis en évidence des impacts sur le développement et la rentabilité du système logistique, il est évident que les procédures de prise de décision pour concevoir et lancer un réseau pratique sont essentielles pour les gestionnaires de chaîne. Le code général de la gestion globale de la chaîne d'approvisionnement fournit les facultés des procédures de prise de décision pour tous les aspects de la chaîne, qu'il s'agisse de la conception de l'architecture ou de la gestion des opérations. Sur la base de ce cognitif, la typologie des décisions principales qui couvrent tous les processus connexes de la conception,

[17] Supply Chain Network Design

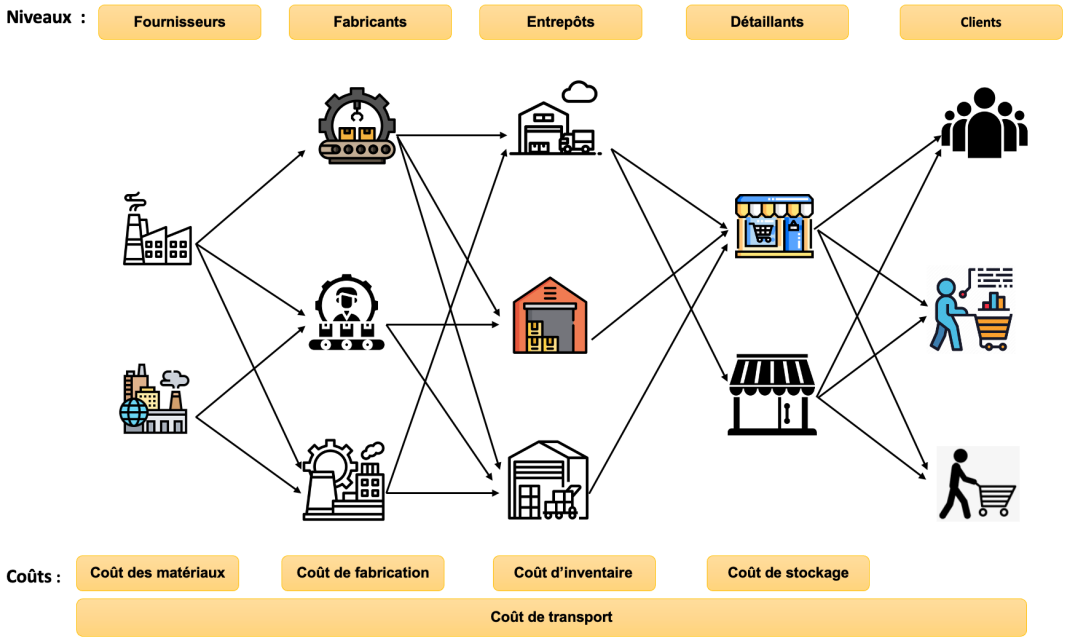


FIGURE 3.1 – Réseau de la chaîne d'approvisionnement

l'intégration et de l'optimisation de la chaîne d'approvisionnement, est organisée en trois niveau ou horizons généraux [Daskin, 2011, Chopra and Meindl, 2007] :

- **Les décisions stratégiques (long terme) :** Décisions stratégiques concernant toutes les décisions liées à l'objectif global et à la structure du système et affectant la chaîne d'approvisionnement à long terme, telles que l'emplacement de l'installation, la détermination du nombre optimal d'installations, d'entrepôts et de centres de distribution, la gestion de l'externalisation et des fournisseurs, et des plans de développement de produits [Fernandes et al., 2014, Muñoz et al., 2012] ;
- **Les décisions tactiques (moyen terme) :** Le niveau tactique correspond aux décisions à moyen terme qui déterminent les politiques globales de la chaîne concernant les optimisations et les contrôles des processus tels que la gestion des inventaires et de la logistique afin de déterminer la quantité optimale de commande ou de délai en mise en œuvre ;
- **Les décisions pratiques (court terme) :** Le niveau pratique couvre toutes les décisions quotidiennes dans la chaîne d'approvisionnement afin de mettre en œuvre les stratégies de deux niveaux inférieurs (stratégique et tactique) tels que la planification quotidienne ou l'ordonnancement des processus d'approvisionnement, de fabrication, de transport et de couverture des demandes [Wu and Zhang, 2014].

Étant donné que la conception de la chaîne d'approvisionnement est une fonction de prise de décision qui a nécessairement un impact sur tous les aspects de la fonctionnalité du réseau, elle devrait englober les ensembles de trois niveaux de prise de décision d'un point de vue stratégique, tactique et pratique. Avant la présence prépondérante de la chaîne d'approvisionnement

dans la gestion logistique, les revues de littérature des mécanismes managériaux ne sont normalement conduites qu'à un seul niveau décisionnel (stratégique ou tactique ou pratique) afin d'éviter la complexité du problème. Par exemple, plusieurs études sont axées sur les décisions de localisation de niveau stratégiques [Daskin et al., 2005, Cornuéjols et al., 1983], et d'autre part, [Cachon and Fisher, 2000, Singh and Verma, 2018] ont concentré sur la gestion de l'inventaire de niveau tactique. D'autre part, le problème de la conception du réseau de systèmes logistiques est un enjeu fondamental qui nécessite de prendre en compte simultanément tous les niveaux stratégiques, tactiques et pratiques. Pour cette raison, des problèmes hybrides pour les systèmes logistiques sont proposés afin d'aborder les mécanismes et modèles multi-objectifs qui mettent en évidence tous les niveaux de décision en même temps. Ces problèmes hybrides abordent les décisions à long terme (telles que l'emplacement, le problème d'allocation) ainsi que les stratégies à moyen et à court terme telles que (le contrôle des stocks, les problèmes de transport et d'acheminement, et la planification de la production), tout en mettant en œuvre la conception complète du réseau [Contreras et al., 2012].

En termes de conception du réseau de la chaîne d'approvisionnement, certaines décisions stratégiques, tactiques et opérationnelles majeures telles que l'emplacement des installations, la planification de la production, le contrôle des stocks, la distribution et les décisions logistiques doivent être prises en compte pour atteindre le réseau de chaîne d'approvisionnement coordonné et intégré. En raison de la forte interaction et de la dépendance de ces décisions, des problèmes hybrides qui abordent plusieurs problèmes de prise de décision conjointe sont considérés afin de concevoir et d'optimiser le réseau de la chaîne d'approvisionnement, tels que: problème d'inventaires, localisation [18], problème d'emplacement-allocation [19], problème de véhicule - routage [20], problème d'emplacement - routage [21], problème d'inventaire - routage [22] et problème d'inventaire - emplacement - routage [23] [Ahmadi Javid and Azad, 2010, Zhang et al., 2014].

Par exemple, les considérations d'inventaire ont un impact significatif sur le positionnement des installations et, d'autre part, la planification des itinéraires et les décisions d'acheminement affectent l'emplacement stratégique des dépôts. L'objectif des problèmes d'emplacement-allocation est de choisir le meilleur emplacement pour les installations du fournisseur ou du fabricant, référentiels et entrepôts de données pour minimiser les coûts totaux de gestion des installations et d'allocation de ces installations aux clients. Le problème du contrôle des stocks consiste à déterminer la quantité optimale de la quantité de commande ou du délai d'exécution pour maximiser la demande couverte et la satisfaction du client, minimiser le risque de rupture de stock et de perte de client, ou minimiser le coût de conservation des stocks. En outre, le problème de localisation-routage consiste à calibrer les itinéraires de livraison des produits ou services des installations et entrepôts actifs aux clients voisins dans la zone environnante afin de maximiser la demande de couverture et de minimiser les coûts de transport.

Le but de l'utilisation de problèmes de décision conjointe est d'atteindre le système d'optimisation qui est géré de manière cohérente et intégré avec des stratégies perceptibles et multifonctionnelles. Les stratégies hybrides ou la prise de décision combinée permettent aux organisations

[18] Inventory-Location problem
[19] Location-Allocation problem (LAP)
[20] Vehicle-Routing problem(LRP)
[21] Location-Routing problem
[22] Inventory-Routing problem
[23] Inventory-Location-Routing problem (ILRP)

d'avoir des perspectives étendues sur tous les aspects des systèmes logistiques afin de concevoir un réseau de chaîne d'approvisionnement parfait sous la de considérations stratégiques, tactiques et pratiques.

3.5 Résumé

Dans ce chapitre, nous avons parlé de la chaîne d'approvisionnement comme un exemple bien connu et efficace de systèmes logistiques qui fournissent une pleine capacité de facultés pour approvisionner et livrer des produits à des clients. Les demandes croissantes du marché ainsi que l'environnement concurrentiel mettent en évidence des stratégies et des outils pratiques, innovants et rentables de gestion et de prise des décisions qui permettent aux organisations d'ajouter de la valeur, d'augmenter la fidélité des clients, de surmonter les défis et de faire face aux risques internes et externes éventuels. Pour cette raison, les stratégies de gestion de la chaîne d'approvisionnement et les structures de réseau des membres et des entités sont constamment mises à jour afin qu'elles soient cohérentes avec les différentes demandes et situations du marché variées actuelles. Par conséquent, les systèmes logistiques pourraient être une bonne source d'inspiration pour toutes les infrastructures qui subissent les concepts logistiques pour fournir et mettre à disposition leurs services ou produits aux consommateurs ciblés.

Chapitre 4

Vers une vision logistique du lac de données

"L'analogie suppose un modèle et son imitation régulière. Une forme analogique est une forme faite à l'image d'une ou plusieurs autres d'après une règle déterminée."

– Ferdinand de Saussure 1857-1913

4.1 Introduction

L'analogie est un mécanisme approprié pour tirer profit des règles et stratégies d'un phénomène particulier pour mettre en œuvre les disciplines pour d'autres facultés grâce aux similitudes existantes entre ces faits. Le système est un phénomène omniprésent qui existe dans tous les domaines qu'il s'agisse d'un système naturel, ou d'un système biologique, ou d'un système organisationnel, ou d'un système logistique ou d'un système informatique. Il est évident que malgré les différents éléments et réglementations qui construisent un système, en fonction de sa tendance et de son intention, il existe plusieurs points similaires entre tous les systèmes basés sur la définition générale du système.

Un système est une faculté constituée de plusieurs éléments intégrés qui ingèrent les entrées déterministes, les transforme selon des règles prédéfinies, et génère les sorties pour les demandes attendues. Si nous considérons le système S avec les ensembles d'éléments coordonnés comme S=$\{s_1, s_2, s_3\}$, les caractéristiques du système peuvent être définies comme [Hall and Fagen, 2017, Backlund, 2000] :

- L'intervention et le comportement de chaque élément s_1, s_2, s_3 influencent l'ensemble du système ;
- Les éléments s_1, s_2, s_3 interagissent les uns avec les autres de manière collaborative et ils ont les relations d'interdépendance selon les protocoles déterministes ;
- L'environnement du système est définitif et est séparé de l'extérieur par une frontière ;
- Les éléments s_1, s_2, s_3 sont associés pour atteindre l'objectif global du système S selon les politiques et règles prédéfinies.

En général, tous les systèmes, qu'ils soient logistiques, informatiques ou naturels, suivent ces caractéristiques sous la définition de la structure intégrée d'un système alors qu'ils englobent différentes manières ou réglementations pour atteindre ses objectifs. Cette similitude intrinsèque est un atout essentiel de ces différents systèmes pour qu'ils puissent profiter des disciplines et des stratégies efficaces de l'un des autres pour gérer leurs structures.

Étant donné que dans cette étude nous nous appuyons sur la structure systémique du lac de données et nous abordons les méthodes interdisciplinaires inspirées de deux systèmes logistiques et naturels pour la gestion des plate-formes informatiques, nous orientons cette étude vers l'analogie des systèmes essentiels et utiles afin que nous peut mettre en évidence et remarquer les points similaires entre ces trois systèmes (le lac de données comme système informatique, la chaîne d'approvisionnement comme système logistique et l'écosystème comme système biologique

ou naturel). Cette analogie nous permet de tirer parti des principes et stratégies de gestion pour augmenter la fonctionnalité de la structure systémique du lac de données. Pour cette raison, nous choisissons deux systèmes connus et, notamment instinctivement différents, comme un système naturel tel qu'un écosystème (le nom du lac de données est dérivé d'un système naturel), un système logistique tel que la chaîne d'approvisionnement (basé sur l'objectif du lac de données pour fournir et livrer des données en tant que produit, ainsi que la chaîne d'approvisionnement est un système efficace de gestion du cycle de vie du produit) pour faire une analogie générale entre lac de données et ces deux systèmes. Cette approche met en évidence les similitudes entre ces trois systèmes afin d'inventer la nouvelle architecture du lac de données et les stratégies pour gérer et optimiser les performances de ce système informatique grâce aux outils imités de deux systèmes tout en gardant la définition générale et les principales règles d'un système.

Dans ce chapitre nous allons aborder :

- L'analogie entre trois systèmes naturels, logistique et informatique ;
- Les outils, stratégies et disciplines logistiques pour la gestion des lacs de données ;
- Les outils, stratégies et disciplines naturels pour la gestion des lacs de données ;
- Les bases essentielles pour la conception de l'architecture inspirée des systèmes logistiques pour un lac de données.

4.2 Analogie basée sur la perspective systémique

Les métaphores et l'analogie entre les phénomènes sont souvent utilisées en science par les méthodes multidisciplinaires pour générer de nouvelles approches ou stratégies pour un domaine à partir de concepts et de principes d'autres domaines dans des conditions semi-similaires.

Selon [Barbot et al., 2019] :

"Les méthodes et la pensée analogiques sont les mécanismes de pensée applicables pour comparer des faits, des situations ou des objets selon des critères similaires afin de tirer des conclusions ou d'inventer de nouvelles facultés à partir de la comparaison effectuée."

Sur la base de ces connaissances, de nombreuses sciences ont émergé d'après des concepts d'autres domaines tels que le système neurologique basé sur les neurosciences, l'intelligence artificielle basée sur le système cognitif humain, l'algorithme évolutif basé sur des phénomènes naturels, et en particulier le lac de données imaginé par analogie au lac naturel. Ces exemples réussis nous conduisent à l'idée que des facultés d'autres sciences pourraient s'inspirer efficacement pour la gestion des systèmes informatiques. Étant donné que le lac de données est considéré comme un système de gestion et de stockage de données, tous les phénomènes systémiques qui agissent comme un ensemble d'éléments intégrés et unifiés vers l'objectif global, pourraient potentiellement être une source d'inspiration pour imiter les outils pratiques. Pour cette raison, nous pensons à deux systèmes connus, l'un dans la nature et l'autre dans l'organisation, afin de construire les fondations d'une nouvelle architecture et méthodes de gestion des lacs de données basées sur les principes et règles de ces deux systèmes et plutôt le système logistique. Des

études ont démontré que ces deux systèmes ont la capacité davantage d'intégrer et de coordonner les membres et les niveaux inclus et d'optimiser la structure de leur environnement grâce aux stratégies qu'ils utilisent [Anderson et al., 2007, Chopra and Meindl, 2007, Mentzer et al., 2001, Lambert and Cooper, 2000, Ayers, 2000, Panayides, 2006, Janvier-James, 2012, Murdoch and Oaten, 1975, Combes, 2001].

Revenant à la définition du système, nous pouvons imaginer un système comme une boîte qui absorbe les entrées, les transforme et les retire comme résultat final. Par conséquent, selon cette définition, on peut montrer que les trois systèmes représentés utilisent la même procédure sous la forme d'un système pour effectuer une action. Le système logistique transforme les matériaux en produits ou services particuliers, l'écosystème en tant que système naturel, prend les différentes formes d'énergie et de matière pour créer les structures biologiques, et de la même manière, le lac de données en tant que système de stockage de données centralisé, ingère les données brutes des sources éparpillées et les stocke pour générer les informations. La figure 4.1 manifeste différentes structures sous la définition des systèmes.

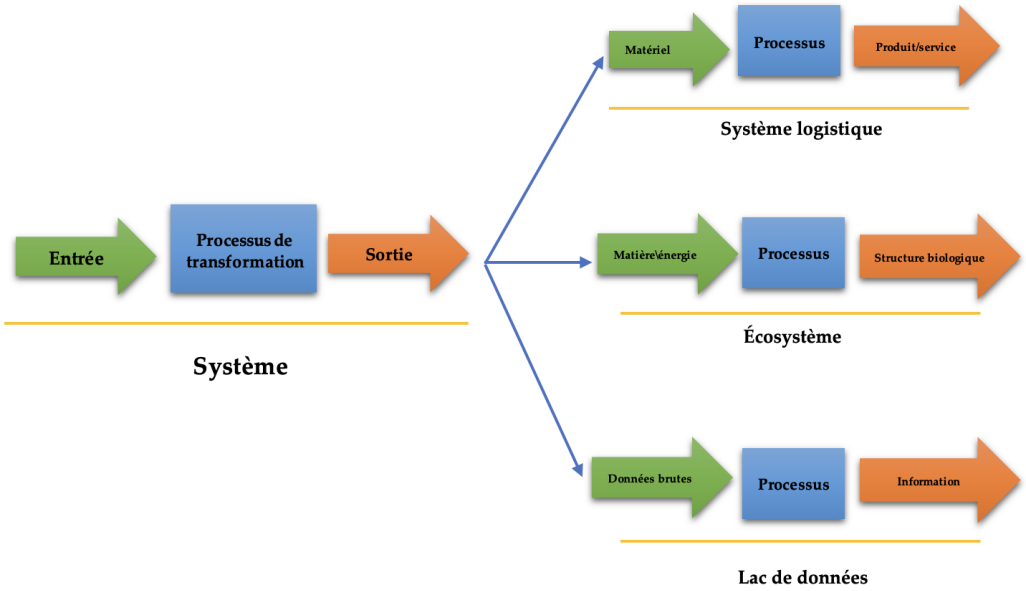


FIGURE 4.1 – Les différents structures systémiques

Dans ce stade, les questions qui ont surgi sont :

- *Quels aspects du lac de données sont comparables aux systèmes logistiques et aux systèmes naturels?*
- *Quels critères devraient être appliqués pour comparer les structures systémiques?*
- *Quelles stratégies ou outils pourraient être dérivés de cette analogie pour la gestion ou l'optimisation du lac de données?*
- *Comment les stratégies imitées et inspirées des systèmes logistiques et naturels sont-elles applicables pour améliorer les performances du lac de données?*

Dans cette section, nous tentons d'appporter une réponse à ces questions avec :

- *La mise en œuvre de méthodes analogiques pour comparer les trois systèmes indiqués élément par élément selon les modules définis* ;
- *Imitation des principes et des mécanismes naturels pour la gestion du lac de données (L'approche naturelle)* ;
- *Application des stratégies de la gestion de la chaîne d'approvisionnement pour l'optimisation des systèmes de stockage de données (L'approche logistique)* ;
- *Inspiration des méthodes d'évaluation des produits comme protocoles de gouvernance des données* ;
- *Concevoir une architecture logistique pour lac de données* .

4.2.1 La méthode analogique

Nous avons remarqué dans les sections précédentes que la chaîne d'approvisionnement, l'écosystème et le lac de données agissent intrinsèquement comme un système intégré. Pourtant, la question qui se pose est à quel point on pourrait se rendre compte que ces trois systèmes pourraient s'inspirer l'un pour l'autre. Pour cette raison, nous choisissons une méthode analogique basée sur la définition du système pour effectuer une comparaison précise sur ces trois systèmes. Pour la première étape, nous lançons les tableaux de comparaison avec les modules explicatifs afin d'échanger des points de vue similaires entre les systèmes ciblés. Pour la deuxième étape, nous détaillerons des points similaires et proposerons les méthodes de gestion imitées pour le lac de données à partir des deux systèmes de référence. Les tableaux 4.1 et 4.2 présentent une comparaison générale des trois systèmes.

Les tableaux 4.1 et 4.2 répondent aux deux premières questions : *"Quels aspects du lac de données sont comparables aux systèmes logistiques et aux systèmes naturels?"* et *Quels critères devraient être appliqués pour comparer les structures systémiques?*. Le lac de données est vu comme un système intégré de stockage de données, il est comparable en plusieurs points avec les deux systèmes logistique et naturel. Pour bien aborder le sujet, nous avons choisi la chaîne d'approvisionnement comme exemple de système logistique et le lac naturel comme exemple de système naturel et nous avons abordé toutes les définitions et problématiques correspondant aux structures systémiques. Dans les sections suivantes, nous mettrons l'accent sur la fonctionnalité de cette analogie pour répondre aux deux questions restantes de la section 4.2 sur la façon dont nous pouvons tirer parti des points similaires entre trois systèmes de gestion du lac de données.

4.3 Le lac de données et la chaîne d'approvisionnement

L'objectif principal de cette étude est de définir un pipeline logistique pour le lac de données et d'optimiser cette architecture grâce aux stratégies dérivées de la gestion des chaînes d'approvisionnement. Par conséquence, dans cette section, nous détaillerons chaque module déterminé

TABLE 4.1 – Analogie de la chaîne d'approvisionnement, de l'écosystème et du lac de données

Module	Chaîne d'approvisionnement	Lac naturel (Espèces et écosystèmes)	Lac de données
Membres/Niveaux	Fournisseur Fabricant Distributeur Détaillant Client	Composants de l'écosystème (Animaux, Plantes, Micro-organismes) Processus biologiques (Élever, Naître, Grandir et Mourir) Processus écologiques (Manger et être mangé)	Couche d'ingestion Couche du stokage Couche de traitement Couche d'accès Utilisateur final
Produits	Marchandise (Flux vers l'avant) Information et produits recyclables (Flux vers l'arrière)	Biodiversité (Diversité des espèces) Complexité écologique (plus d'espèces plus complexe) Biomasse	Données
Stratégie de gestion	GCL allégée GCL agile GCL ajournements GCL Spéculation GCL verte GCL allégé-agile GCL résilient GCL Fuzzy Conception de réseau de CL	Évolution des espèces (Mutation, Recombinaison, Dérive, Sélection) Compétition Parasitisme (association négative) Mutualisme (Association positive) Prédation	Lac de données allégé Lac de données agile Gestion des métadonnées Gestion de données Gouvernance des données Lac de données allégé-agile Lac de données résilient Lac de données Fuzzy Conception de pipeline
Fonctions objectifs	Minimisation des coûts Maximisation des ventes Maximisation des profits Minimisation des délais	Au niveau de l'espèce : Maximiser la reproduction et survivre (Fitness) Au niveau de l'écosystème : Maximiser la résilience	Minimisation des temps d'exécution Maximisation du taux de remplissage Minimisation de temps de réponse Minimisation des coûts Maximisation des profits

TABLE 4.2 – Spécifications du problème de la chaîne d'approvisionnement, de l'écosystème et du lac de données

Module	Chaîne d'approvisionnement	Lac naturel (Espèces et écosystèmes)	Lac de données
Variables de décision			
	Nombre de transportation		Séquence des services
	Délai de livraison		Temps d'exécution
	Quantité de commande		Nombre d'utilisateurs
	Quantité de stock de sécurité	Homéostasie	Nombre de Jobs de traitement
	Séquence des services		Planification des job
	Nombre de niveaux		
	Nombre d'installations		
	Délai de mise en œuvre		
Contraintes			
	Budget	Émergence du lac	Capacité mémoire
	Nombre d'entrepôts	Perturbation	Capacité de CPU
	Nombre d'installations actives	Changements globaux	Gravité des données
	Capacité des installations		Principes de gouvernance des données
	Conformité des services		Conformité des services
	Délai de préparation et livraison		Mapping
Risques			
	Risque de perte le client		Échec du mappage des données
	Risque de produit défectueux		Panne de machine
	Risque de défaillance de l'information		Sécurité des données
	Risque de défaillance de la qualité	Fortes perturbations	Risque de fiabilité de données
	Risque de sur-stock de produits		Risque d'échec d'accessibilité
	Risque de retard		Risque de perte de données
	Risque de goulot d'étrangelement		Risque de temps d'exécution élevé
Mesure de la performance			
	Satisfaction du client	Richesse en espèces	Qualité des données
	Satisfaction des transactions	Fonctions des écosystèmes	Flexibilité du lac de données
	Flexibilité	Resilience	Intégration des données
	Intégration des informations		Accès rapide aux données brutes
	Délai d'exécution faible		Délai de temps d'exécution faible
	Performances des fournisseurs		Agilité
	Performances de fabrication		Niveau d'optimisation
	État du transportations		
	Niveau d'optimisation		

dans les tableaux 4.1 et 4.2 et nous expliquerons comment les procédures logistiques peuvent être appliquées dans le lac de données dans un manière efficace et pratique.

4.3.1 Membres/Niveaux

Chaque système contient les éléments qui peuvent être considérés comme des membres, des participants ou des niveaux et s'organisent en aval et en amont. Les performances de chaque participant (niveau) sont les impacts directs ou indirects sur les autres niveaux et en particulier sur l'ensemble du système [Hall and Fagen, 2017, Backlund, 2000]. Pour cette raison, l'intégration verticale et horizontale, et la coordination des niveaux du système sont les objectifs essentiels à atteindre à travers les stratégies de gestion des systèmes pour les décisionnaires. Une chaîne d'approvisionnement qui agit comme un système logistique est composée de participants tels que des fournisseurs, des fabricants, des distributeurs, des détaillants et des clients qui approvisionnent des produits ou des services particuliers. Chaque niveau de la chaîne d'approvisionnement doit être géré et régi par des décisions correctes et ponctuelles afin d'améliorer la fonctionnalité et le rendement globales de la chaîne. Par exemple, la gestion de la sélection des fournisseurs, la planification de la production, le contrôle d'inventaire, la planification des itinéraires et des transports et la gestion du service client. De la même manière dans le lac de données en tant que système de gestion de données, on pourrait définir les éléments du système comme les couches dans lesquelles chaque couche est responsable de la tâche particulière pour augmenter l'efficacité de l'ensemble du système.

Selon les architectures proposées pour le lac de données au chapitre 2 telles que 2.2, 2.3, 2.4, et 2.5, la plateforme globale du lac de données est basée sur les plusieurs couches qui sont ***la couche d'ingestion*** (responsable d'accueillir toutes les formes de données), ***la couche de stockage*** (responsable du stockage des données), ***la couche de traitement*** (responsable de la transformation et de la manipulation des données), et ***la couche d'accès*** (responsable de fournir les informations aux utilisateurs finaux). En considérant le raisonnement systémique et logistique, toutes les couches sont interdépendantes et pour développer l'efficacité de l'architecture, les stratégies d'intégration, de coordination et en particulier d'optimisation sont exigeantes.

4.3.2 Produit

Le résultat d'unions et d'associations de systèmes est un produit ou un service. Dans la chaîne d'approvisionnement bidirectionnelle, ce produit ou service peut prendre deux formes selon le sens de circulation dans la chaîne. Le système logistique en boucle fermée prend en charge deux directions de service, le flux vers l'avant de marchandises et le flux vers l'arrière des informations en retour ou les produits recyclables assemblées à partir des clients [Borgström, 2010]. Le produit du lac de données est la donnée. Étant donné que les données sont un atout important pour les organisations, d'un point de vue systémique, les données sont considérées comme un produit ou un service dans la chaîne d'approvisionnement qui est fourni, stocké, traité et mis à la disposition des utilisateurs via des systèmes de gestion de données. Ainsi, la qualité des données et des services du lac de données sont les critères problématiques qu'ils faudra être gérés afin de garantir une plateforme rentable et bénéfique. De plus, la définition de flux vers l'arrière pour le lac de données pourrait être une nouvelle idée pour démarrer un lac de données bidirectionnel ou en boucle

fermée pour surveiller le parcours des données après utilisation et recueillir des informations supplémentaires pour améliorer la qualité des données.

Les figures 4.2 et 4.3 manifestent la chaîne d'approvisionnement et le lac de données bidirectionnel avec deux flux de produit (données) et d'informations.

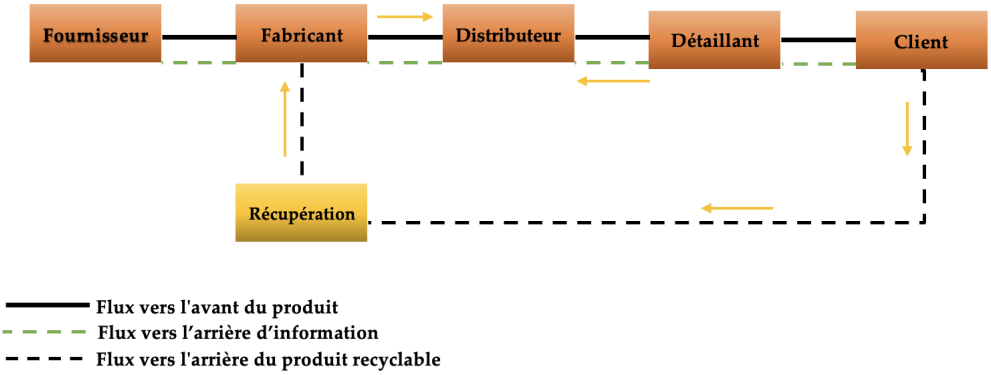


FIGURE 4.2 – Chaîne d'approvisionnement en boucle fermée

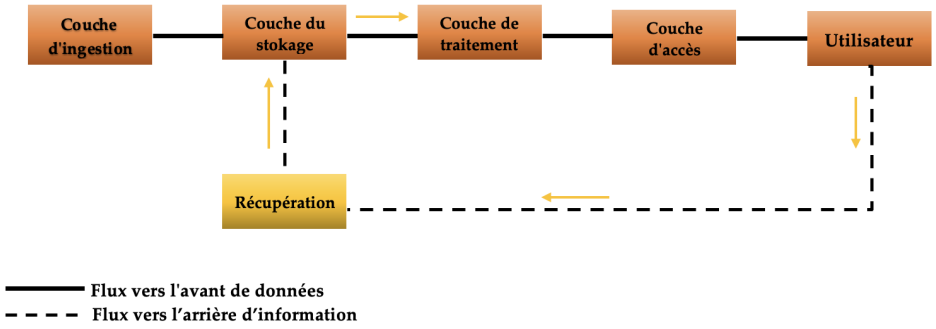


FIGURE 4.3 – Lac de données en boucle fermée

4.3.3 Stratégie de gestion

Lorsque l'on parle de rentabilité ou d'optimisation des systèmes, on parle de l'exigence du programme intégré de planification et de gestion, basé sur des stratégies à court et à long terme. Les stratégies de gestion sont définies en fonction de la fonctionnalité, des objectifs, des contraintes et des demandes requises des systèmes. La philosophie des stratégies de gestion des systèmes logistiques est de prendre toutes les décisions et d'implémenter toutes les activités liées au flux de produits et d'informations tout au long du système [Mentzer et al., 2001]. Sur la base de cette philosophie, les stratégies de chaîne d'approvisionnement contiennent tous les instructions pour faire face aux incertitudes possibles des fournisseurs et des demandes, ainsi qu'unifier tous les membres et entités de la chaîne et aligner les intérêts pour ajouter les valeurs à l'ensemble du système logistique.

Comme nous l'avons indiqué au chapitre 3, les stratégies de gestion de la chaîne d'approvision-

nement sont classées en plusieurs méthodes et outils selon les structures de la chaîne (centralisée ou décentralisée), la gamme d'incertitudes de l'environnement externe (approvisionnement et demande), les interactions et les coopérations internes, et les types de besoins des clients. Les stratégies les plus importantes et pratiques dans ce domaine qui peuvent apporter les bonnes solutions aux problèmes de la chaîne d'approvisionnement sont nommées comme la stratégie allégée (les coûts supplémentaires doivent être réduits ou éliminés), la stratégie agile (flexibilité pour répondre à divers besoins), la stratégie allégée-agile (un compromis entre l'allègement et l'agilité), la stratégie résiliente (résistance aux pannes), la stratégie ajournement [1] (la réduction de risque d'incertitude avec suspension de l'état du produit jusqu'à ce que la demande finale soit faite) [Pagh and Cooper, 1998], la conception du réseau de chaînes (la planification stratégique de la structure de la chaîne), et la stratégie verte (la chaîne avec considération environnementale) [Beamon, 1998]. Evidemment, le lac de données en tant que système informatique nécessite des stratégies de gestion spécifiques afin d'augmenter sa rentabilité et sa qualité fonctionnelle. Pour atteindre cet objectif, des études sont menées dans le cadre de l'amélioration du lac de données, soit sur son architecture du pipeline, soit sur sa fonction technique, soit sur le choix des logiciels appropriés, soit sur la réglementation et la gouvernance des données, soit sur la gestion des métadonnées et des données. En outre, les méthodologies systémiques de cohérence des entités du système pourraient être les bases et une source d'inspiration pour discipliner et synchroniser le lac de données de manière exhaustive. En tenant compte du tableau d'analogie 4.1, nous pouvons donner les perspectives d'utilisation de stratégies de gestion des systèmes logistiques au sein du lac de données.

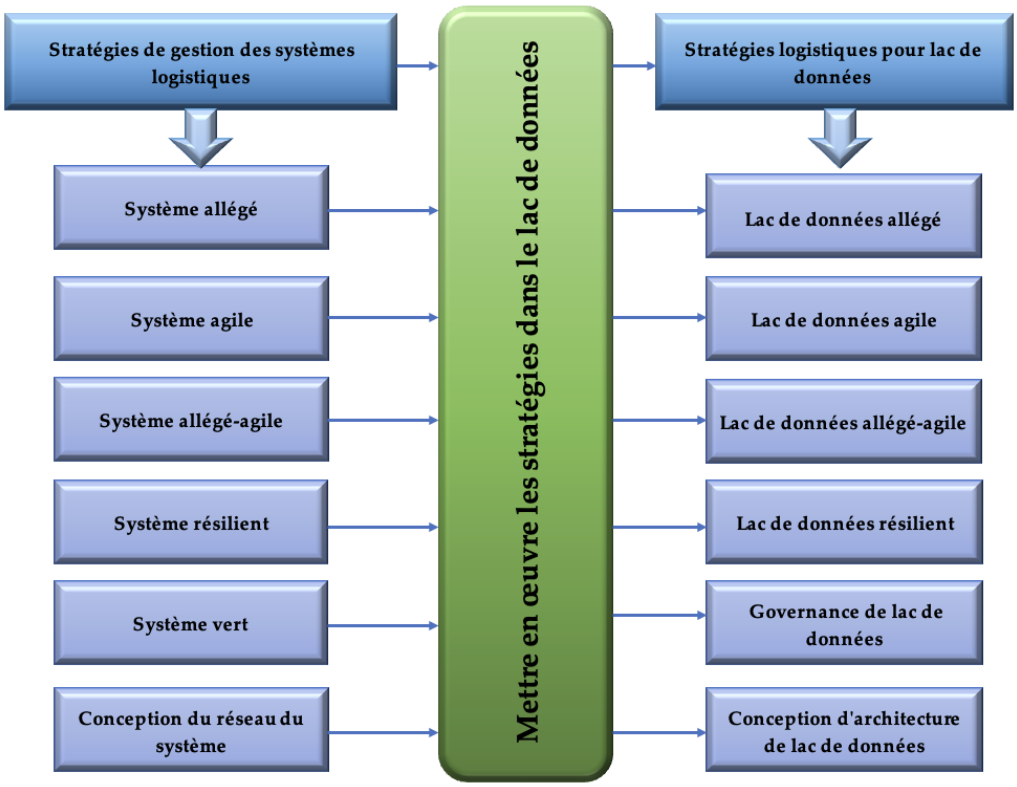


FIGURE 4.4 – Analogie des stratégies de la gestion des systèmes

[1] Postpone strategy

La figure 4.4 montre que chaque stratégie de gestion de la chaîne d'approvisionnement a un correspondant dans la cadre de gestion du lac de données. Afin de répondre aux troisième et quatrième questions posées dans la section 4.2 dont ***Quelles stratégies*** et ***Comment*** sont-elles applicables pour la gestion ou l'optimisation du lac de données, nous aborderons l'explication de la mise en œuvre de la stratégie logistique dans le cadre du lac de données comme suit.

Lac de données allégé

Définition : Selon la définition de chaîne d'approvisionnement allégée dans les sections 3.3.4, la stratégie allégée pour lac de données est définie comme toutes les décisions et actions qui tentent de réduire ou d'éliminer les activités et les processus coûteux dans le lac de données. Le coût peut être considéré comme les coûts de lancement du lac de données (coûts d'investissement, coûts matériels et logiciels) ou comme les processus qui augmentent le temps d'exécution du système. Généralement, lac de données allégé est un système de stockage de données massif qui minimise tous les coûts supplémentaires, qu'il s'agisse des coûts de lancement ou des coûts d'exécution (temps d'exécution), en éliminant tous les opérations ou les matériaux superflus et inutiles avec un compromis entre les avantages et les inconvénients de critères décisionnels. Le lac de données allégé est une perspective favorable qui est l'objectif principal de l'innovation lac de données par rapport aux autres systèmes de gestion de données [Sundaram and Vidhya, 2016, Ravat and Zhao, 2019, Russom, 2017].

Objectifs :

- Minimiser le temps d'exécution du système ;
- Éliminer les procédures excessives et sans valeurs ;
- Minimiser le temps d'ingestion, de stockage, de traitement et de réponse des requêtes.

Méthodes :

En implémentant les méthodes de stratégie 3.3.4 pour le lac de données nous avons travaillé à :

- L'optimisation de l'architecture et du pipeline du lac de données grâce aux stratégies de chaîne d'approvisionnements allégée ;
- La sélection de logiciels et de matériel économique ;
- Le service de lac de données en cloud ;
- La sélection du fournisseur de services efficace avec la méthode MCDM [2] [Yazdani, 2014, Petrović et al., 2019] ;
- La cartographie des flux dans lac de données ;

[2] Multi-criteria decision-making

- L'amélioration continue de l'état du lac de données avec l'évaluation de l'état actuel et création de l'état idéal du lac [Czarnecka et al., 2017] ;
- La libération de la mémoire en nettoyant les données inutiles ou répétitives comme le contrôle d'inventaire de la chaîne d'approvisionnement ;
- La prise de conscience des impacts de causes et effets de tous les processus, matériaux ou services coûteux et non-valeurs [Kovac, 2013].

Lac de données agile

Définition :

L'agilité est instinctivement prise en compte dans la définition du lac de données lorsque l'on prend en compte cette définition :

" Un lac de données est un dépôt de stockage de données qui héberge toutes les données brutes dans des formats natifs tels que les données structurées des bases de données relationnelles (base de données SQL), les données semi-structurées (CVS, XML, JASON) et les données non structurées (emails, photos, vidéos) de bases de données NoSQL ainsi que prise en charge des opérations de traitement transactionel en ligne et du traitement analytique en ligne des données [Khine and Wang, 2018]. "

Cette définition montre que la gamme de services du lac de données est bien étendue pour répondre aux besoins variés des utilisateurs dans l'environnement du big data [Madera and Laurent, 2016]. Par conséquent, la flexibilité et l'agilité sont les vertus évidentes du système de stockage de données volumineuses qui devraient être renforcées par des stratégies pragmatiques. La chaîne d'approvisionnement agile et sa méthodologie d'agilité sont de bons modèles inspirants pour réaliser un lac de données agile systémique.

Objectifs :

- Maximiser la vitesse de réponse aux demandes personnalisées des utilisateurs ;
- Maximiser la flexibilité des plates-formes pour accueillir des données dans divers formats depuis diverses sources.

Méthodes :

En implémentant les méthodes de stratégie 3.3.5 pour le lac de données nous avons travaillé à :

- La conception d'une architecture hybride de lac de données qui supporte les données en temps réel et en lot en même temps ;
- La mobilisation des dispositions pour les différentes approches analytiques [Miloslavskaya and Tolstoy, 2016] ;

- La sensibilisation du système aux demandes variées de utilisateurs ;
- aux dispositions alternatives en cas de panne de machine telles que la promotion de la gestion de l'externalisation dans la chaîne d'approvisionnement ;
- La mise en œuvre de l'agilité dans toutes les couches du lac de données.

Lac de données allégé-agile

Définition :

Un système idéal est celui qui trouve un équilibre entre l'allègement et l'agilité. Cependant, cette caractéristique est difficile à obtenir car il est nécessaire d'établir un point de découplage entre l'allègement et l'agilité du système [Mason-Jones et al., 2000, Goldsby et al., 2006]. Dans les systèmes logistiques, ce point est séparé entre l'aval et l'amont en déterminant le temps le plus long supportable par les clients. Par conséquent, afin d'achever un lac de données allégé-agile, un compromis entre coût, qualité, rapidité et satisfaction des utilisateurs devra être pris en compte.

Objectifs :

Les objectifs de cette stratégie sont classés sous la problématique des fonctions multi-objectifs afin de réussir à établir un équilibre entre les conflits d'intérêts.

- Maximiser la rentabilité en minimisant les coûts totaux du système ;
- Maximiser le niveau du service en minimisant le temps d'exécution.

Méthodes :

- Concevoir une architecture flexible sans ajouter de coûts supplémentaires aux niveaux de logiciel et de temps d'exécution ;
- Réduire les coûts (par exemple, le temps d'exécution dépassé) sans restreindre l'agilité du système pour répondre aux demandes personnalisées [Mason-Jones et al., 2000].

Lac de données résilient

Définition :

Sur la base de la définition d'un système résilient dans la section 3.3.3, un lac de données résilient pourrait être manifesté comme un réseau résistant qui réagit rapidement, au bon moment et avec les bonnes stratégies contre les vulnérabilités, les perturbations et risques externes et internes ainsi qu'il a la capacité de se récupérer son état initial sans les impacts destructeurs à

long terme. Deux termes importants ***l'événement perturbateur*** et ***le temps de récupération*** sont dérivés de cette définition. L'événement perturbateur est lié à tous les risques et événements qui interrompent l'état normal du lac de données tels que, panne de serveur ou de machine ou défaillance des couches, qui pourraient avoir des effets destructeurs sur la fonctionnalité du lac de données à long ou à court terme. Afin de mesurer l'état de fonctionnalité du système sous les impacts d'événements perturbateurs, [Chatterjee and Layton, 2020a, Castet and Saleh, 2012] indique la propriété de survie des systèmes qui est appelée "***Capacité de survie***" [3]. La survivabilité manifeste l'état de la fonctionnalité du système après une perturbation imprévue et estimée comme suit :

$$S = 1 - \left(\frac{F(t_i) - F(t_s)}{F(t_i) - F(t_s)_{bas}} \right) \tag{4.1}$$

L'équation 4.1 est présentée par [Chatterjee and Layton, 2020a] pour déterminer la capacité de survie du réseau de systèmes tels que la chaîne d'approvisionnement. Évidemment, cette équation pourrait être étendue pour mesurer la survivabilité du lac de données dans laquelle $F(t_i)$ indique la fonctionnalité du lac de données à l'état initial (avant la perturbation), $F(t_s)$ et $F(t_s)_{bas}$ sont respectivement la fonctionnalité minimale et la fonctionnalité la plus basse (la plus mauvaise) du lac de données après la perturbation.

Une autre métrique proposée par [Chatterjee and Layton, 2020a] pour évaluer la résilience du système est le temps de récupération après un état anormal. Le temps de récupération T_R manifeste le temps mis par le système pour revenir en état normal (état initial). Rationnellement, plus le temps de récupération est long, plus la résistance du système est faible. Cette métrique est estimée à partir de la durée entre le temps de l'interruption T_i et le temps du retour à la normale T_n.

$$T_R = T_n - T_i \tag{4.2}$$

[3] Survivability

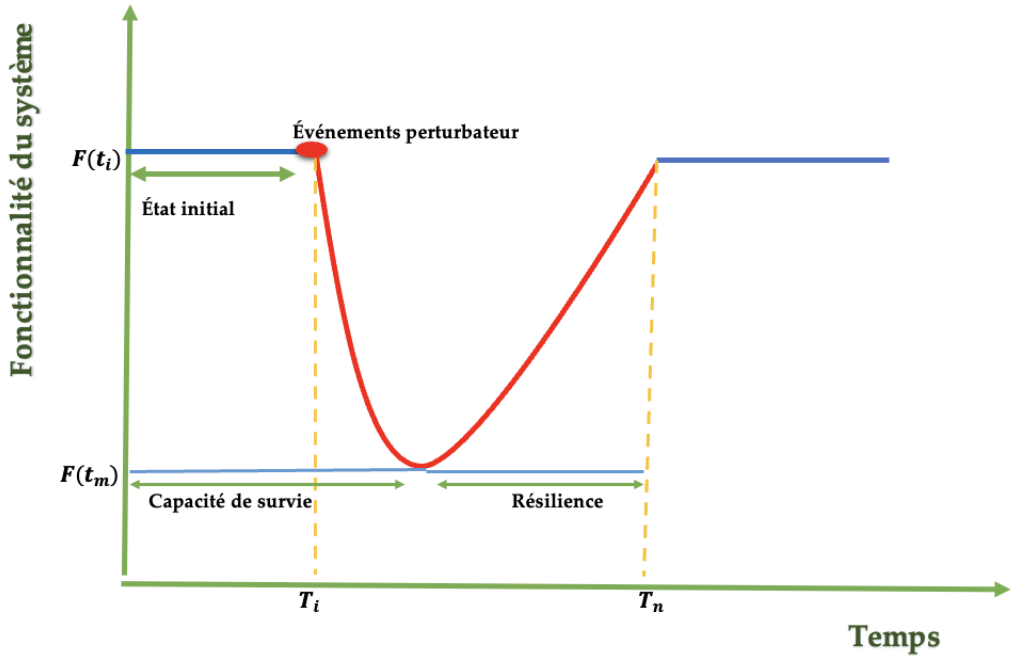


FIGURE 4.5 – Cycle de résilience du système [Bešinović, 2020, Linkov and Palma-Oliveira, 2017]

D'autres études sont menées sur l'impact des structures du système et des relations entre les entités pour examiner la résilience des réseaux de systèmes logistiques [Kim et al., 2015]. Selon cette étude, la structure des systèmes pourrait être une source de perturbations locales ou globales ainsi que pourrait declancher ou atténuer les événements perturbateurs. [Kim et al., 2015] tirent parti de la théorie des graphes pour analyser et diagnostiquer les influences de différentes structures de réseau sur l'aptitud de résilience du système. En se basant sur cette étude, nous pouvons supposer le lac de données, qui répond en parallèle à la demande des utilisateurs, comme un graphe avec les nœuds (les entités de chaque couche) et les arcs (le flux de données) dans lesquels la fonctionnalité et le l'état de chaque nœud pourrait affecter l'état de capacité de survie ou de vulnérabilité du système. L'interruption de chaque nœud perturbe le flux de données au sein du lac de données de manière inefficace. Pour cette raison, des stratégies et des méthodes de gestion qui pourraient anticiper ou prévoir les risques d'événements perturbateurs ou gérer et réhabiliter le système après une perturbation destructrice sont essentielles dans les systèmes de gestion de données.

Objectifs :

- Minimiser les risques de perturbations internes ou externes du lac de données ;
- Minimiser le temps de récupération en cas de perturbation ;
- Maximiser la flexibilité, la recouvrabilité, et la capacité de survie du du lac de données contre les événements perturbateurs.

Méthodes :

En référence à la section 3.3.3, les méthodes pour la mise en œuvre un lac de données résilient sont :

- L'évaluation continue de la vulnérabilité du lac de données ;
- La détection et diagnostic des anomalies et événements perturbateurs avec contrôle sensible du lac de données ;
- La conception d'une structure résiliente avec des dispositions alternatives contre les changements destructeurs.

Lac de données en boucle fermée

Définition :

Toutes les structures systémiques fonctionnent sous contrôle de rétroaction. Le retour d'informations et d'expérience sont indispensables pour analyser les données expérimentales ou garantir la cohérence et la sécurité du système grâce aux informations retournées [Van den Hof, 1997, Ruan, 2017]. Cette caractéristique est obtenue grâce à des systèmes en boucle fermée qui prennent en charge toute la circulation(flux) directe et inverse du produit et de l'information. Comme expliqué dans la section 3.3.2, la logistique inverse dans la chaîne d'approvisionnement aborde des considérations environnementales telles que la réception ou la reproduction de produits recyclables ainsi que le retour d'informations expérimentales des consommateurs. En termes d'analogie avec les systèmes logistiques, le lac de données en tant que plate-forme systémique nécessite des contrôles de rétroaction pour optimiser l'état d'équilibre du réseau. La structure en boucle fermée du lac de données met en évidence les analyses des expériences des utilisateurs en relation avec les services rendus ainsi que l'augmentation de la cohérence et de l'intégrité des données en enrichissant le système avec informations supplémentaires ou de données traitées. Pour cette raison, nous essayons d'implémenter cette logistique inverse des données dans le lac de données en proposant l'architecture ***ALLD*** qui sera étudiée dans le chapitre suivant afin d'introduire un lac de données en boucle fermé.

Objectifs :

- Maximiser la cohérence du système avec les données expérimentales ;
- Être vigilant vis-à-vis des reports effectués par les utilisateurs afin d'augmenter la qualité de service ;
- Rendre disponible la réutilisation des données traitées et analysées par les utilisateurs.

Méthodes :

- Organiser les procédures de récupération des données analysées ou visualisées par les utilisateurs via la logistique inverse dans le lac de données ;

- Provisionner des dispositions de couche de stockage pour stocker les données de rétroaction ;
- Évaluer de manière continue les signaux de l'utilisateur.

Gouvernance de lac de données

Définition :

Avec la croissance de l'hétérogénéité des données dans l'environnement technologique, la gouvernance des données devient un enjeu qui fait l'objet de débats pour les systèmes de gestion de données massives. De plus, avec la révolution des systèmes informatiques et l'émergence de référentiels centralisés tels que le lac de données, le lignage, la véracité, la fiabilité, la facilité à trouver, la qualité, la pureté et la viabilité des données mettent en évidence les exigences des réglementations et des protocoles pour la gestion et la gouvernance des données [Giebler et al., 2019]. Pour cette raison, les termes l'ontologie et les données FAIR ont émergé pour décrire un système de gestion de données gérable et interprétable. Les lacs de données dus à l'ingestion de données multi-structurées et brutes sont plutôt ciblés pour les stratégies de gestion de données afin d'éviter du phénomène de marécage de données inutile. Pour appliquer les méthodes de gouvernance des données efficace, il faut surveiller les données tout au long de leur cycle de vie afin de détecter les enjeux qui doivent être modifiés ou éliminés. Au chapitre 2.4.2, on a expliqué la gouvernance des données et son rôle important dans la gestion du cycle de vie des données. Dans cette section, nous envisageons de présenter une stratégie basée sur des stratégies de chaîne d'approvisionnement verte ou durable pour proposer une méthode unique de gouvernance des lacs de données.

Afin d'établir une méthode analogique pour la gestion du cycle de vie des données, nous mobilisons le concept de gestion du cycle de la vie du produit [4] au sein de la chaîne d'approvisionnement en référence à des approches analogiques. Pour la première étape de cette analogie, nous nous référons à la durabilité [5], qui concerne les vertus d'un système ou d'une société qui pourrait maintenir ses compétences et ses caractéristiques présentes à long terme sans compromettre les générations futures [Heinberg and Lerch, 2010]. De manière générale, le développement durable concerne des procédures organisationnelles fondées sur la considération et la responsabilité environnementales afin de développer la performance des organisations sans avoir d'impacts négatifs sur l'environnement. Pour cette raison, des stratégies et des outils d'évaluation des systèmes sont proposés qui vérifient les processus logistiques d'un produit au cours de son cycle de vie afin d'améliorer la durabilité des organisations ou de la chaîne d'approvisionnement.

Dans le cadre de la gestion de chaîne d'approvisionnement durable ou verte, les stratégies de *gestion environnementale de la chaîne d'approvisionnement(GECA)* [6] sont appliqués. Le paradigme GECA contient toutes les stratégies, activités, objectifs et décisions liés à la structure, aux produits et aux procédures de la chaîne d'approvisionnement qui minimisent ou éliminent les effets négatifs sur l'environnement [Zsidisin and Siferd, 2001]. La pensée du cycle de vie des produits dans le système logistique est considérable pour la gestion environnementale afin de sensibiliser chaque phase de préparation et d'approvisionnement du produit aux enjeux écolo-

[4] Life Cycle Management (LCM)
[5] Sustainability
[6] Environmental supply chain management(ESCM)

giques. Ainsi, le code d'évaluation de cycle de vie [7](ECV) est généré pour justifier la pérennité du processus d'approvisionnement et de production du produit depuis la matière première jusqu'à la consommation, le recyclage, la reproduction ou la destruction. Le code 'évaluation de cycle de vie et son extensions tels que évaluation du cycle de vie du produit [8], évaluation du cycle de vie social [9],évaluation de la durabilité du cycle de vie [10], sont fondées sur des réglementations et des normes pour chaque phase de vie du produit afin de faire converger les considérations environnementales dans l'organisation et maximiser la durabilité du système en minimisant ou en éliminant les impacts antiécologiques tout au long de l'existence du produit.

Le code complet et les cadres de mise en œuvre de l'ECV sont définis par *"Organisation internationale de normalisation* [11] *"* sous la norme ISO 14040. Dans le cadre de l'ECV, on utilise les différents outils tels que les outils analytiques, les outils qualitatifs et de gestion, les outils de surveillance, les certification et l'outil de reportage pour évaluer les différents aspects du produit à chaque niveau de sa vie [Benoît et al., 2010]. Cet outil permet de mesurer et qualifier les points d'équivalence de qualité du produit tout au long de son approvisionnement. Selon [Muralikrishna and Manickam, 2017, Hauschild et al., 2018, Protocol, 2011, Heilala et al., 2014] ces outils sont appliqués à travers les phases suivantes :

- *Définition de l'objectif et des portées :* Déterminer soigneusement les aspects et les critères d'évaluation du produit, définir les limites des espaces de recherche, les hypothèses et les limites, les méthodes d'analyse, et les caractéristiques attendues pour le produit ou le service.
- *Cycle de vie d'inventaire :* Collecter des données et des informations sur toutes les entrées et sorties (ressources, matériaux, produits finaux) du système liés au processus de préparation du produit, lancer un inventaire des données d'information collectées et évaluer la qualité du produit selon des critères définis.
- *Évaluation de l'impact :* Sélectionner d'indicateurs et de méthodes pour évaluer les impacts des produits à partir des données d'inventaire, catégoriser et standardiser les qualités des produits en fonction des résultats des études d'impact.
- *Interprétation :* Interpréter les résultats obtenus, mettre en œuvre les points significatifs pour développer la qualité du produit ou du service, définir les stratégies de prise de décision et identifier les opportunités d'amélioration du système.
- *Rapports et revue critique :* Générer des rapports selon la phase d'interprétations et déterminer des revues critiques, des assurances ou des certificats pour augmenter la connaissance des utilisateurs du système.

Les stratégies de gestion de la chaîne d'approvisionnement verte et les codes de gestion et d'évaluation du cycle de vie du produit sont considérés comme une source d'inspiration pour notre étude afin de définir un nouveau paysage (*landscape*) protocolaire pour la gouvernance

[7] Life cycle assessments (LCA)
[8] Product life cycle assessment(PLCA)
[9] Social life cycle assessment(SLCA)
[10] Life cycle sustainability assessment(LCSA)
[11] International Organization for Standardization (https://www.iso.org)

des données et lac de données. Sur la base de cette hypothèse selon laquelle les données sont considérées comme des produits, on pourrait imaginer que des données de mauvaise qualité ou inutiles ont des impacts destructeurs sur les performances et la fiabilité du lac de données ou à grande échelle, sur la réputation de l'organisation. Pour cette raison, les codes de gouvernance des données sont définis avec plusieurs normalisations, considérations et significations. Cependant, la gestion de la qualité des données doit être impliquée à tous les niveaux de préparation des données au sein des systèmes de stockage, comme expliqué dans la section 2.4.2. Cette remarque souligne l'importance de la réflexion sur le cycle de vie des données afin de mettre en place un cadre de gouvernance pertinent, précis, complet, cohérent et transparent.

Sur la base de cette idée, nous lançons une approche systémique basée sur des cadres ECV pour gouverner et démocratiser les données au cours de leur vie dans le lac de données en mettant l'accent sur la durabilité des systèmes de gestion des données. Nous définissons *la durabilité* ou *la pérennité* des systèmes de stockage de données comme *"la caractéristique d'un système qui évite de conserver, préserver, stocker, diffuser ou distribuer des données sans valeur, de mauvaise qualité, inutiles et sensibles qui ont des impacts négatifs sur le système ou qui sont confidentielles à partager, afin de protéger la pureté de l'état du système pour une utilisation actuelle et future."*

Le cadre de gestion du cycle de vie des données pour la gouvernance des lacs de données est proposé selon 5 phases essentielles de l'ECV qui sont définis ci-dessus, comme suit :

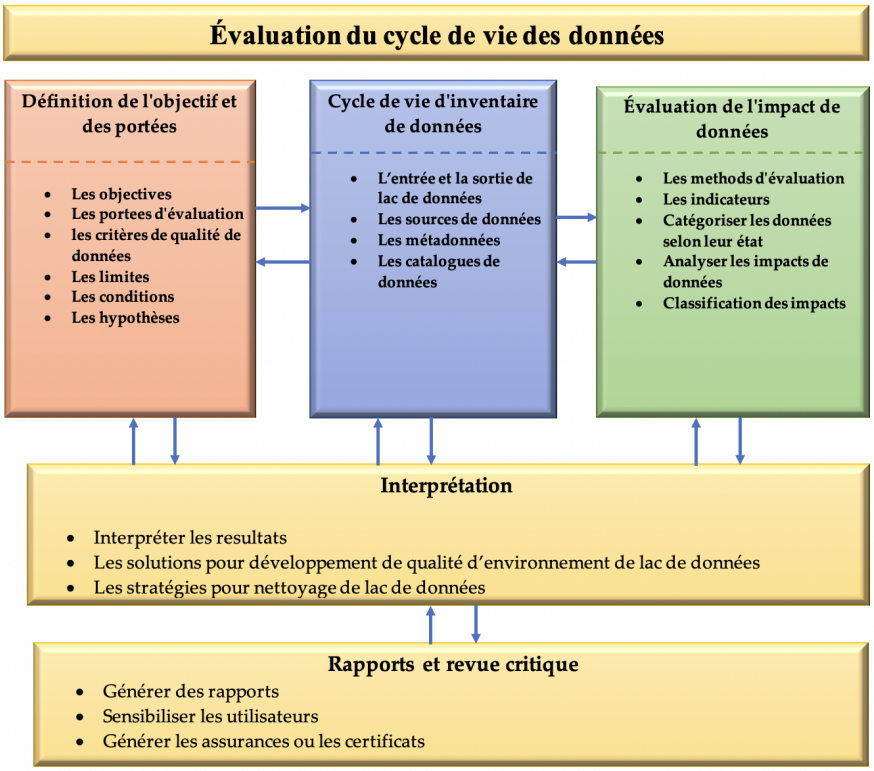


FIGURE 4.6 – Évaluation du cycle de vie des données

- *Définition de l'objectif et des portées :* Déterminer les objectifs et les critères d'évaluation

de données ainsi que préciser les contraint et les frontières pour analyser la qualité de données.

- *Cycle de vie d'inventaire de données :* Collecter et vérifier les informations sur les sources de données.
- *Évaluation de l'impact :* Sélectionner des indicateurs et des méthodes pour évaluer les impacts de différents types de données sur le lac de données et catégoriser les données et leur impact selon des standards définis et étudiés.
- *Interprétation :* Interpréter les résultats des étapes précédentes et définir les stratégies et protocoles pour démocratiser les données et gouverner le lac de données.
- *Rapports et revue critique :* Actualiser les rapports, les certificats et les assurances pour sensibiliser les utilisateurs et améliorer les évaluations ultérieures.

Les ***objectifs*** sont présentés ci-dessous :

- Maximiser la qualité, la fiabilité, la sécurité, la cohérence et la liaison des données au sein d'un lac de données ;
- Maximiser la durabilité du lac de données ;
- Minimiser la présence, le stockage et la distribution de données inutiles ou non pertinentes ;
- Minimiser le risque de marécages de données.

Les ***méthodes*** sont les suivants :

- Code exhaustif de gouvernance des données ;
- Outils et protocoles d'évaluation des données tout au long de leur vie (gestion du cycle de vie des données) ;
- Définition des critères et des revues critiques pour l'évaluation des données ;
- Normalisation et classification de la qualité des données ;
- Certificats et assurance des applications de gouvernance des données.

Conception d'architecture de lac de données

Définition :

La structure et le positionnement des entités des systèmes sont les impacts directs et décisifs sur la rendement globale. Du point de vue des systèmes logistiques, la conception et la configuration du canal du participant est une planification stratégique qui a des effets durables sur les entités locales et la performance globale. La conception du réseau de la chaîne

d'approvisionnement qui est en quelque sorte des stratégies de gestion de la chaîne. Cette stratégie aborde des problèmes d'optimisation singuliers ou hybrides qui recherchent un équilibre entre le nombre et l'emplacement des installations ainsi que des décisions tactiques concernant des problèmes d'environnement déterministes ou incertains, des dispositions disponibles et des contraintes potentielles dans la chaîne d'approvisionnement. La fourniture [Santoso et al., 2005, Eskandarpour et al., 2015]. Du point de vue des systèmes d'information, l'architecture du lac de données, en tant que partie importante des systèmes d'aide à la décision, est un avantage concurrentiel qui attire l'attention de nombreuses recherches et études dans ce domaine. La conception d'un pipeline de lac de données est un enjeu essentiel pour optimiser les performances et la rentabilité de ce système de gestion de données centralisé. Malgré les études considérables menées dans le développement de la fondation et de la plate-forme du lac de données que nous avons abordées dans le chapitre 2, elles restent encore pleines d'opportunités de recherche pour concevoir une infrastructure dédiée au lac de données optimisé et rentable.

Sur la base de la méthodologie analogique que nous avons choisie pour cette étude, on pourrait s'appuyer sur les approches imitées de la conception de réseaux de chaîne d'approvisionnement pour proposer une nouvelle architecture logistique pour le lac de données. Cette méthodologie nous permet d'élaborer une structure en couches et de détailler la fonctionnalité de chaque couche dans des fonctions de réflexion logistique afin d'optimiser l'efficacité du lac de données. Nous aborderons en détail cette architecture et les méthodes d'optimisation dans les chapitres 5, 6, et 7.

Les ***objectifs*** sont :

- Concevoir une architecture logistique pour minimiser les coûts locaux et globaux du système et maximiser la qualité de service ;
- Optimiser les performances du lac de données.

Les ***méthodes*** sont les suivants :

- Canaliser l'architecture du lac de données ;
- Utiliser des méthodes de conception de la chaîne d'approvisionnement pour structurer le pipeline du lac de données ;
- Mettre en œuvre les problèmes de décisions conjointes (stratégiques, tactiques et techniques) de gestion de la chaîne d'approvisionnement ;
- Détermination de la fonctionnalité optimale des éléments du lac de données selon des méthodes d'optimisations imitées des systèmes logistiques.

4.3.4 Fonctions objectifs

Dans le cadre de la gestion des membres des systèmes ou de l'optimisation des performances, on définit les objectifs locaux et globaux à atteindre en fonction des stratégies définies pour

le système. L'optimisation des performances des systèmes est obtenue sur la base de fonctions objectifs qui représentent les variables associées aux problèmes traités et qui doivent être rendues optimales. Un problème d'optimisation peut contenir une ou plusieurs fonctions objectifs [Savic, 2002]. Dans les systèmes logistiques, des fonctions objectifs singulières sont définies. Il s'agira par exemple de minimiser les coûts, de minimiser les délais de la logistique des produits, ou encore de maximiser les profits et les services aux clients [Mirjalili, 2019]. Les fonctions objectifs multiples concernent des problèmes d'optimisation qui prennent en compte des fonctions objectifs contradictoires telles que minimiser les coûts totaux tout en maximisant la rentabilité des services [Fernandes et al., 2014, Contreras et al., 2012, Cornuéjols et al., 1983, Pasandideh et al., 2013].

Pour le problème d'optimisation du lac de données, les fonctions objectifs singulières peuvent être par exemple définies comme étant basées sur la minimisation des temps d'exécution, la minimisation des coûts de démarrage, la maximisation des bénéfices et la maximisation de la couverture des demandes des utilisateurs, ainsi que de multiples fonctions objectifs telles que la maximisation du niveau de service en minimisant le temps d'exécution ou maximiser la durabilité du système en minimisant les coûts totaux de démarrage [Zagan and Danubianu, 2021, Rohde and Vidal, 2020, Herodotou and Babu, 2011].

4.3.5 Variables de décision

Les variables de décision sont les valeurs nécessaires à optimiser à travers les résultats des modèles d'optimisation (les fonctions objectifs et les contraintes) [Cooper, 1963]. Dans le cadre de la gestion de la structure de la chaîne d'approvisionnement, ces valeurs pourraient être définies comme le nombre optimal d'installations et la manière dont ils sont attribués à d'autres installations ou clients, les paramètres de contrôle des stocks (quantité de commande, quantité de sécurité des magasins, délai d'administration, ..) et le nombre de niveaux de la chaîne (couches) [Azarmand and Neishabouri, 2009, Perea-Lopez et al., 2003, Al-Othman et al., 2008].

En considérant le lac de données comme un système logistique, des variables de décision sont marquées telles que les nombres optimaux de Jobs (au sens des traitements informatiques) effectués pour traiter les demandes, le temps d'exécution, les nombres ou types de logiciels mis en œuvre ou la planification des Jobs de manière qui satisfait les objectifs locaux et globales du lac de données [Herodotou and Babu, 2011, Peyravi and Moeini, 2020].

4.3.6 Contraintes

Les contraintes sont les limitations des problèmes d'optimisation qui s'appliquent aux fonctions objectifs ou aux variables de décision. Par exemple en général, les systèmes organisationnels sont restreintes par le budget, le nombre d'installations à lancer, la capacité de production, les commandes ou les livraisons, ou les délais de mise en œuvre et de service [Chen and Lee, 2004, Mahnam et al., 2009]. Analogiquement, les contraintes du lac de données sont liées aux capacités de mémoire ou de CPU, de gravité des données, de protocoles et réglementations de gouvernance et de gestion des données, ou encore à des contraintes techniques de logiciel ou de programmation [Madera, 2018, Thomas, 2006, Yebenes and Zorrilla, 2019].

4.3.7 Risque

La gestion des risques est un enjeu indispensable dans le développement de systèmes pour préparer et mobiliser le système contre les incertitudes externes et internes [Ho et al., 2015]. Tous les évènement prévus ou imprévus en lien avec l'environnement extérieur ou procédure intérieure de réseau qui pourront interrompre ou détruire la productivité de système, sont traités comme les risques nécessaires à gérer. Le domaine de la gestion de risques pour les structures systémiques est assez large, mais par exemple on peut indiquer quelques risques courants qui influencent souvent la performance du système telles que le risque de production de produits défectueux, les pertes de clients et les ventes manquées, les demandes imprévues et les risques environnementaux (catastrophe naturelle) [Linkov and Palma-Oliveira, 2017, Ho et al., 2015]. Pour le système d'information, ces risques se répartissent en risques de panne machine, panne d'accessibilité, qualité et sécurité des données, risques de cyberattaque et risques de perte et de mapping des données [Saed et al., 2018, Rasooli and Down, 2014, Bamrara, 2015].

4.3.8 Mesure de la performance

La mesure de la performance ou du rendement est une étape de grande importance qui évalue et estime la productivité et l'efficacité des stratégies réalisées en achevant les objectifs définis de l'organisation au cours de l'horizon défini. Pour cette raison, des métriques et des mesures quantitatives ou qualitatives sont exploitées pour mesurer l'adaptation et les ajustements de la structure et des éléments de la chaîne d'approvisionnement (au niveau local et global) vers des caractéristiques fructueuses et avantageuses [Gunasekaran et al., 2001, Huan et al., 2004]. Au-delà de la rentabilité et du haut niveau de qualité de service qui sont deux vertus importantes pour une chaîne d'approvisionnement efficace, ces deux caractéristiques ne sont néanmoins pas suffisantes pour développer un système logistique à haut niveau de valeur compétitive dans l'environnement concurrentiel actuel. Par exemple, [Closs and McGarrell, 2004] a défini cinq **"V"** pour manifester les caractéristiques importantes du succès de la chaîne d'approvisionnement telles que *la valeur* [12] qui indique la rentabilité du système (elle doit être augmentée), *la velocité* [13] qui note la rapidité du temps de réponse de la chaîne d'approvisionnement (il doit être augmenté), *la variabilité* [14] qui présente la large gamme de produits ou services fournis (il doit être diminué), *la visibilité* [15] qui indique la transparence de l'information entre les participations (elle doit être augmentée), et *la vulnérabilité* [16] qui montre la tolérance du système aux cas destructeurs (elle doit être réduite).

[12] Value
[13] Velocity
[14] Variability
[15] Visibility
[16] Vulnerability

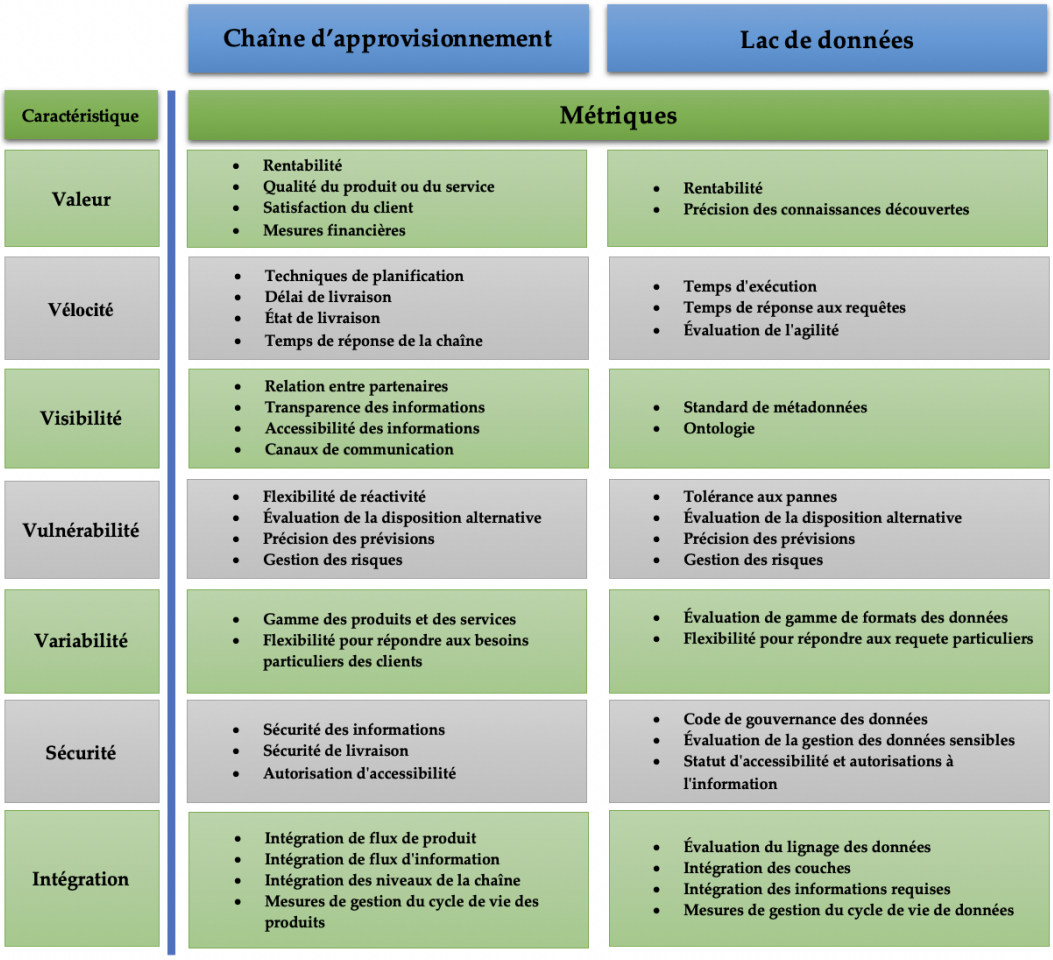


FIGURE 4.7 – Les métriques d'évaluation de la performance

En plus de ces caractéristiques qui précèdent, la sécurité, la résilience, l'intégration de l'information et la coordination des partenaires sont des vertus remarquables à considérer pour un système logistique réussi. Afin d'évaluer ces caractéristiques au sein du ce système, il est nécessaire de définir les métriques appropriées correspondant à chaque niveau d'approvisionnement et d'acheminement du produit. [Gunasekaran et al., 2004] propose les métriques selon le contexte des activités de la chaîne qui sont catégorisées au niveau de contrôle d'inventaire (la quantité de commande, le temps de mise en œuvre, la maintenance et le transport des commandes), au niveau d'approvisionnement (évaluation des fournisseurs, conditions d'approvisionnement), au niveau de la fabrication des produits (qualité de service, planification de la production, gamme de produits,...), au niveau de la livraison (rapidité, itinéraires et satisfaction des clients), au niveau relation avec les partenaires et au niveau de la phase de post-consommation (services après-vente).

Il est évident que les caractéristiques indiquées ci-dessus sont potentiellement applicables pour toutes les structures qui fonctionnent en termes de pensée systémique. Il existe des points communs entre les valeurs importantes d'un système logistique et d'un système de gestion de

données en tant que lac de données, telles que la vélocité, la sécurité, la résilience, la visibilité, l'intégration, la fiabilité et la valeur [Sundaram and Vidhya, 2016, Ravat and Zhao, 2019, Munshi and Mohamed, 2018]. Il est nécessaire qu'un système de mesure convivial ou imité soit en place pour évaluer et estimer les mesures de performance du lac de données au niveau de chaque couche afin d'améliorer la fonctionnalité actuelle et d'étendre l'utilisation future. Nous pouvons proposer un nouveau système métrique pour mesurer les performances d'un lac de données en utilisant une logique de comparaison de systèmes qui est présenté sur la figure 4.7

4.4 Lac de données et écosystème

Dans cette section, nous nous concentrons sur le deuxième système qui pourrait être une source d'inspiration pour la gestion des lacs de données, "***écosystème***". Étant donné que le nom de lac de données est dérivé d'un lac naturel où vivent de nombreuses espèces avec différents organismes, l'écosystème naturel est un modèle précieux pour l'extraction de méthodes imitées aux fins de cette étude. Pour atteindre cet objectif, nous allons réviser une nouvelle fois les modules des tableaux de comparaison, élément par élément, sous une approche analogique de lac de données et d'écosystèmes.

4.4.1 Membres/Niveaux

Le lac naturel est un écosystème naturel qui héberge une vaste gamme d'espèces vivantes différentes telles que des animaux, des plantes, des micro-organismes, etc. Ces membres des systèmes naturels interagissent intrinsèquement, influencent le système avec leurs fonctions communes (reproduction et survie) ou différentes (méthode nutritionnelle) et sont affectés les uns par les autres [Trivers and Dawkins, 1976]. Malgré les différents organismes qui distinguent les espèces de différentes familles, ils sont interdépendants et chacun a des impacts spécifiques sur un autre membre ou à grande échelle sur l'ensemble de l'écosystème, au cours du cycle de vie. Par conséquent, tous les organes vivants sont mobilisés automatiquement pour faire évoluer le système naturel et surtout pour conserver ses vertus importantes telles que l'homéostasie et la résilience [Morgan Ernest and Brown, 2001]. L'ontologie des espèces vivantes d'un écosystème définit les régulations internes et externes afin de répondre aux objectifs instinctifs du système naturel (lac naturel) et de générer le produit vital des écosystèmes qu'est la biodiversité.

Selon l'analogie entre le lac de données et le lac naturel, les données en tant que produit principal des systèmes de gestion de données, passent par différents niveaux de maturité jusqu'à l'extraction des connaissances. Dans le lac de données, les couches de gestion et de préparation des données sont considérées comme les différentes composantes et niveaux du lac naturel où chaque couche est responsable de la création de valeurs liées à une phase spécifique de la durée de vie des données en tant que processus biologique depuis la naissance des éléments jusqu'à leur décès.

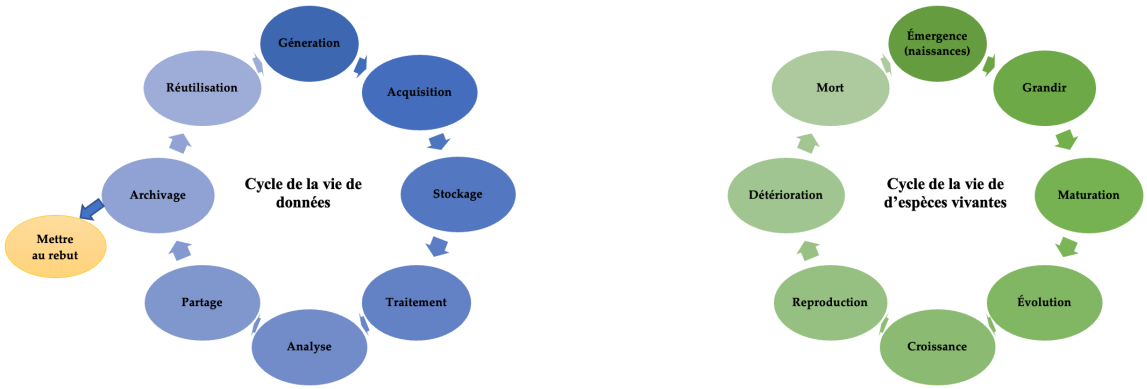


FIGURE 4.8 – Cycle de vie en lac de données et en nature

Les membres des systèmes naturels (espèces vivantes), comme tous les participants aux structures systémiques, passent par des étapes de maturité dans le cycle de la vie. Selon la figure 4.8, dans le système biologique, ce cycle commence par la naissance d'espèces vivantes qui grandissent et évoluent au cours de leur vie. Selon la théorie de l'évolution, les plus puissants et les plus adaptés ont les meilleures chances de rester en vie et de reproduire la nouvelle progéniture. Le cycle biologique se termine par la détérioration et la mort des organes et la transformation en d'autres formes de l'espèce naturelle [Beyer and Schwefel, 2002]. Les étapes de maturité cyclique s'appliquent également aux données à différents niveaux, qui commencent par l'acquisition, le stockage, la récupération des données, puis l'archivage des données validées ou l'élimination des données invalides. Comme nous l'avons indiqué dans les sections précédentes, la gestion du cycle de vie des données est devenue une approche essentielle pour la valorisation des informations dérivées. C'est pourquoi dans cette section nous nous appuierons cette fois sur les stratégies naturelles comme méthodes mimiques afin de trouver des solutions innovantes pour gérer le cycle de vie des données à chaque phase de maturité.

4.4.2 Produit

La biodiversité des espèces, la biomasse, les séquences d'ADN et les complexités biologiques sont les produits les plus importants des systèmes naturels qui sont générés ou préservés avec certains mécanismes biologiques [Cottingham et al., 2001, Oliver et al., 2015]. Les produits de l'écosystème sont diversifiés selon les niveaux d'organes, les fonctions écologiques et les processus biologiques. De même, les données du lac de données pourraient être considérées comme les espèces vivantes du lac naturel qui ont des formats et des structures similaires ou totalement différents, pourtant elles sont hébergées dans le référentiel centralisé afin d'engendrer un système complet de gestion de données massives. Les données, comme les espèces vivantes sont diversifiées selon leur domaine (données biologiques, spatiales, géographiques, financières, médicales,..) ou selon leurs ressources (données de capteurs, données web, données générées par l'homme ou la machine), ou selon leurs structures (données structurées, non structurées, semi-structurées). La Diversité des données et niveaux de granularité ; de même genre de la biodiversité des espèces, sont des raisons importantes de la complexité des systèmes de gestion des données actuels.

4.4.3 Stratégie de gestion

À mesure que la complexité du système augmente, le besoin de stratégies efficaces pour le gérer devient plus évident. En termes de systèmes biologiques, ces stratégies sont classées en mécanismes d'évolution des espèces (tels que les fonctions de recombinaison, de mutation, de dérive génétique et de sélection la plus adoptée), de compétition, de parasitisme de mutualisme et de prédation [Combes, 2001, Boucher et al., 1982]. Les stratégies naturelles sont la grande source d'orientation vers des approches innovantes dans les domaines informatiques [Beyer and Schwefel, 2002, Yang and He, 2020]. Par exemple, les processus d'évolution des espèces tels que l'opérateur de sélection, les mutations ou les recombinaisons, s'inspirent pour développer des algorithmes heuristiques ou méta-heuristiques dans le domaine de l'optimisation des problèmes ou dans le cadre de l'IA pour la programmation informatique (codage) [Yu and Gen, 2010, Talbi, 2009, Fister Jr et al., 2013]. Sur la base de cette cognition, les mécanismes naturels qui sont employés par la nature pourraient être employés dans le lac de données tels que :

- Stratégies de gestion des taux d'entrée et de sortie des écosystèmes en tant qu'outil de mimétisme pour la gestion des ressources et des sorties de données ;
- Stratégies pour évaluer les membres de ce système biologique (l'espèce), en tant que stratégie pratique pour l'évaluation de la qualité des données ;
- Mécanisme de sélection des espèces plus ajusté (niveau élevé de fitness) comme mécanisme de gouvernance des données pour garder les données plus valides ;
- Mécanisme d'effacement des espèces plus faibles, défectueuses et imparfaites (prédation et parasitisme) comme une méthode de suppression ou d'affaiblissement des données en mauvaise qualité ou inutiles [Bush et al., 2001] ;
- Stratégies de l'interaction entre d'organismes (Mutualisme) comme les méthodes pour la mise en œuvre l'intégrité entre les différentes couches du lac de données afin d'optimiser ses performances ou d'améliorer les approches de découverte des connaissances [Boucher et al., 1982] ;
- Stratégies pour la résilience du système naturel contre des événements perturbateurs imprévus comme des mesures de gestion pour atténuer la vulnérabilité du système informatique auprès des scénarios destructeurs [Chatterjee and Layton, 2020b].

Stratégies de gouvernance de données

Les régulations biologiques qui intrinsèquement impliquées pour discipliner des espèces et contrôler des équilibres écologiques, pourraient être vues comme l'assortiment de règles pour gouverner des systèmes de données brutes chaotiques. Les stratégies naturelles comme, la gestion des taux d'entrée et de sortie, la gestion de flux des énergies, les mécanisme de sélection des espèces plus adaptées ou d'effacement des espèces plus faibles, et les stratégies de gestion de l'interaction entre d'organismes (Mutualisme), nous font penser à proposer les méthodes mimiques issues du système biologique comme stratégies de gestion de données massives en lac de données [Boucher et al., 1982, Bush et al., 2001, Westman, 1978, Dessalles et al., 2016]. En

comparant l'écosystème au lac de données, on pourrait tirer parti des réglementations biologiques comme les stratégies de gouvernance des données dans lac de données présentées dans la figure 4.9 :

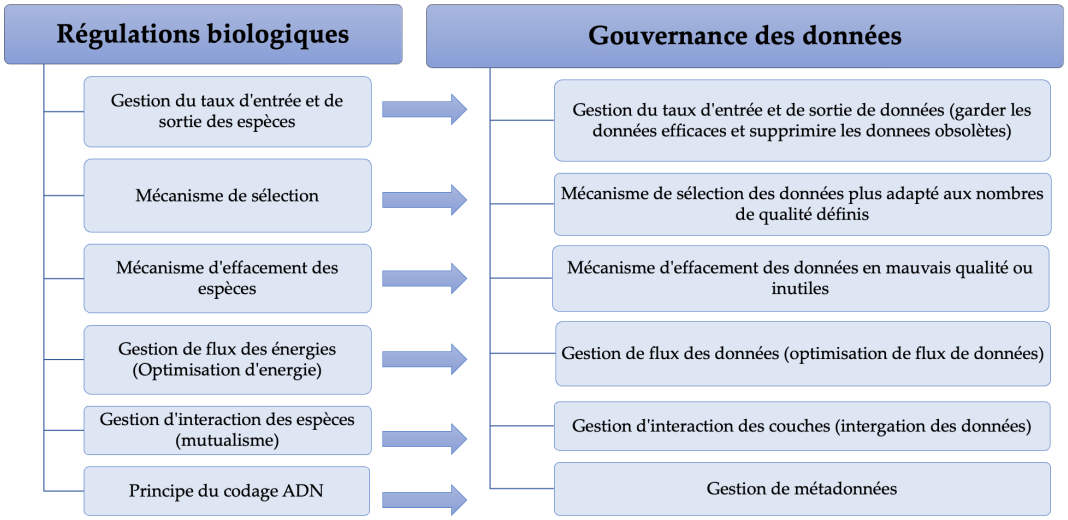


FIGURE 4.9 – Réglementations biologiques comme les stratégies de gouvernance de données

Pour clarifier le sujet, on considère les organismes vivants et en particulier l'ADN et la brique de base, le gène qui contient l'information des espèces vivantes. La richesse des informations dépend des activités des lectures, il s'agit que les données qui ne sont pas lues par les lectures sont affaiblies et seront supprimées ou détruites au fil du temps. Au contraire, si les données sont utilisées et lues en continu, elles sont renforcées et seront rendues permanentes dans le système naturel. En revanche, on ne peut pas négliger l'effet de la chance sur la multiplication et la génération automatique de nouvelles données pour équilibrer cette caractéristique. Cette vertu naturelle fait penser à des mécanismes mimétiques de gouvernance du lac de données. De la même manière, si les données dans les systèmes informatiques ne sont pas très utilisées ou valorisées par les utilisateurs, elles menacent la véracité du lac de données par le marécage de données inutiles. De plus, les données seront rendues éligibles et précieuses si elles sont davantage utilisées pour des procédures de récupération.

Les fonctions de régulation interne de l'écosystème, ainsi que les stratégies de gestion des espèces vivantes, maximisent la survie de ses membres les plus forts et les plus adaptés et garantissent l'évolution du système dans un état équilibré. Parmi les fonctions internes de la nature, les mécanismes de sélection naturelle sont les plus pertinents pour être employés comme source d'inspiration pour les stratégies de gouvernance des lacs de données. Les stratégies de sélection naturelle augmentent les adaptations des membres de l'écosystème et garantissent la persistance, la stabilité génétique et environnementale par la reproduction du génome (mutation, changement aléatoire, copie, combinaison,...). Ces fonctions et régulations naturels pourraient intervenir dans les domaines de la gouvernance des lacs de données pour discipliner et fortifier les données éligibles comme méthodes interdisciplinaires. Les figures 4.10 et 4.11 montrent la réalisation de mécanismes naturels comme stratégies efficaces pour gouverner le lac de données.

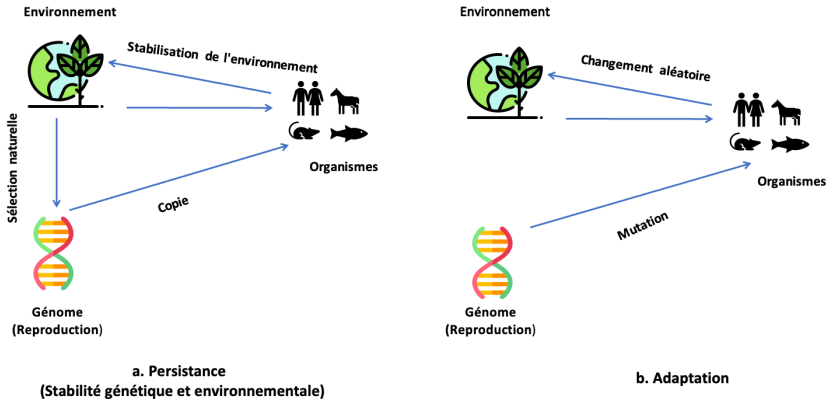


FIGURE 4.10 – Stratégies naturelles pour la gouvernance des membres

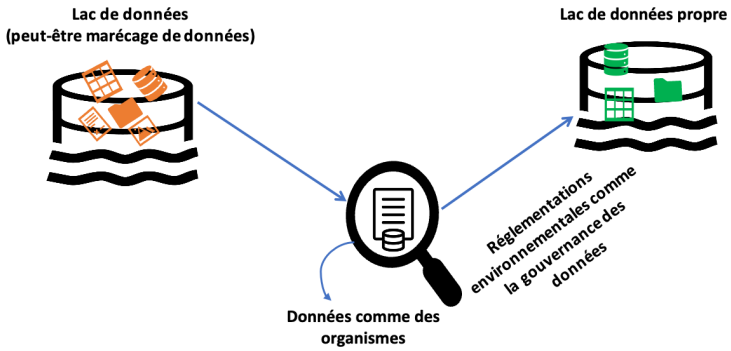


FIGURE 4.11 – Stratégies naturelles pour la gouvernance du lac de données

Stratégies de stabilité

Le population d'espèce et la relation entre eux influencent directement sur la performance et la fonction d'écosystème. Les changements prévues ou imprévu du système pourrait avoir grand impacts sur la balance nécessaire entre le taux d'entrée d'énergie (approvisionnement) et le taux de sorties (consommation). Cette stabilité est garantie par l'homéostasie qui maintient l'état normal du système et conserve l'équilibre du taux d'énergie même dans des situations anormales [Morgan Ernest and Brown, 2001]. L'homéostasie favorise l'écosystème pour maintenir sa tendance à rester dans un état stable [Lerner et al., 1954, Tschirhart, 2000]. Par conséquent, cette propriété est une caractéristique importante pour toute structure systémique qui pourrait être copiée du système naturel.

La stabilité et la tendance à préserver la situation stable sont l'objectif essentiel de tous les systèmes informatiques confrontés en permanence au risque d'anomalies. L'homéostasie dans système de gestion de données comme lac de données pourrait être considéré comme toutes les

efforts et les stratégies dérivées de nature pour minimiser l'écart grave entre l'état normal et l'état instable. Dans le lac de données, la préservation de la stabilité se traduit par un taux de données équilibré, un temps d'exécution approprié, un taux de réponse favorable, une tolérance aux pannes normale, un débit stable des serveurs, une compatibilité des structures de données, l'intégrité entre les couches et les données, et les tentatives standard et validées des utilisateurs qui devront rester durable aux changements éventuels du système.

Stratégies résilientes

Une des stratégies naturelles qui pourrait être appliquée par plusieurs domaines est la stratégie de renforcement de la résistance et de la durabilité des systèmes telles que les stratégies résilientes. Compte tenu de section 4.3.3, la qualité et le niveau de résilience des systèmes informatiques sont si importants que les stratégies logistiques ne pourraient pas être les seules méthodes suffisantes pour atténuer la vulnérabilité de ces types de systèmes. Pour cette raison, on peut profiter de stratégies provenant d'autres sources d'inspiration interdisciplinaires telles que les stratégies résilientes de la nature.

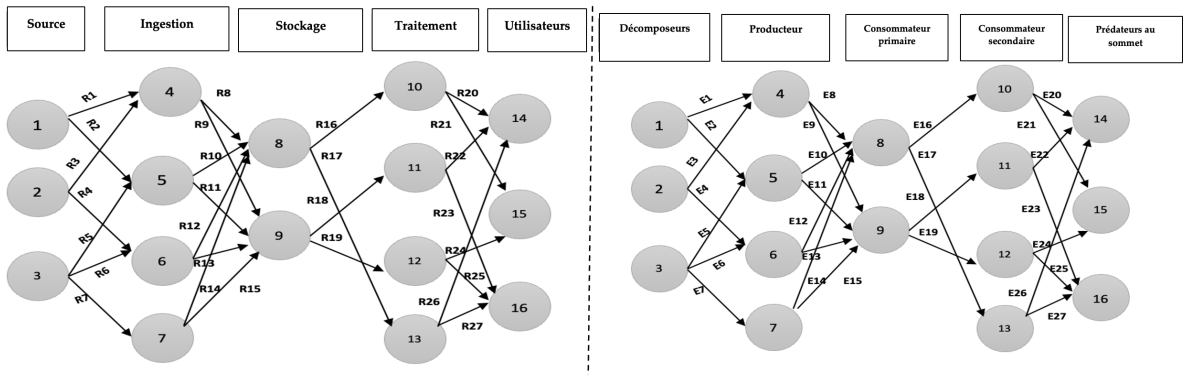
Dans écosystème, la capacité de résilience est considérée comme l'une des propriétés écologiques essentielles qui préserve et protège les fonctions homéostatiques avant les événements interrompus et reconstruit ou restaure rapidement et efficacement le système à l'état stable et initial après les événements perturbateurs d'origine naturelle ou humaine [Holling, 1973]. Malgré la difficulté de déterminer les mesures pour évaluer la résilience d'un système naturel, [Westman, 1978] propose les caractéristiques du système résilient. Ces caractéristiques qui sont montrés dans la figure 4.12, pourraient également être comprises comme des mesures pour qualifier la qualité de la résilience dans d'autres systèmes tels que le lac de données.

	Écosystéme	Lac de données
Caractéristique	Défination	
Inertie	Résistance du système aux anomalies ou instabilités telles que le changement climatique et la tendance à rester dans l'état normal initial	Capacité à maintenir l'état normal du système contre les événements perturbateurs comme les cyberattaques.
Élasticité	Capacité à restaurer le système rapidement après des interruptions (temps de récupération)	Faculté du système informatique qui pourrait redémarrer et rétablir immédiatement le système à l'état normal.
Amplitude	Seuil de fragilité d'écosystème qui pourrait résister contre les changements graves.	Gamme de durabilité du système contre les événements perturbateurs sans perte d'informations ou de sécurité après un scénario destructeur.
Malléabilité	Degré de similitude entre l'état initial avant l'interruption et le nouvel état après les perturbations.	Degré de compatibilité du lac de données pour restaurer le système à l'état identique ou presque similaire à l'état d'origine avant la panne

FIGURE 4.12 – Métriques pour mesurer la résilience des systèmes naturels et informatiques

Comme nous avons indiqué dans la figure 4.12, les métriques associées à l'évaluation de la vulnérabilité du système biologique sont pleinement applicables en tant que critères pour mesurer la résilience de l'infrastructure informatique du lac de données. Sur la base de cette idée, des métriques prédéfinies telles que, *Inertie* [17] qui fait référence à la résistance aux anomalies, *Élasticité* [18] qui fait référence au système de temps de récupération, *Amplitude* [19] qui est le seuil du système contre les changements internes ou externes du système, et *Malléabilité* [20] qui est le taux à lequel l'état final du système reste similaire à l'état initial après un dysfonctionnement, sont identiques à utiliser pour analyser la résilience dans toutes les structures systémiques, en particulier le lac de données qui est le système considéré dans cette étude.

Au-delà des caractéristiques normalement utilisées pour déterminer le niveau de résilience du système, des outils plus précis sont définis pour évaluer la vulnérabilité des structures systémiques issues de méthodes biologiques. Ces outils sont basés sur la relation et les interactions des membres ou des niveaux du réseau, que ce soit les espèces de l'écosystème ou les couches du lac de données, qui pourraient influencer la qualité résiliente du système [Chatterjee and Layton, 2020b]. Dans l'écosystème, la structure du réseau comme les réseaux trophiques [21], montre clairement la relation et l'impact de chaque membre avec et sur les autres acteurs. La structure en réseau du réseau trophique est basée sur la théorie des graphes qui contient des nœuds qui représentent qui mange quoi et les arcs qui montre le flux de la relation alimentaire [Pimm et al., 1991]. De manière analogique, nous pouvons illustrer que le lac de données en tant qu'architecture de réseau obéit également aux principes de la théorie des graphes où les nœuds représentent les couches (sources, ingestion, stockage, traitement et accès) et les arcs indiquent les flux de données au sein du lac de données. La figure 4.13 montre les similitudes entre ces deux réseaux basés la théorie des graphes.



a) **Graphe de réseau du lac de données** b) **Graphe du réseau trophique**

FIGURE 4.13 – Structures de réseaux systémiques basées sur la théorie des graphes

[17] Inertia
[18] Elasticity
[19] Amplitude
[20] Malleability
[21] Food web

Notations de la théorie des graphes		
Notation	Réseau trophique	Lac de données
Nœud	Espèce	Couches
Arc	Relation d'espèce	Flux de données
Noeud(s) de départ	Organismes producteurs	Sources des données
Nœud(s) de fin	Prédateurs puissants	Utilisateurs
Métrique du taux de flux	Énergie alimentaire (E)	Taux de données(mbps)(R)
Marche (Une séquence de nœuds et d'arcs)	Qui mange qui	Quoi se transforme en quoi

TABLE 4.3 – Notations de la théorie des graphes pour les structures systémiques

Sur la base du schéma de la figure 4.13, la table 4.3 montre les notations de la théorie des graphes pour deux structures systémiques.

Cette comparaison nous amène à utiliser des stratégies bio-inspirées pour concevoir l'architecture du lac de données résilient en tirant parti des outils qui sont utilisés pour mesurer la résilience dans le système naturel. Afin d'analyser les systèmes écologiques comme le réseau trophique et de surveiller la fonctionnalité de leurs principaux membres, de nombreux outils et stratégies sont proposés par les écologues. Parmi ces outils, l'Analyse de Réseau Ecologique [22] est plus pratique, que ce soit pour l'étude des écosystèmes, ou pour la recherche de systèmes bio-inspirés [Fath et al., 2007, Westman, 1978, Ulanowicz, 2004]. L'Analyse de Réseau Ecologique fournit des mesures écologiques telles que la fonction de aptitude [23] écologique pour évaluer les caractéristiques naturelles et les fonctions d'aptitude telles que la résilience, en termes d'interaction des membres du réseau dans le contexte de la structure du graphe. Ces mesures biologiques pourraient être applicables pour analyser et développer des niveaux de résilience dans d'autres structures systémiques comme un lac de données.

4.4.4 Fonctions objectifs

Selon les principes de gestion des taux d'entrée et de sortie des écosystèmes ou de l'évolution des espèces vivantes, la minimisation des impacts liés aux risques ou de facteurs troublants externes ou internes, la maximisation de la survie et de la reproduction des membres les plus adoptés, la maximisation de la résistance (résilience) du système, la minimisation des instabilités, et la maximisation de flux d'énergies, la maximisation d'équilibre entre taux entrées et sortie, sont les objectifs globaux du système biologique [Dessalles et al., 2016, Oliver et al., 2015]. En comparaison avec le lac de données, ces fonctions objectives sont appliquées pour la minimisation du stockage de données invalides (l'omission des données les moins appropriées), la minimisation des cas instables, la maximisation d'équilibre de flux de données, ou la maximisation de la

[22] Ecological Network Analysis
[23] Fitness function

résilience des systèmes contre les événements perturbateurs (stratégies de réseau trophique dans l'écosystème) [Chatterjee and Layton, 2020a].

4.4.5 Variables de décision

La nature utilise souvent automatiquement les meilleures stratégies intelligentes pour mettre en balance ses éléments interdépendants et optimiser les conditions des écosystèmes. Afin d'atteindre les objectifs indiqués, certaines valeurs doivent être contrôlées par la nature de manière spontanée. L'homéostasie est une vertu biologique qui fait référence à la capacité ou à la tendance des organes d'espèces ou d'autres écosystèmes à conserver un état équilibré, optimal et stable en ce qui concerne les effets externes ou internes [Morgan Ernest and Brown, 2001]. La nature utilise des "réglementations" pour garantir l'homéostasie et la résilience du système biologique [Berntson et al., 2017]. Pour cette raison, l'homéostasie est un paramètre important à optimiser pour la stabilité et la santé du système qui est lié au nombre d'espèces, au nombre de régulations internes et externes ainsi qu'il influence la complexité de l'écosystème.

Comme le montre le tableau 4.2, l'homéostasie pourrait être considérée comme une variable de décision dans le contexte de l'optimisation du système naturel. En revanche, dans le lac de données en plus de l'autre variable de décision définie dans la section précédente, on pourrait mettre en comparaison l'homéostasie dans la nature avec la tendance du lac de données à avoir un état normal et stable sous pression de données hétérogènes.

4.4.6 Contraintes et risques

Les contraintes et les risques prévus ou imprévus, restreignent les fonctions objectifs à réaliser. Les perturbations avec des agents internes ou externes tels que d'autres systèmes logistiques ou informatiques, et les changements globaux sont de sérieuses imitations pour l'harmonie des écosystèmes et imposent des limites aux interactions optimales entre les espèces [Tylianakis et al., 2008, Kane, 1997]. Pour cette raison, ces contraintes doivent être prises en compte pour la meilleure compréhension et analyse du fonctionnement du système. La nature emploie des stratégies de gestion des risques ou de dépassement de contraintes. Par exemple, on pourrait se référer aux opérateurs de mutation, à la recombinaison, à la dérive génétique et à la sélection comme une stratégie pour diminuer le risque d'existence d'espèces anormales ou éliminer les pertes destructrices, ou des mécanismes pour équilibrer les taux d'entrée et de sortie ou des méthodes d'auto-guérison contre les effets dévastateurs [Trivers and Dawkins, 1976].

Les solutions basées sur la nature sont toujours considérées comme les meilleures stratégies imitées pour la gestion d'organisations ou de structures systémiques avec de beaux éléments interdépendants. Par conséquent, dans le contexte de la gestion des risques des lacs de données, ces solutions basées sur la nature pourraient permettre aux scientifiques ou aux entreprises de développer et d'améliorer les performances des systèmes informatiques, y compris les systèmes de gestion de données en tant que sujet tendance.

4.4.7 Mesure de la performance

A l'instar des systèmes logistiques, la résilience et la richesse des produits est l'une des caractéristiques importantes pour parfaire la qualité des performances des systèmes [Karp et al., 2011]. De même pour les écosystèmes ces vertus sont mises en évidence et les biologistes tentent de définir les métriques pour mesurer l'état de perfection de ces caractéristiques. Par exemple, les indicateurs qui sont utilisés pour évaluer la résilience des systèmes écologiques pourraient être classés comme la diversité des espèces, la biomasse, la proposition de couverture des espèces, la température, la complexité structurelle, les nutriments (pollution), et l'impact physique humain [McClanahan et al., 2012]. En référence aux métriques logistiques imitées pour l'évaluation de la performance du lac de données, ces métriques et indicateurs biologiques sont de bonnes sources d'inspiration pour la mesure de la fonctionnalité des systèmes informatiques. Par exemple, des indicateurs biologiques tels que la diversité des espèces, la complexité structurelle et l'impact physique humain pourraient être transformés en diversité des données (sources ou domaines), complexité structurelle des données pour le stockage ou le traitement, et impacts externes sur la vitesse, la sécurité et la fiabilité du système, respectivement.

La méthode analogique abordée dans ce chapitre nous a conduit à réfléchir à plusieurs stratégies interdisciplinaires, qu'elles soient logistiques ou naturelles, pour la gestion du lac de données. En revanche, l'objectif principal de cette étude s'est contenté d'optimiser les performances du lac de données et il s'agit de savoir comment ces stratégies interdisciplinaires pourraient intervenir comme méthodes efficaces pour optimiser la fonctionnalité des systèmes informatiques. Pour atteindre cet objectif, nous visons à choisir l'une des stratégies mimétiques discutées pour analyser et détailler son efficacité dans l'optimisation du lac de données. En effet, les systèmes logistiques jouent un rôle important dans l'optimisation des structures systémiques et la modélisation des modèles mathématiques. Pour cette raison, nous considérerons les stratégies de gestion des systèmes logistiques tels qu'une chaîne d'approvisionnement comme un candidat pour les méthodes interdisciplinaires. Partant de cette idée, pour mettre en place des stratégies d'optimisation logistique, il faut d'abord définir une structure logistique pour le lac de données. Par conséquent, dans le chapitre 5, nous allons réaliser une architecture innovante pour le lac de données basée sur la définition du système logistique afin d'examiner les méthodes proposées pour l'optimisation de ce système de gestion de données.

4.5 Résumé

Dans ce chapitre, Dans ce chapitre, nous avons abordé le raisonnement analogique comme une méthode déductive pour révéler les points communs entre deux ou plusieurs phénomènes afin de tirer parti des avantages particuliers de chacun pour les autres. En appliquant cette méthode pour des structures systémiques comme les systèmes logistiques, naturels et informatiques, on atteint ces objectifs que les similitudes entre ces trois systèmes sont interchangeables afin de les mettre en œuvre l'un et les autres de manière efficace. La structure logistique du lac de données en tant que système de gestion de données est le fruit de cette approche analogique qui nous permet d'utiliser les stratégies et les outils pratiques des deux autres systèmes afin de parvenir à un système productif.

Chapitre 5

Architecture logistique du lac de données ALLD

"La structure est plus importante que le contenu dans la transmission de l'information."

– Abbie Hoffman 1936-1989

5.1 Introduction

Quand on parle d'architecture d'un système, on parle du processus de planification, de conception, d'harmonisation et de d'encadrement de sa structure en fonction des disciplines particulières. Cependant, à mesure que le nombre d'éléments et d'entités du système augmente et que sa fonctionnalité devient plus compliquée, la complexité du système impose chaos et désordre à la structure. Cette désorganisation structurelle devrait être agencée avec la bonne conception architecturale. Dans le cadre de la construction d'édifices, l'approche « Urbanisation » est utilisée pour mettre en ordre le chaos généré par la croissance des populations et les bâtiments modernes en tant qu'entités du système urbain [Bertinelli and Black, 2004].

Les systèmes d'information sont un bon exemple de structure complexe qui comprend de nombreux sous-domaines, notamment les systèmes informatiques, les logiciels, les informations, les données et les personnels. Dans le cadre de la conception d'une architecture efficace des systèmes d'information, des approches et méthodes pratiques sont proposées afin de discipliner les entités et faire face aux désordres potentiels de tels systèmes compliqués. Pour mener à bien ce travail, la démarche d'urbanisation a été mise en place afin d'aboutir à une architecture exhaustive et évolutive du système d'information et de ses sous-ensembles [Servigne, 2008]. Dans cette méthode, qui repose sur une simple analogie, les systèmes d'information sont considérés comme des agencements urbains dans lesquels la construction de leurs infrastructures obéit à des processus d'urbanisation. Sur la base de cette méthode innovante, [Madera, 2018] étend le processus d'urbanisation pour concevoir l'architecture réussie du lac de données comme un élément important des systèmes d'information. Elle s'est appuyée sur quatre étapes successives (métier, fonctionnelle, applicative et technique) de la méthode d'urbanisation proposée par [Servigne, 2008] pour urbaniser l'architecture du lac de données de manière analogique.

Ces études nous ont conduits à utiliser des approches analogiques et mimétiques pour structurer et gérer l'architecture des lacs de données. Afin de construire cette infrastructure raisonnable et imitée, nous définissons d'abord les phases importantes de la conception de l'architecture du lac de données, inspirée du modèle **MERISE**. Le modèle MERISE est une méthode européenne de conception et de développement de systèmes d'information qui repose sur trois niveaux ***Modélisation conceptuel***, ***Modélisation logique*** et ***Modélisation physique*** [Baptiste, 2009]. Selon cette méthode, chaque phase de conception du système d'information des entreprises modifie un enjeu pertinent dans la construction du système. Étant donné que le lac de données est une partie importante des systèmes d'information, les méthodes de conception de systèmes d'information peuvent être appliquées pour concevoir une architecture de lac de données.

À l'instar des approches analogiques, nous retrouvons la méthode MERISE comme déclencheur pour modéliser l'architecture mimétique du lac de données de manière chronologique. Pour cette raison, nous allons promouvoir les niveaux de la méthode MERISE en quatre phases de

conception et de modélisation de la structure systématique du lac de données, comme détaillé ci-dessous.

- ***Phase de modélisation conceptuelle***

  L'objectif de la modélisation conceptuelle est de représenter la structure initiale et les relations entre les différentes entités des systèmes à l'aide de représentations simples afin de préparer un interlocuteur commun de l'architecture principale entre les partenaires techniques et non techniques. Dans cette phase, il est nécessaire de clarifier l'objectif clé de la conception d'une architecture de lac de données en répondant à la question ***"Quoi"*** [Avison, 1991].

- ***Phase de modélisation logique***

  La modélisation logique adresse les raisonnements logiques de mise en pratique de l'architecture sur la base de la phase conceptuelle. Cette phase explique ***"comment"*** les entités d'architecture associées et ***"comment"*** les processus du système fonctionnent.

- ***Phase de modélisation technique (physique)***

  Cette phase porte sur la mise en œuvre technique de l'architecture définie dans les phases précédentes à l'aide de technologies et de logiciels appropriés. Cette modélisation démontre qu'***"avec quels"*** dispositifs et composantss on peut lancer l'architecte applicative, conviviale, rentable, sécurisée et intégrée.

- ***Phase de modélisation de développement (optimisation)***

  Enfin et surtout, c'est la phase de modélisation du développement qui est le socle de l'amélioration et du progrès des systèmes, en particulier des systèmes d'information et de leurs sous-ensembles dans l'environnement technologique varié. Cette phase aborde des stratégies et des méthodes pour concevoir, agencement des éléments importants et modéliser l'architecture du lac de données afin que les performances du système soient continuellement optimisées. L'optimisation de la méthode pourrait être liée au type et à la agencement des composants de la structure (d'un point de vue financier) ou à la fonctionnalité de chaque niveau du système (d'un point de vue efficacité)

En considérant le fait de concevoir l'architecture de lac de données comme un processus cohérent basé sur la méthode MERISE, on atteint un modèle innovant de structuration du lac de données. Les études d'architecture de lac de données se concentrent sur une ou deux phases définies ci-dessus pour concevoir ou fournir un pipeline de lac de données efficace. En revanche, même dans les recherches qui envisagent plusieurs phases, les phases de développement ou d'optimisation sont moins mises en avant. Dans cette étude, nous nous concentrons sur toutes les phases de la modélisation du lac de données depuis l'idée conceptuelle, l'arrangement logique (basé sur l'analogie avec le système logistique), la proposition technique, jusqu'à la modélisation d'un état optimal.

Dans le chapitre qui suit, nous aborderons :

- L'état de l'art et l'analyse des travaux menés sur l'architecture de lac de données en fonction de 4 phases de conception du lac de données ;

- L'architecture logistique d'un lac de données en nous basant sur les quatre phases proposées de la conception du système de gestion de données.

5.2 État de l'art

Dans le chapitre 2 et la section 2.3, nous avons passé en revue une petite histoire architecturale définie par plusieurs recherches menées dans ce domaine. Cependant, comme nous l'avons indiqué, nombre de ces études ne considèrent pas une approche suffisamment cohérente pour concevoir l'architecture des référentiels centralisés de données hétérogènes alors que l'infrastructure et le pipeline des systèmes informatiques ont des impacts directs sur les performances et l'optimisation temporaire ou permanente de ces systèmes. Au regard de ces exigences, nous allons réviser les points forts et faibles des architectures indiquées dans le chapitre 2 afin de distinguer l'état de l'art de cette étude par rapport aux autres recherches menées.

Pour la première étape, nous nous référons à l'architecture d'Inmon basée sur la typologie des données [Inmon, 2016]. Cette architecture est fondée sur plusieurs bassins (Ponds) qui s'occupent des données en fonction leurs types tels que le bassin de données brutes, le bassin de données analogiques, le bassin de données d'application et le bassin d'archivage.

- ***Points forts***
  - Modélisation conceptuelle du lac de données (Première phase de la conception de lac de données) ;
  - Organisation des données pour éviter un lac de données unilatéral ;
  - Taxonomie des données selon l'architecture des données afin de faciliter leur accessibilité pour les scientifiques ou leur utilisation future.
- ***Points faibles***
  - Manque de phases logiques, physiques et d'optimisation du lac de données ;
  - Absence de l'architecture fonctionnelle ;
  - Risque de perte de données pendant le flux de données au sein des bassins (Ponds) ;
  - Manque de composant pour gouvernance de donnnées.

L'architecture de référence de Zaloni va au-delà de la phase conceptuelle en prenant en charge la définition d'un environnement pour un lac de données qui comprend des zones particulières telles que *Transient landing zone*, *Raw zone*, *Trusted zone*, *Refined zone*, et *Sandbox zone* ainsi que la gouvernance des données est appliquée à toutes les zones. Les zones sont définies en fonction de la maturité et de la pureté des données au cours du cycle de vie dans le lac de données.

- ***Points forts***
  - Modélisation conceptuelle plus détaillée du lac de données (Première phase de la conception de lac de données) ;
  - Taxonomie des données selon degrés de raffinage des données ;
  - Garantir la sécurité des données en considérant le *Trusted zone* ;
- ***Point faible***
  - Absence de mise en œuvre de la perspective fonctionnelle en architecture ;
  - Manque de phases logique, physiques et d'optimisation du lac de données ;
  - Risque de perte de lignage des données et de redondance des données répétitives à cause de plusieurs zones de raffinement [Sawadogo and Darmont, 2021].

Les inconvénients des architectures précédentes dirigent les travaux scientifiques ver la structure plus fiable et pragmatique de lac de données qui s'appelle architecture fonctionnelle. Ces architectures sont construites à des niveaux qui indiquent les fonctions pertinentes liées à chaque phase de cycle le la vie des données. Par exemple, [Mehmood et al., 2019] propose une architecture fonctionnelle composée de couches de collecte de données, d'ingestion de données, de stockage de données, d'exploration et d'analyse de données et de visualisation de données.

- ***Points forts***
  - Modélisation conceptuelle, logique, et physique du lac de données (première, deuxième et troisième phases de la conception de lac de données) ;
  - Prise en compte des dispositions fonctionnelles de l'architecture qui facilite la mise en œuvre du lac de données en terme réel ;
  - Proposition de technologies big data pour lancer chaque couche du pipeline.
- ***Point faible***
  - Manque de phase d'optimisation du lac de données ;
  - Absence la couche de gouvernance de données ;
  - Manque d'affinement et de nettoyage des données pendant le cycle de vie.

Les infrastructures de lac de données se développent véritablement avec l'émergence d'architectures hybrides. Le terme architecture hybride désigne deux types d'architectures. D'une part, l'architecture qui considère les phases de maturité et le raffinement des données ainsi que les aspects fonctionnels du lac de données tels que le travail de [Ravat and Zhao, 2019]. D'autre part, l'architecture qui permet des traitements hybrides (traitement des données en batch et en temps réel) à faible latence comme les architectures Lamda et BRAID [Tomcy, 2017, Giebler. et al., 2018]. L'architecture hybride résout presque les lacunes concernant les architectures à mono-considération et augmente la chance que le lac de données soit une utilisation pratique pour les organisations de Big Data.

D'après les études menées dans ce domaine, les forces et faiblesses de l'architecture dirigée sont les suivantes :

- ***Points forts***
  - Modélisation conceptuelle et logique de lac de données (première et deuxième phases du la conception de lac de données) ;
  - Traitement à faible latence ;
  - Haute flexibilité aux données en temps réel ;
  - La considération de la gouvernance de lac de données ;
  - Haute niveau de fiabilité des résultats obtenus.
- ***Point faible***
  - Manque de phases de technique et d'optimisation du lac de données ;
  - Absence de flux inverse de données (lac de données en boucle fermé).

En outre, [Madera, 2018] considère le lac de données comme une composante complémentaire des systèmes d'information. Dans ce travail, l'approche d'urbanisation des systèmes d'information d'entreprise proposée par [Servigne, 2008] est appliquée pour structurer la nouvelle architecture de lac de données. Cependant, dans cette étude, le positionnement du lac de données dans les systèmes d'information est davantage discuté que les conceptions détaillées du pipeline du lac de données.

- ***Points forts***
  - Modélisation conceptuelle du lac de données (première phases de la conception de lac de données) ;
  - Amélioration la complétude des systèmes d'information avec la définition du lac de données comme un nouveau composant ;
  - Approche innovante et analogique pour la conception d'un lac de données (approche urbanisation) ;
  - Vérification de la gravité des données et de son impact sur l'architecture du lac de données.
- ***Point faible***
  - Manque de phases logique, technique et d'optimisation du lac de données ;
  - Absence de flux inverse de données (lac de données en boucle fermé).

Enfin, une architecture de lac de données intelligente a récemment été proposée et traite d'une infrastructure et d'un cadre de lac de données complet tenant compte de la gestion optimale des métadonnées. Ce travail se concentre sur les phases essentielles à l'urbanisation des lacs de données, ce qui constitue un avantage ponctuel par rapport aux autres études menées dans ce domaine [Kachaoui and Belangour, 2020].

Phase de conception du lac de données	Architecture étudiée						
	ALLD	Inmon [Inmon, 2016]	Zaloni	Fonctionnelle [Mehmood et al., 2019]	Hybride [Ravat and Zhao, 2019, Tomcy, 2017, Giebler. et al., 2018]	Madera [Madera, 2018]	Intelligente [Kachaoui and Belangour, 2020]
Modélisation conceptuelle	✓	✓	✓	✓	✓	✓	✓
Modélisation logique	✓			✓	✓		✓
Modélisation technique (physique)	✓			✓			
Modélisation de développement (optimisation)	✓						

FIGURE 5.1 – Les phases de conception de l'architecture du lac de données dans les études réalisées

- ***Points forts***
  - Modélisation conceptuelle et logique du lac de données (première et deuxième phases de la conception de lac de données) ;
  - Réservoir hybride composé du lac de données et entrepôts de données ;
  - Haute compatibilité avec de différentes attentes.
- ***Points faibles***
  - Manque de phases technique et d'optimisation du lac de données ;
  - Absence de flux inverse de données (lac de données en boucle fermé).

Ces observations nous permettent d'identifier des écarts importants dans la structure organisationnelle du lac de données notamment par rapport à la phase de développement qui a fourni une conception optimale. Partant de la connaissance de cette étude, nous abordons sur l'architecture logistique du lac de données qui est appelé (ALLD) en utilisant l'analogie réalisée dans le chapitre 4. La démarche d'urbanisation d'ALLD encadre à la fois les quatre phases de l'urbanisation du lac de données afin d'offrir une structure innovante, inspirante et surtout optimisée. Nous essayons de couvrir les inconvénients liés à l'architecture étudiée en prenant en compte une méthode phasée et systémique pour modéliser le lac de données allant du modèle conceptuel au modèle d'optimisation. Le figure 5.1 démontre la comparaison entre ALD et d'autres architectures indiquées par rapport des phases de modélisation.

Dans la section suivante, nous détaillons chaque phase de la conception d'ALLD en envisageant la réalisation d'une infrastructure innovante.

5.3 Architecture logistique de lac de données (ALLD)

L'idée de l'architecture logistique du lac de données proposée (ALLD) est formée par des analogies systémiques du lac de données avec les systèmes de gestion logistique. Le lac de données en tant

qu'institution d'organisation des données est traité comme une structure logistique axée sur la valorisation des données comme étant le produit principal. Les entités du lac de données, telles que les participants à la chaîne d'approvisionnement, sont coordonnées pour obtenir une satisfaction maximale des consommateurs, ce qui exige une qualité de service et des performances élevées. Par conséquent, une fondation comme le lac de données a besoin d'une urbanisation intégrée qui est le socle d'une conception optimale et rentable de l'infrastructure générale. Nous envisageons que ces caractéristiques bénéfiques soient obtenues grâce à des architectures systémiques basées sur des outils de gestion organisationnelle et logistique.

Dans la section qui suit, nous expliquerons en détail le chemin que nous avons emprunté pour proposer ALLD basé sur les phases quaternaires de la modélisation de l'architecture des lacs de données

5.3.1 Phase de modélisation conceptuelle

La phase de modélisation conceptuelle devrait répondre à cette question "qu'est-ce que l'architecture ou la structure du lac de données" d'une manière simple et compréhensible. En ce qui concerne la structure logistique, la phase de modélisation conceptuelle doit clarifier l'objectif, les principes, les règles et les processus importants de mise en œuvre du système. Par exemple, dans la chaîne d'approvisionnement pour mener à bien la phase conceptuelle, on met en évidence les types de produits ou les services, les réglementations et les normes générales de fabrication et de contrôle des marchandises, les différentes directions des produits et les informations au sein du système, les considérations importantes à maintenir ou à détruire des produits, et etc. De la même manière pour structurer ALLD conceptuel, les approches et les règles générales de valorisation des données au sein d'un lac de données doivent être prises en compte.

Selon les aspects du DALF, qui est un framework pour lancer l'architecture du lac de données, une structure de lac de données exhaustive doit prendre en charge des fonctionnalités essentielles telles que l'acquisition, le stockage, le traitement, l'analyse et la visualisation des données, ainsi que garantir l'accessibilité, la traçabilité et la sécurité des données [Giebler et al., 2021]. DALF nous propose 9 caractéristiques à prendre en compte lors de la conception du lac de données dans lesquelles 7 caractéristiques (traitement des données, organisation des données, modélisation des données, flux de données, stockage des données, infrastructures et gestion des métadonnées) sont appliquées conceptuellement et physiquement et 2 caractéristiques (sécurité et confidentialité des données et qualité des données) sont simplement appliquées conceptuellement à l'architecture du lac de données. La figure 5.2 représente les aspects importants pour réaliser une architecture complète de lac de données.

On définit l'architecture conceptuelle de lac de données sur la base des aspects du DALF qui devrait encadrer les caractéristiques reprises ci-dessous.

- ***L'infrastructure générale :*** L'environnement ou la plate-forme principale qui prend en charge toutes les autres dispositions liées à la gestion et à l'amélioration des données qui pourraient être choisies en fonction des objectifs ou des possibilités de la mise en œuvre en tant que lac de données on-permise, lac de données cloud, lac de données hybride (cloud et on-permise) et lac de données multi-cloud.

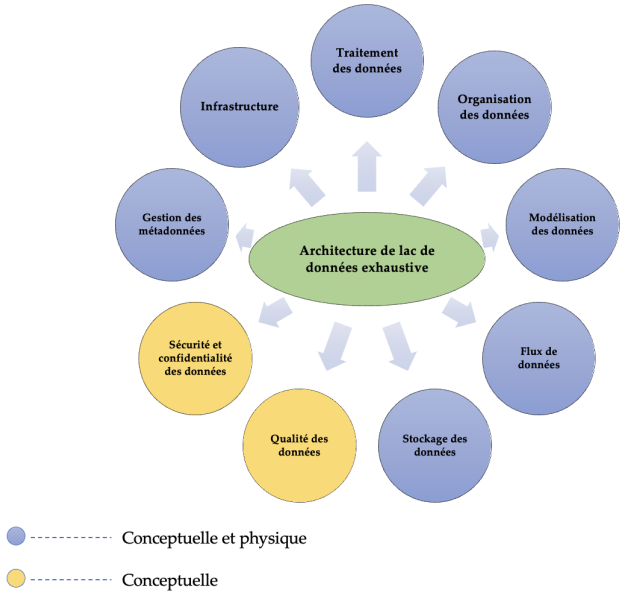


FIGURE 5.2 – Aspects de cadre DALF pour la conception d'architecture de lac de données [Giebler et al., 2021]

- ***L'acquisition de données :*** Les dispositions d'ingestion de données qui sont définies selon les types de sources et de formats de données, la compatibilité API, le flux de données (en lot ou en temps réel), et la planification d'utilisation et de mise à jour des données (ingestion manuelle ou automatique) [Mehmood et al., 2019].

- ***L'accessibilité, la traçabilité (la gestion des métadonnées) :*** Le système efficace de gestion des métadonnées, de catalogues de données et d'indexation, facilite et améliore le processus de requête d'une part, et réduit le risque de marécage de données d'autre part. Le système de métadonnées joue un rôle crucial dans l'accessibilité et la traçabilité des différents types de données au sein du data lake. A ce titre, la mise en place de dispositions pratiques dans le corpus du lac de données qui propose les techniques d'extraction et de stockage des métadonnées, est indispensable.

  La typologie de l'architecture du lac de données basée sur l'organisation des métadonnées est distinguée entre l'architecture qui gère les métadonnées comme un composant global et l'architecture qui organise les métadonnées selon les différentes structures de données [Sawadogo et al., 2019b, Haslhofer and Klas, 2010]. D'une manière générale, l'un des enjeux importants dans la conception de l'architecture du lac de données est de définir le système de métadonnées et d'indexation compatible, les techniques pour extraire facilement les métadonnées et les formats pour les stocker de manière à rendre les données plus trouvables, réutilisable et accessible (Les données FAIR). La disposition des métadonnées pourrait être omniprésente dans toutes les couches de l'architecture du lac de données. Par exemple, dans la couche d'ingestion, les métadonnées sont chargées avec des données, dans la couche de stockage, les métadonnées sont stockées avec les données au format prédéfini, dans les couches de traitement, les métadonnées sont mises à jour après le traitement des données et dans la couche d'accès, les métadonnées jouent un rôle vital dans l'interogration des données [Eichler et al., 2021].

- ***Le stokage de données :*** Les décisions concernant la structure de la couche de stockage de données pourraient être considérées comme l'agencement du stockage de données tel qu'un système de fichiers distribué unique ou un système de stockage multiple ainsi que les formats choisis pour le stockage de données tels que JSON, CSV, Apache Iceberg [1] ou Delta Lake [2] [Zikopoulos, 2015, Giebler et al., 2021, Belov and Nikulchev, 2021].

- ***La traitement de donnée :*** En fonction des objectifs définis pour le lac de données et des attentes avérées, la couche de traitement des données pourra être conçue pour traiter des données en lots comme les technologies du MapReduce dans un environnement hadoop, traiter des données temps réel comme Spark ou une combinaison de deux technologies en architecture hybride comme l'architecture du Lambda [Tomcy, 2017].

- ***L'organisation de données :*** L'organisation des données aborde la classification des données selon leurs schémas, leurs maturités (brutes, traitées ou utilisées), et leur sensibilité et confidentialité. Les dispositions de partitionnement des données dans l'architecture du lac améliorent l'orchestration et l'harmonisation des données et assurent la préservation de données propres [3] [Inmon, 2016, Nargesian et al., 2019].

- ***La modélisation de données :*** La modélisation du schéma de données est un élément important pour une architecture de lac de données organisée et consultable qui indique avec précision les données sont trouvées comment, où et avec quelles relations sémantiques. Les modèles de données les plus connus dans le domaine de la conception de données sont classés en modèles multidimensionnels tels que les modèles de flocon de neige dans l'entrepôt de données et les modèles d'ensemble tels que *Data vaults* et la modélisation d'ancre [4] [Nogueira et al., 2018].

- ***La gouvernance et la qualités de données :*** Ces deux considérations conceptuelles garantissent que les bonnes données sont accessibles au bon moment à la bonne personne. Les ajustements de gouvernance et les mesures d'évolution de la qualité des données doivent être pris en compte dans la conception du lac de données afin d'augmenter la viabilité et la variabilité des données et de réduire le risque d'être submergé par des données inutiles et sans valeurs. Des dispositions de gestion de la qualité des données et de gouvernance de la sécurité doivent être en place tout au long du cycle de vie des données dans l'infrastructure du lac de données, pour surveiller la propreté des données et détecter les cas anormaux ou inappropriés.

Sur la base des considérations initiales de la conceptions du lac de données, nous modélisons l'architecture conceptuelle du lac de données comme illustré par la figure 5.3.

D'après la figure 5.3, nous définissons l'architecture conceptuelle comme étant basée sur les quatre fonctions du lac de données définies par [LaPlante, 2016]. Ces fonctions sont l'ingestion, le stockage, le traitement et l'accès des données. Le flux direct de données commence à partir des différentes sources de génération de données, passe par les quatre phases de valorisation des

[1] https://iceberg.apache.org
[2] https://docs.delta.io/latest/index.html
[3] Data garbage
[4] Anchor modeling

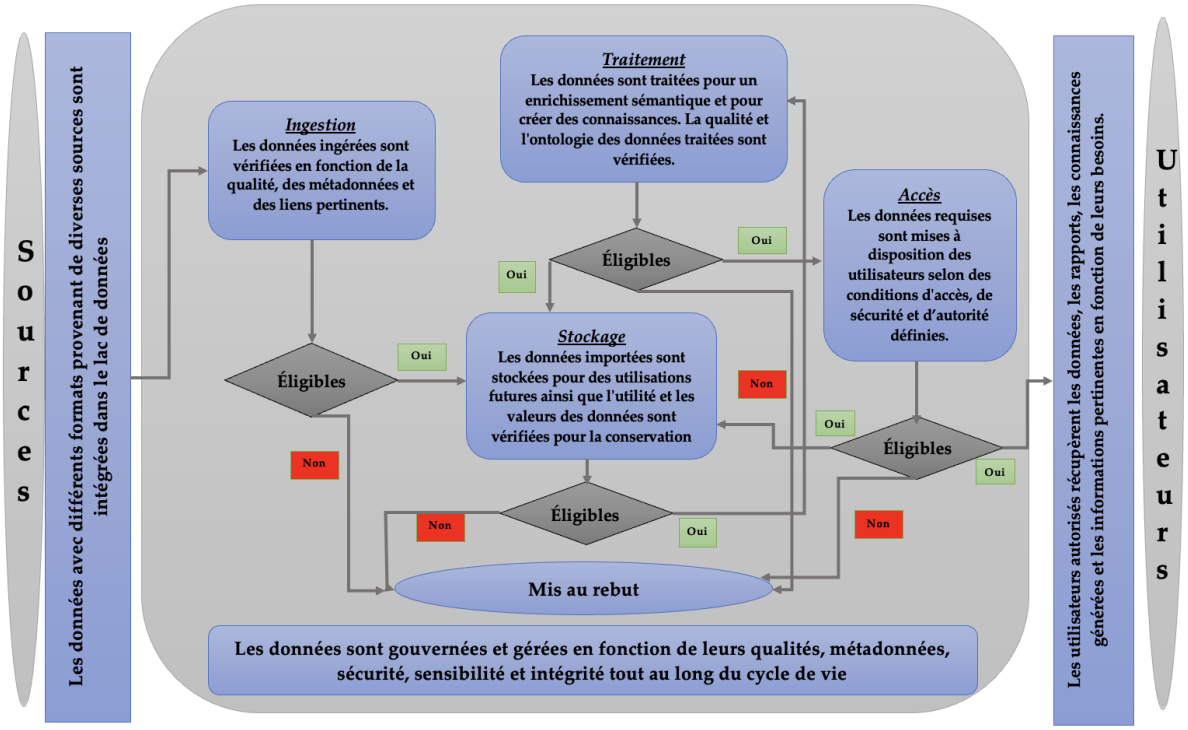


FIGURE 5.3 – Architecture conceptuelle du lac de données

données au sein du lac de données et se termine par la satisfaction des besoins des utilisateurs. Pour effectuer chaque fonction sur les données, les processus de contrôle d'éligibilité des données sont mis en œuvre afin d'autoriser la circulation des données pour l'action suivante. Le processus d'éligibilité contient toutes les stratégies, réglementations et protocoles de gouvernance et de gestion des données pour vérifier la qualité, la véracité, le traçage et la sécurité des données. Les données éligibles sont stockées, traitées et rendues accessibles aux utilisateurs, ainsi que restockées sous forme de données raffinées dans un flux inverse, par exemple, pour les fonctions de traitement et l'accès à un flux inverse de données éligibles vers la fonction de stockage. En revanche, les données non autorisées sont rejetées, mises en rebut ou supprimées de sorte que la qualité des données est garantie et le risque de marécage de données réduit.

Dans cette architecture, toutes les métriques essentielles de DALF pour concevoir une structure de lac de données efficace sont prises en compte. Cependant, la modélisation conceptuelle ne permet que de présenter la fonctionnalité générale de l'architecture et de répondre à cette question que ***Qu'est-ce que l'architecture de lac de données*** . Pour compléter l'architecture défini, il faut également répondre à ces question ***Comment fonctionne l'architecture du lac de données?*** Ou ***Comment l'architecture conceptuelle du lac de données sera réalisée ?***. Afin de répondre à ces questions, on va aborder la phase de modélisation logique dans la section suivante.

5.3.2 Phase de modélisation logique

Dans la phase de modélisation logique, nous essayons de détailler et de réaliser l'architecture conceptuelle à l'aide d'une méthode mimétique pour structurer et urbaniser l'environnement du lac de données de manière plus logique. Évidemment, il est nécessaire de préciser la manière dont chaque élément de la structure conceptuelle pourrait être réalisé. En se référant à l'analogie avec le système logistique, pour réaliser la phase de modélisation logique dans ce système, il est nécessaire de prendre en compte le fait que nous avons besoin de quel et combien de niveaux de membres et d'échelons et avec quelles fonctions pour mettre en œuvre toutes les considérations définies dans la phase de modélisation conceptuelle. Dans la chaîne d'approvisionnement, le nombre de niveaux essentiels de la chaîne, les responsabilités de chaque membre, les stratégies managériales pour les niveaux amont et aval, et l'organisation centralisée à décentralisée (verticale ou horizontale) du système sont vérifiés.

Comment définir une architecture garantissant l'intégration verticale des différentes phases de valorisation des données et surveiller la qualité et la sécurité des données tout au long de leur parcours dans les systèmes de gestion des données. Pour repondre à cettes questions, nous revenons sur l'analogie du système logistique avec lac de données. Dans cette analogie que nous avons détaillée au chapitre 4, et en se référant au tableau de comparaison de 4.1, nous avons vu que les points similaires entre le système logisque de chaîne d'approvisionnement et le lac de données nous permettent de tirer parti des méthodes de conception de réseau de chaîne d'approvisionnement et des stratégies de gestion logistique pour concevoir et gérer l'environnement du lac de données. Sur la base de ces résultats et de la littérature sur les architectures de lac de données en couches, nous proposons l'architecture logique du lac de données sur la base de logique d'analogies réalisées entre deux systèmes. On remarque les hypothèses de cette architecture sous forme de suites :

Hypothèse de conception logique :

- Les principes logiques de l'architecture du lac de données reposent sur le réseau d'une chaîne d'approvisionnement en boucle fermée à 4 niveaux (fournisseurs, fabricant, distributeur et détaillants).

- Le produit principal de lac de données est présenté dans différents formats de données, tels que des données structurées, non-structurées et semi-structurées provenant de ressources dispersées.

- La structure du lac de données se concentre sur $ALLD = \{N, F, D\}$, où N est le nombre de nœuds ou chouches de lac de données qui sont 4 niveaux de valorisation des données comme l'ingestion, le stockage, le traitement et l'accès, F qui indique à la fois deux flux de données avant et arrière, et D qui correspond aux trois typologies de formats de données, structurés, non structurés et semi-structurés.

- Il existe deux niveaux de gestion des entités du lac de données qui sont les niveaux locaux (gestion locale) basés sur la structure de décision décentralisée de la chaîne d'approvisionnement et le niveau global (gestion globale) basé sur la structure de décision centralisée de la chaîne d'approvisionnement.

La figure 5.4 montre la structure d'ALLD, une architecture de lac de données basée sur le raisonnement des systèmes logistiques. Cette architecture montre comment l'architecture conceptuelle pourrait être mise en œuvre. Les données dans cette architecture sont gérées par deux niveaux d'administration des données, un ***niveau global*** de gouvernance et de gestion des données qui supervise et surveille les données tout au long de leur parcours et les ***niveaux locaux*** qui sont subordonnés au niveau global et sont divisés dans chaque couche afin de gérer les données selon la phase particulière du cycle de vie.

Le niveau global s'inspire d'une structure décisionnelle centralisée de la chaîne d'approvisionnement où l'autorité et l'authenticité de la prise de décision et la détermination des réglementations sont attribuées à une entité ou à un niveau [Giannoccaro, 2018]. Dans ALLD, la zone de la gouvernance et de la gestion des données est le dirigeant du système qui définit des protocoles et des règles généraux et complets pour vérifier et surveiller la sécurité, la lignée, l'intégrité, la traçabilité, la trouvabilité, l'interprétabilité, la qualité, la protection et l'orchestration des données pendant leur présence dans le système et garantir l'intégration horizontale des entités du système. En outre, les ***niveaux locaux*** gèrent les données en tant que structure décisionnelle décentralisée des systèmes logistiques où chaque couche est un dirigeant distinct tels que gestionnaire d'ingestion, de stokage, de traitemet et d'accès qui vérifie la mise en œuvre du protocole défini général sous la direction du niveau global [Lee and Billington, 1993]. La différence entre la structure décentralisée d'ALLD et la chaîne d'approvisionnement est que les gestionnaires locaux sont interdépendants et ne sont pas totalement isolés ainsi que sont responsables de l'intégration horizontale dans sa couche sous l'autorité du gestionnaire global. Par conséquent, dans chaque couche, l'éligibilité des données à circuler vers d'autres couches doit être autorisée par le gestionnaire de la couche concernée. **Logiquement, la gouvernance et la gestion globales sont comparables à l'évaluation du cycle de vie des produits dans les systèmes logistiques.**

La logistique des données au sein d'ALLD est déroulée comme suit selon les phases principles :

- **Phase 1 :** Le parcours des données dans ALLD est commencé par un flux vers l'avant à partir de différentes sources de génération de données telles que des capteurs, des réseaux sociaux, des entrepôts de données, des Logs, des données transactionnelles, des services Web ou cloud dans des formats divers.
- **Phase 2 :** Dès que les données entrent dans le lac de données, elles sont temporairement embarquées dans la zone d'ingestion ; à l'état brut et d'origine et sans aucune transformation. Dans la couche d'ingestion, l'extraction des métadonnées, l'indexation et le tagging sont effectués conformément au modèle de métadonnées et au catalogue de données, des liens logiques sont générés et l'éligibilité des données est évaluée par le niveau local de gestion d'ingestion [Zhao et al., 2021, Sawadogo et al., 2019b]. Les données éligibles avec les métadonnées pertinentes sont acheminées vers la couche de stockage pour être conservées dans les formats appropriés. Cette phase est considérée comme le niveau d'approvisionnement et de préparation matérielle du produit dans la chaîne d'approvisionnement dans laquelle les sources et la qualité des matériaux sont vérifiées avant d'effectuer d'autres procédures.
- **Phase 3 :** Les données éligibles et bien contrôlées sont transmises à la couche de stockage pour être conservées et stockées pour une utilisation future en fonction des besoins particuliers des utilisateurs. Le stratégie pour concevoir le couche de stockage pourrait être axé

sur deux pôles : stockage en bassin (Pond) unique ou stockage sur multiples bassins. La structure de cette couche est basée sur l'architecture d'Inmon dans laquelle les données sont stockées en plusieurs bassines (Ponds) selon leurs formats [Inmon, 2016]. En revanche, dans ALLD, dans la couche de stockage, les données sont stockées selon le degré de maturité dans trois bassins différentes, les données brutes qui proviennent de la couche d'ingestion, les données archivées qui proviennent de la couche de traitement et les données d'occasion qui vient de la couche d'accès ou des utilisateurs. De plus, la couche de stockage est la seule couche qui accueille à la fois les flux de données aller-retour en tant que système logistique en boucle fermée. Dans la couche de stockage, les métadonnées sont stockées dans des formats pertinents pour faciliter la recherche et la découverte de données pour les utilisateurs. Ainsi, l'arrangement de mapping de données qui garantit l'intégration, l'immigration et la transformation des données au format de destination, sont mis en œuvre dans cette couche. Par conséquent, toutes les procédures de couche d'ingestion sont effectuées sous la supervision des principes de la gestion du stockage afin de qualifier les valeurs de données stockées pour les requêtes ciblées et de réduire le risque de marécage de données. Cette phase est considérée comme le niveau de stockage des produits dans la chaîne d'approvisionnement dans lequel les produits finaux, semi-finis ou produits retournés par le flux inverse, sont stockés pour d'éventuels traitements.

- **Phase 4 :** Toutes les transformations, enrichissement sémantique et affinement des données par lots ou en temps réel sont implémentés dans la couche de traitement selon les exigences requises. Étant donné que les données sont importées telles quelles dans le lac de données et qu'aucune transformation n'est autorisée dans les autres couches susmentionnées, cette zone est la couche décisive qui pourrait avoir un impact important sur la génération de connaissances valides et la rentabilité du lac de données. Pour cette raison, tous les outils et dispositions d'analyse de données tels que l'intelligence artificielle et l'apprentissage automatique, les algorithmes d'exploration, les tâches d'enrichissement sémantique et les opérations de valorisation des données sont mobilisés dans cette couche afin de traiter les données de manière évolutionniste [Joaquim and dos Santos Mello, 2020, Mehmood et al., 2019]. De plus, cette couche est encadrée par la technologie pour la gestion et la planification des calculs et l'exécution des tâches concernées. Tous les processus de récupération et de transformation des données sont effectués sous le contrôle du gestionnaire du traitement afin de garantir l'intégrité, l'accessibilité, la réutilisabilité et l'interprobabilité des données traitées. Les données ont deux destinations dans la couche de traitement, un flux vers l'avant vers la couche d'accès et un flux vers derrière vers la couche de stockage. Cette phase est comparée au niveau de fabrication du produit dans la chaîne d'approvisionnement dans laquelle les produits finaux, fabriqués selon les demandes soulevées des clients.

- **Phase 5 :** Les données traitées sont prêtes à être mises à disposition des utilisateurs finaux. Les demandes spécifiques des utilisateurs sont effectuées via des requêtes par la couche d'interaction. Cette couche prépare une plate-forme Self-service de consommation de données pour faciliter l'exploitation et l'exploitation des données avec un haut niveau de gestion des métadonnées, des services API compatibles et des consoles d'interrogation pour les requêtes SQL et NoSQL. Toutes les visualisations, rapports analytiques et informations statistiques qui amélioreront le processus de prise de décision pour les organisations (utilisateurs particuliers) sont générés et disponibles dans cette couche. L'administration de la couche d'accès contient toutes les stratégies et tous les principes pour gérer les niveaux d'accessibilité des utilisateurs, augmenter la commodité d'interroger les informations re-

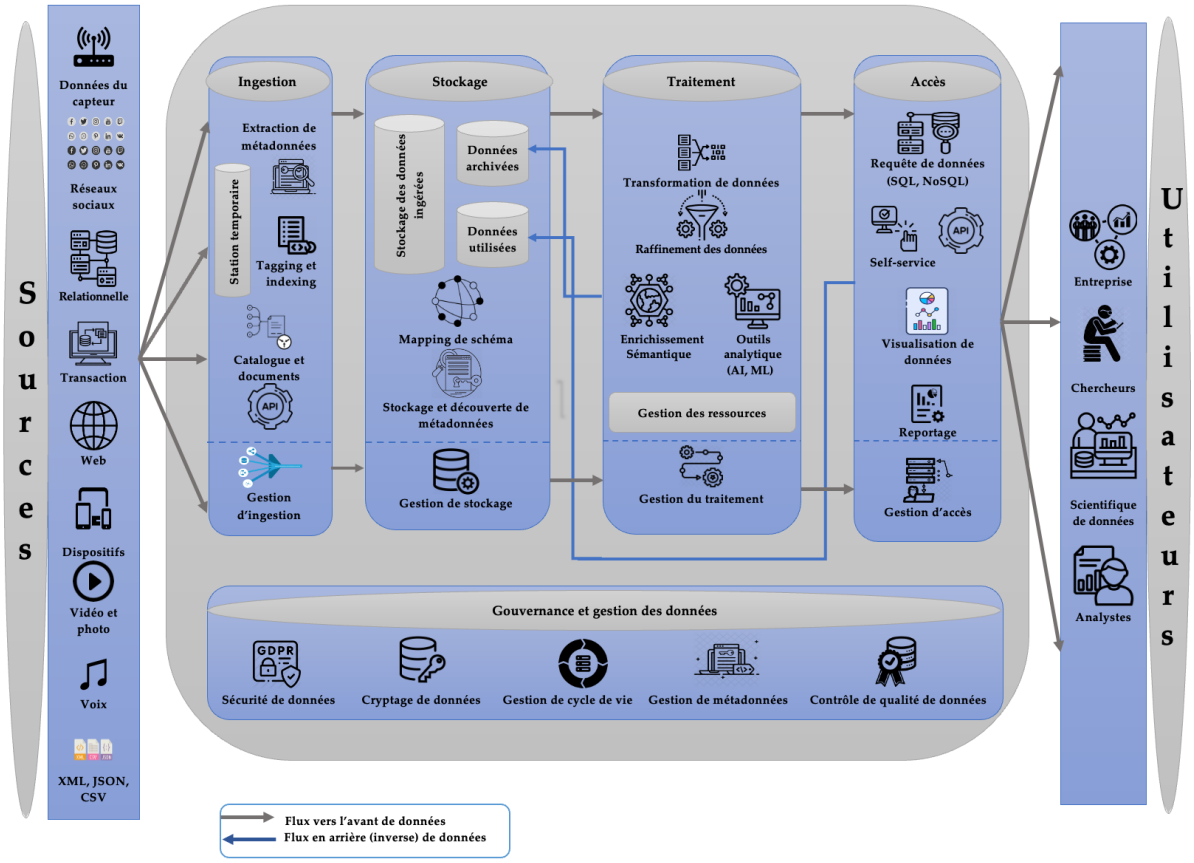


FIGURE 5.4 – Architecture logique du lac de données

quises, gérer les métadonnées pour faciliter la récupération des données et protéger les données sensibles. Les données ont deux destinations dans la couche d'accès, un flux vers l'avant vers l'utilisateur d'accès et un flux vers l'arrière vers la couche de stockage. Cette phase est comparée au niveau de détaillant du produit dans la chaîne d'approvisionnement dans laquelle les produits finaux sont directement disponibles pour la consommation.

L'architecture logique susmentionnée répond aux questions ***Comment fonctionne l'architecture du lac de données ?*** Ou ***Comment l'architecture conceptuelle du lac de données sera réalisée ?***, en utilisant une analogie avec la chaîne d'approvisionnement et les littératures existantes. Cependant, la mise en œuvre de l'architecture logique reste encore un sujet problématique pour les organisations qui met en évidence l'exigence d'architecture technique et applicative. Pour cette raison, dans la section suivante, nous envisageons également de répondre à cette question que ***Comment l'architecture logique du lac de données sera-t-elle réalisée en utilisant les technologies appropriées ?***

5.3.3 Phase de modélisation technique (physique)

La troisième phase de l'urbanisation du lac de données est la phase de modélisation technique et applicative. Cette phase adresse le déploiement de l'architecture logique à l'aide des technologies et services actuels. L'architecture physique est une structure informatique du lac de données qui manifeste chaque élément de ce système fonctionnant avec quelle disposition et application technique ainsi que la détermination des structures de données [Madera, 2018]. Par rapport à un système logistique, pour réaliser cette phase il faut considérer avec quelles dispositions et technologies (machines de fabrication, types de transport, types d'entrepôts,...) la phase logique pourrait être effectuée.

Comme nous l'avons indiqué au chapitre 2, l'implémentation de l'architecture applicative du lac de données est un enjeu décisif pour les organisations dans l'environnement concurrentiel des fournisseurs de services de big data tels que Amazon, IBM, Oracle, Azure, Google,... . Malgré les énormes marchés de logiciels, de technologies, de plates-formes et d'infrastructures offerts par les principales coopérations technologiques et fournisseurs de services, ils ne existent pas de la meilleure sélection d'architecture technique unique pour toutes les organisations. Le choix de la meilleure architecture (pipeline) technique et des technologies associées est un processus de prise de décision multicritère qui dépend des besoins des entreprises, de leurs objectifs, de leurs attentes de valorisation des données, de leurs budgets, de leurs équipements, et de leurs capacités techniques.

Dans cette étude, nous proposons une architecture technique selon deux facteurs importants :

- **Facteur 1 :** Architecture logistique du lac de données (ALLD)
- **Facteur 2 :** Objectifs d'optimisation du lac de données

En ce qui concerne le **Facteur 1**, nous envisageons que chaque couche ou niveau du lac de données est mis en œuvre avec les technologies appropriées qui prennent en charge le flux logistique de données dans le réseau et sont compatibles avec les concepts logistiques. D'autre part, pour tenir en compte le **Facteur 2** et en fonction de l'objectif défini pour l'optimisation du lac de données, nous avons choisi les technologies qui se comparent aux approches et aux modèles mathématiques d'optimisation du lac de données selon des méthodes choisies. Le figure 5.5 montre la modélisation technique proposée pour ALLD on-premise dans écosystème Hadoop.

Compte tenu de la figure 5.5, chaque couche est chargée par les technologies concernées de ses fonctionnalités comme décrit ci-après.

Couche d'ingestion

- **Apache Tika** [5] **:** Il s'agit d'une librairie permettant d'extraire des métadonnées de plusieurs formats pour détecter les types de documents et analyser les contenus.

[5] https://tika.apache.org

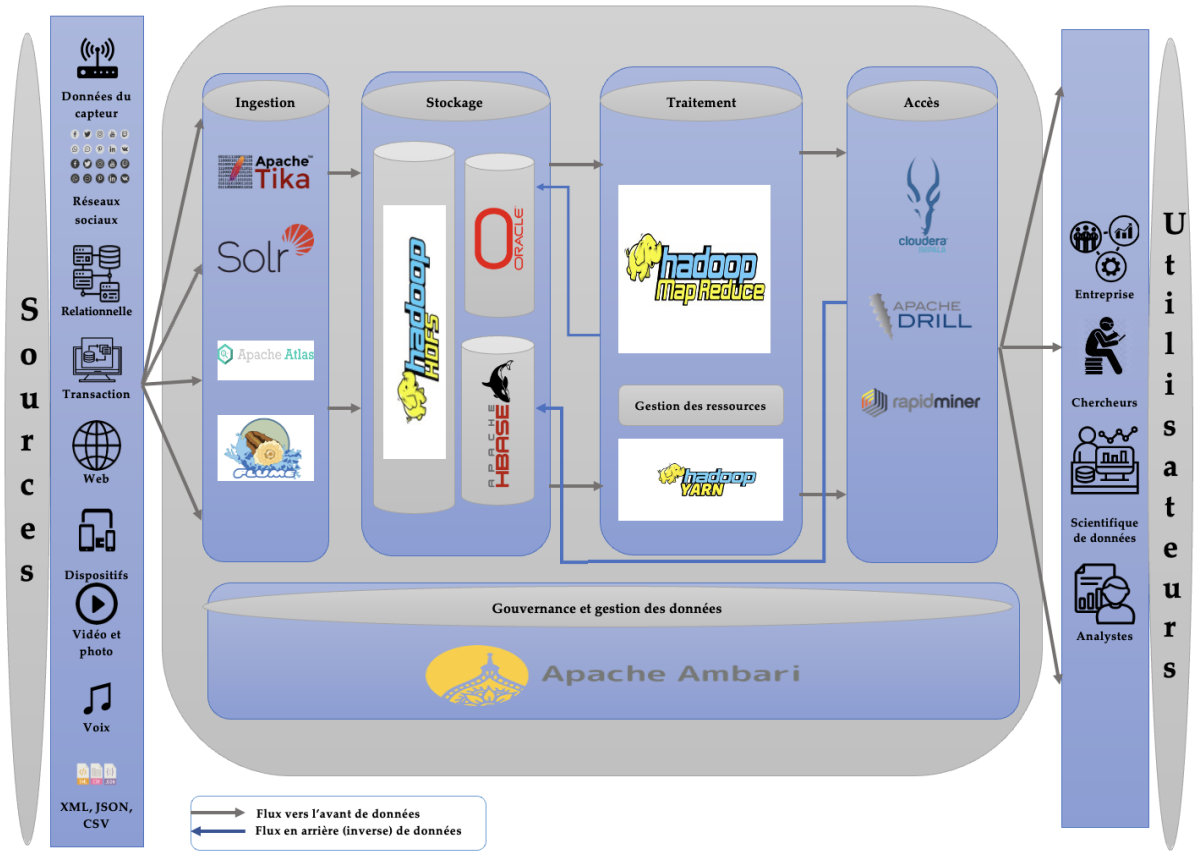


FIGURE 5.5 – Architecture technique du lac de données

- **Apache Solr** [6] **:** Il s'agit d'une plateforme open-source pour indexer et rechercher en texte intégral depuis des sources variées.
- **Apache Atlas** [7] **:** Il s'agit d'un service de gestion des métadonnées qui facilite l'organisation du catalogue, la traçabilité, la classification et la sécurité des données.
- **Apache Flume** [8] **:** On utilise ce service distribué pour collecter, agréger des données non structurées et les importer dans l'écosystème Hadoop qu'on utilise pour ingestion les données

Couche de stockage

- **HDFS** [9] **:** Cette technologie est définie comme un système de fichiers distribué avec une tolérance aux pannes élevée.Pour cette raison, elle pourrait être efficace pour héberger les données brutes qui sont ingérées directement à partir de la couche d'ingestion.

[6] https://solr.apache.org
[7] https://atlas.apache.org
[8] https://flume.apache.org
[9] https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

- **Oracle Database** [10] **:** Oracle Database est un système de gestion de base de données relationnelle (SGBDR) flexible et économique qui stocke et gère les données de manière conviviale. Cette base de données est proposée dans le cadre de la couche de stockage de données pour charger les données traitées comme un ensemble qui sont exportées de la couche de traitement.
- **Hbase** [11] **:** Il s'agit d'un système de gestion de base de données NoSQL orienté colonnes.Pour cette raison, nous définissons une autre partie dans la couche de stockage afin d'héberger les données d'occasions en retour provenant de la couche d'accès.

Couche de traitement

- **MapReduce** [12] **:** Compte tenu des caractéristiques de MapReduce que nous avons indiquées, on suppose dans cette étude que Hadoop MapReduce pourrait être utilisé comme une technologie de traitement de données à grande échelle en exécutant des Jobs sur les clusters de serveurs dans ALLD.
- **Apache Hadoop YARN** [13] **:** Il s'agit d'un technologie complémentaire pour MapReduce qui est responsable de gestion de l'allocation des ressources du système et les fonctions de Mapreduce ainsi que planifications de Jobs.

Couche d'accès

- **Apache Impala** [14] **:** Il s'agit d'un moteur de requête SQL à traitement parallèle pour l'écosystème Apache Hadoop qui est compatible avec nombreuse des outils analytiques et prépare une interaction avec utilisateur à faible latence pour effectuer des requêtes et recevoir des informations pertinentes
- **Apache Drill** [15] **:** Il s'agit d'un moteur de requête NoSQL compatible avec diverses bases de données et systèmes de fichiers tels que HDFS et HBbase et analyse les données non structurées avec un minimum d'effort et avec un niveau élevé de flexibilité et d'agilité.
- **Rapidminer** [16] **:** Il s'agit d'un technologie open-source pour l'exploration, l'exploitation et la visualisation des données.

Couche de gouvernance et gestion de données

- **Ambari** [17] **:** Il s'agit d'un logiciel de projet de *Apache Software Foundation* qui a la capacité de gérer et de superviser les données au sein de l'écosystème Hadoop. Ambari fournit toutes les fonctionnalités pour sécuriser, configurer, gérer et gouverner les données de manière conviviale.

[10] https://www.oracle.com/database/
[11] https://hbase.apache.org
[12] https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html
[13] https://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/YARN.html
[14] https://impala.apache.org
[15] https://drill.apache.org
[16] https://rapidminer.com
[17] https://ambari.apache.org

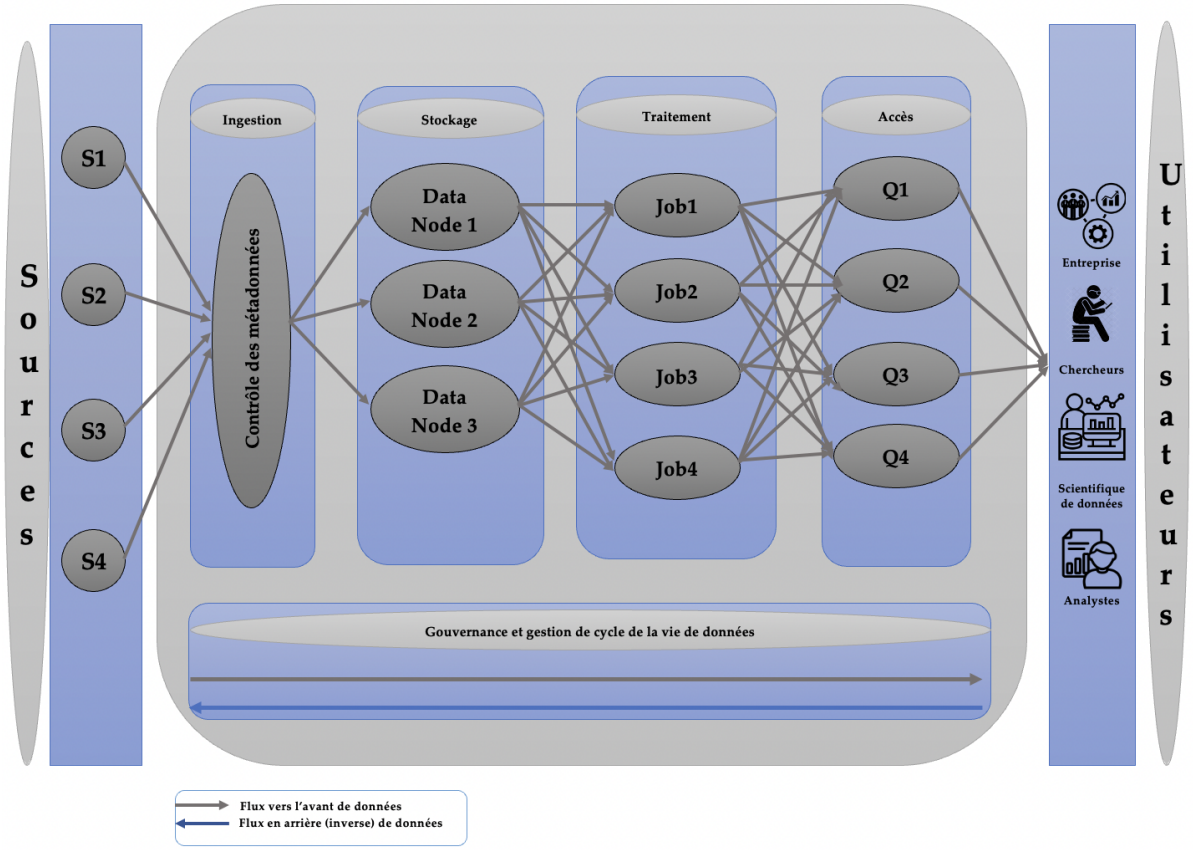


FIGURE 5.6 – Architecture d'optimisation du lac de données

5.3.4 Phase de modélisation optimale

La dernière phase de modélisation, qui est moins abordée dans la littérature sur l'urbanisation des lacs de données, est la phase d'optimisation du pipeline des lacs de données. Le développement continu et l'optimisation pratique sont les socles importantes dans l'amélioration des structures systémiques que ce soit les systèmes logistiques ou le système informatique. Étant donné que ALLD est censé s'appuyer sur la logique des systèmes logistiques, nous affirmons que l'une des stratégies efficaces pour concevoir le réseau de chaîne d'approvisionnement optimisé est le problème conjoint de localisation-allocation. Dans la phase de modélisation optimale basée sur cette stratégie, il est nécessaire de déterminer combien de niveaux ou de dispositions (fabricants, entrepôts, transport, ...) doivent être nécessaires afin d'augmenter la qualité des services tout en minimisant les coûts totaux. De même pour une architecture optimale de lac de données, il convient de vérifier comment la fonctionnalité du système serait développée en configurant efficacement les paramètres essentiels des technologies choisies dans chaque couche, par exemple (le nombre de nœuds ou le nombre de jobs actifs).

Pour cette raison, nous étendons les phases de conception de l'architecture du lac de données en 4 phases où la phase de modélisation optimale complète les trois autres phases mentionnées. L'optimisation de l'architecture du lac de données pourrait adresser à deux pôles généraux :

- **Optimisation financière** qui contient toutes les décisions concernant les questions financières telles que l'investissement technique ou physique pour la mise en œuvre du lac de données. Par exemple, minimiser les coûts de préparation des plates-formes, des technologies, et des logiciels pertinents tout en maximisant la rentabilité du système pour les organisations.
- **Optimisation fonctionnelle** qui traite toutes les décisions concernant le fonctionnement global du lac de données en choisissant des méthodes efficaces de gestion et de traitement des données pour optimiser les performances du système. Par exemple, minimiser le temps d'exécution du lac de données en maximisant le taux de réponse des utilisateurs.

Dans cette étude nous décidons d'appuyer sur le pôle d'optimisation fonctionnelle plutôt que le pôle financier afin de proposer des approches innovantes pour optimiser la fonction du lac de données sous sa structure logistique. Par conséquent, selon le thème analogique de ce travail, nous utilisons des stratégies d'imitation inspirées de la chaîne d'approvisionnement pour atteindre cet objectif. Sur la base de cette méthode analogique avec les systèmes logistiques et aussi les technologies choisies pour les couches indiquées, l'architecture d'optimisation du lac de données est réalisée sous la forme de la figure 5.6 où les données acheminent un parcours logistique sur plusieurs voies pour servir les utilisateurs. Étant donné que l'optimisation de l'architecture des lacs de données est l'objectif principal de cette étude, nous expliquerons en détail cette phase d'urbanisation des lacs de données ainsi que l'anatomie de l'architecture dans les chapitres 6 et 7.

5.4 Résumé

Dans ce chapitre, nous avons parlé de l'architecture des systèmes logistiques qui est une décision stratégique et a un grand impact sur la fonctionnalité et la productivité du système. Une conception d'infrastructure bien délibérée et bien étudiée pourrait garantir l'évolutivité, la flexibilité, l'agilité et la rentabilité du système. La vision logistique du lac de données ainsi que la littérature existante sur l'infrastructure des systèmes de gestion des données, nous permettent de concevoir l'architecture appropriée qui est baptisée ALLD et qui est compatible avec le raisonnement logistique et les concepts de chaîne d'approvisionnement. La conception d'ALLD progresse sur une voie de modélisation évolutionniste afin d'obtenir une structure pratique et adaptée aux processus d'optimisation logistique basés sur des stratégies de chaîne d'approvisionnement.

Chapitre 6

Optimisation de l'architecture du lac de données

"Une réponse approximative au bon problème vaut beaucoup plus qu'une réponse exacte à un problème approximatif."

– John Tukey

6.1 Introduction

L'optimisation est l'une des plus vieilles branches des sciences mathématiques. Elle joue un rôle de catalyseur pour le développement de modèles mathématiques, et elle est également devenue d'un grand intérêt dans les domaines multidisciplinaires et dans les activités quotidiennes [Kenneth, 2013]. L'optimisation concerne toutes les stratégies et les techniques mathématiques qui conçoivent et modélisent des mécanismes pour améliorer les fonctionnalités d'un problème en minimisant ou en maximisant les objectifs prédéfinis. Selon [Gill et al., 2019] :

" Un problème d'optimisation générique contient les étapes pour rechercher les valeurs optimales d'une ou plusieurs fonctions objectives qui couvrent tous les domaines scientifiques tels que l'ingénierie, la médecine ou la gestion. "

Compte tenu de l'impact de l'optimisation sur la productivité, l'optimisation a été signalée comme une valeur compétitive et indispensable dans tous les domaines multidisciplinaires que ce soit les organisations, les systèmes logistiques ou les systèmes informatiques. Dans les systèmes logistiques tels que la chaîne d'approvisionnement, l'optimisation est réalisée grâce à la modélisation du réseau et aux modèles mathématiques de recherche opérationnelle afin de minimiser les risques, les coûts et les pertes ou de maximiser les performances, la rentabilité ou le rendement. En général, les modèles d'optimisation dans la chaîne d'approvisionnement s'appuient sur la conception de réseaux pour chercher les valeurs optimales de nombre d'installations, de quantités de commandes, de quantités de stocks de sécurité et de temps de préparation des services. De même, dans le domaine informatique, nous profitons des méthodes d'optimisation pour la modélisation des réseaux d'ordinateurs centralisés et distribués afin de résoudre les problèmes de localisation et de capacité de chaque connexion réseau et de concevoir un réseau à faible coût, abordable et fiable [Frank and Chou, 1972]. De plus, l'optimisation joue un rôle important dans le développement des systèmes d'information ou des systèmes de stockage de données. Dans ce domaine, l'optimisation concerne le développement de processus de requêtes ou la conception d'une architecture optimisée tout en minimisant les coûts d'investissement et en favorisant les niveaux d'évaluation des requêtes de données [Panahi and Navimipour, 2019].

Malgré les nombreuses études qui sont menées dans le développement de la fonctionnalité des systèmes informatiques, en particulier dans les systèmes de stockage de données, les thèmes de l'optimisation de l'infrastructure de ces systèmes avec des stratégies managériaux sont moins abordés. Étant donnée l'analogie entre les systèmes logistiques du chapitre 4 et l'utilité des outils de la gestion de la chaîne d'approvisionnement pour faire avancer le fonctionnement des systèmes de stockage de données, nous décidons de tirer parti des méthodes multidisciplinaires pour optimiser l'architecture du lac de données. Pour cette raison, nous nous appuyons sur les modèles de conception de réseaux de chaînes d'approvisionnement qui ont été expliqués au chapitre 3. Afin de modéliser la structure mathématique, nous nous servons des problèmes de prise de déci-

	Les éléments du problème de localisation-allocation		
	Établissements	**Localisation**	**Demandes**
Information d'entrée	- Emplacement actuel ou potentiel - Les coûts d'installation	- Plan d'expédition - Coûts d'expédition	- Demandes effectuées (déterministes ou probabilistes)
Information de sortie	- Le nombre optimal d'établissements - La capacité optimale - L'emplacement optimal	- Le reseaux optimale d'expédition - Allocation optimale aux demandes	- Allocation optimale de chaque établissement aux demandes pertinentes

FIGURE 6.1 – Les éléments du problème localisation-allocation

sion hybrides tels que le problème de localisation-allocation conjointe et utilisons des méthodes méta-heuristiques pour résoudre le problème mathématique et obtenir les valeurs optimales des variables liées de l'architecture logistique du lac de données. Notre modèle aborde l'optimisation de l'architecture du lac de données basée sur la définition de la chaîne d'approvisionnement et les stratégies de gestion du réseau logistique.

Dans ce chapitre nous allons présenter :

- Le problème conjoint de localisation-allocation ;
- Le modèle mimétique pour l'optimisation de l'architecture du lac de données basé sur les principes du problème hybride de localisation-allocation ;
- La description et la formulation du modèle mathématique ;
- Les solutions méta-heuristiques pour résoudre le modèle mathématique défini.

6.2 Le problème conjoint de localisation-allocation

Le modèle de localisation-allocation conjointe est un problème populaire et pratique dans les stratégies de prise de décision hybrides qui concerne l'élaboration de la structure logistique d'une chaîne d'approvisionnement de manière optimum. Les objectifs de ce modèle sont de trouver et de déterminer les meilleurs choix pour le nombre et la localisation des installations disponibles et d'allouer ces installations aux demandes réparties de telle sorte que les coûts totaux de la chaîne soient minimisés ainsi que la rentabilité est augmenté [Cooper, 1963]. Chaque problème localisation-allocation se compose de trois éléments principaux et pour chaque élément, il existe des informations d'entrée et des informations de sortie qui sont répertoriées dans le tableau 6.1.

Comme nous l'avons expliqué au chapitre 3, l'idéologie du problème des stratégies hybrides est de mettre en œuvre simultanément des décisions à long (stratégique), moyen (tactique) et court terme (Pratique), afin de prendre en compte tous les horizons de modélisation des réseaux

<table>
<tr><th>Objectifs globaux</th><th>Objectifs locaux</th><th>Les paramètres d'entrées</th><th>Décisions variables</th><th>Hypothèse</th><th>Fonctions objectives</th><th>Contraintes</th></tr>
<tr><td rowspan="4">Minimisation des coûts totaux</td><td rowspan="2">Nombre optimal d'installations</td><td>Nombre des demandes</td><td rowspan="2">Coût total optimal</td><td rowspan="2">La demande de chaque client peut être fournie par plusieurs installations</td><td rowspan="4">Minimisation (coûts de lancement de l'installation, frais d'expédition, frais d'affectation, frais de préparation des demandes, frais de risques ou de perte de clients, frais de retards)</td><td rowspan="2">Capacité d'installation</td></tr>
<tr><td rowspan="2">Montant de la demande du client</td></tr>
<tr><td rowspan="2">Emplacement optimal des installations</td><td rowspan="2">Nombre optimal d'installations</td><td rowspan="2">La demande de chaque client doit être couverte au moins avec une installation</td><td rowspan="2">Activation d'installations</td></tr>
<tr><td rowspan="3">Maximisation de la rentabilité</td><td rowspan="2">Montant optimal d'allocation des installations aux demandes</td><td rowspan="2">Coût de lancement (activation) de l'installation</td><td rowspan="3">Quantité optimale allouée à chaque demande par chaque installation</td><td>Toutes les demandes doivent être satisfaites</td><td rowspan="2">Maximisation (profit, service client, rendement)</td><td rowspan="3">Attribution des demandes</td></tr>
<tr><td rowspan="2">Capacité optimale de chaque installation</td><td rowspan="2">Capacité de l'installation</td><td rowspan="2">La demande de chaque client doit être attribuée à l'installation active (ouverte)</td></tr>
</table>

FIGURE 6.2 – Les éléments du modèle mathématique général du problème localisation-allocation

logistiques et intégrer la chaîne d'approvisionnement. Dans le problème de localisation-allocation, les décisions stratégiques concernent la recherche des meilleurs emplacements, le remplacement et l'activation d'un ou plusieurs installations actuels ou nouveaux, ainsi que la conception des réseaux d'expédition au sein de la chaîne. D'autre part, les décisions tactiques contiennent toutes les stratégies pour identifier les fonctions requises et affecter les capacités des installations actives aux demandes formulées. Il est à noter que toutes ces stratégies sont prises ensemble considérant que la chaîne d'approvisionnement est optimisée en ce qui concerne la minimisation des coûts totaux, des risques et des délais et la maximisation de la qualité des services et des profits. Afin de réaliser ce problème conjoint et ses objectifs principaux quelques modèles généreux sont proposés par les chercheurs dans ce domaine. On se réfère aux travaux de [Azarmand and Neishabouri, 2009, Cooper, 1963] pour présenter la modélisation générale de ce genre de problème. De manière générale, les éléments pour réaliser un modèle mathématique d'optimisation du problème de localisation-allocation sont répertoriées selon la figure 6.2.

6.2.1 Modèle mathématique

En ce qui concerne des éléments de figures 6.2, le modèle général est formalisé comme décrit ci-après.

- Les ensembles

I Ensembles des demandes de clients potentiels $i = 1, ...n$

J Ensembles des établissemens candidats $j = 1, ...m$

- Les paramètres

Ca_j Capacité d'établissemen j

D_i Demande de client i

f_j Coût de lancement d'établissemen j

V_{ij} Coût d'expédition d'établissemen j à client i

- Les décisions variables

X_{ij} Quantité de demande du client i couverte par l'installation j

$Y_j = \begin{cases} 1 & Si\ l'intssllstion\ j\ est\ active \\ 0 & Sinon\ (\forall j \in J) \end{cases}$

- La fonction objectif

Les coûts de lancement des installations :

$$\sum_{j \in J}^{m} Y_j F_j \tag{6.1}$$

Les coûts de la satisfaction de la demande client :

$$\sum_{i \in I}^{n} \sum_{j \in J}^{m} X_{i,j} D_i V_{ij} \tag{6.2}$$

- Le modèle mathématique général

Selon les paramètres définis, le modèle mathématique général est modélisé comme suit :

$$Min : \sum_{i \in I}^{n} \sum_{j \in J}^{m} X_{i,j} D_i V_{ij} + \sum_{j \in J}^{m} Y_j F_j \tag{6.3}$$

Subject to :

$$\sum_{j \in J} X_{i,j} = D_i \qquad \forall i \in I \tag{6.4}$$

$$\sum_{j \in J} X_{i,j} D_i \leq Ca_j \qquad \forall i \in I \tag{6.5}$$

$$X_{i,j} \leq Y_j \qquad \forall j \in J \;\; i \in I \tag{6.6}$$

$$X_{i,j} \geq 0 \qquad \forall j \in J \;\; i \in I \tag{6.7}$$

$$Y_j \in 0, 1 \qquad \forall j \in J \tag{6.8}$$

L'équation 6.3 minimise les coûts totaux de la chaîne d'approvisionnement qui contient les coûts de lancement des installations et les coûts de couverture des demandes des clients. L'équation 6.4 garantit que toutes les requêtes sont couvertes par des facilités ouvertes. L'équation 6.5 représente les contraintes de capacités des installations et l'équation 6.6 indique que la demande des utilisateurs ne peut être affectée à un installation que si cet installation est actif. Enfin 6.7 et 6.8 indiquent le type de variables de décision.

La résolution de ce modèle mathématique s'oriente vers l'optimisation de la structure de la chaîne logistique qui prend en compte les stratégies hybrides pour obtenir les valeurs optimales des paramètres de décision dans la méthode de la gestion des réseaux logistiques.

Sur cette base, en faisant l'analogie avec les lacs de données, nous présentons un modèle d'optimisation pour l'architecture du lac de données qui prend les éléments du modèle mathématique de construction générale du problème de localisation-allocation. Ce modèle est envisagé pour développer la structure et la fonctionnalité du lac de données avec des outils et des stratégies imitant les systèmes logistiques tels que la chaîne d'approvisionnement.

6.3 Optimisation du lac de données

Avec la révolution des systèmes d'information et l'émergence du concept de gestion centralisée des données, l'optimisation de l'infrastructure du système de stockage des données devient un sujet incontournable pour les organisations. Le triptyque de bonnes plates-formes, de bonnes données et de bonnes interfaces est un élément important du déploiement d'un lac de données efficace [Kachaoui and Belangour, 2020]. Considérant qu'une bonne plate-forme de données influence les performances du lac de données, l'optimisation de l'architecture pourrait être une étape importante dans le développement de l'ubiquité de ce système de stockage de données en tant que outil indispensable pour les grandes entreprises. Normalement, l'optimisation du lac de données se fait dans l'exécution des requêtes, ce qui réduit le temps de préparation, le processus de traitement des données et le coût de stockage [Maccioni and Torlone, 2018, Rohde and Vidal, 2020]. Cependant, pour parvenir à une architecture optimisée, il est nécessaire que les méthodes d'optimisation s'appliquent tout au long du cycle de vie des données, de l'ingestion dans le lac à l'exploitation par les utilisateurs. Nous appelons cette optimisation, "*optimisation globale*" ou "*optimisation compréhensive*" qui aborde toutes les considérations pour concevoir une architecture optimisée. Pour cette raison, nous nous référons à l'architecture de lac de données au chapitre 4 que nous avons définie comme une structure de systèmes logistiques qui couvre toutes les procédures d'approvisionnement de données. Sur la base des définitions des systèmes logistiques, les stratégies de gestion et d'optimisation de ces structures, tels que les modèles décisionnels hybrides, peuvent être utilisés pour modéliser les méthodes d'optimisation des lacs de données.

Selon cette architecture et en se concentrant sur la phase optimale, pour chaque couche telle que la couche d'ingestion, la couche de stockage, la couche de traitement, la couche d'accès et la couche de gouvernance des données qui indiquent les différentes phases du cycle de vie des données au sein du lac de données, des décisions communes (stratégiques et tactiques) doivent être prises. Ces décisions permettent d'avoir une perspective complète afin d'optimiser l'infrastructure en fonction des coûts et bénéfices envisagés pour être respectivement minimisés ou maximisés. Il

convient d'indiquer la précision ci-dessous dans notre modèle d'optimisation :

Les coûts concernent tous les processus qui augmentent le temps de service ou imposent de la latence à la fonction de lac de données et d'autre part, la rentabilité est liée à toutes les étapes et les performances qui valorisent et améliorent la qualité de service aux utilisateurs.

Avant d'aborder la conception de modèles d'optimisation de lac de données, nous nous appuierons sur les coûts et les bénéfices qui sont généralement spécifiés pour chaque couche appartenant au lac de données afin de clarifier les critères qui doivent être minimisés ou maximisés pour achever un lac de données optimisé. En général, le lancement d'un système informatique comporte trois coûts principaux tels que les coûts de préparation du matériel, les coûts de chargement et de support des logiciels et les coûts d'opération [Han et al., 2011]. D'autre part, dans notre étude, nous nous concentrons sur les coûts associés à la performance et à la performance du système dans l'environnement Hadoop, qui sont les coûts d'opération plutôt que les deux autres coûts mentionnés. Évidemment dans les méthodes d'optimisation, nous essayons d'obtenir des solutions qui permettent de minimiser les risques et les coûts éventuels et de maximiser les profits potentiels.

6.3.1 Couche d'ingestion

La couche d'ingestion est responsable de l'acquisition des données hétérogènes et brutes à partir de plusieurs sources. La vitesse et l'évolutivité de cette couche dans l'acquisition des données, la transformation des données en structure ciblée et leur entrée dans le lac de données, sont les caractéristiques importantes que cette couche doit signaler. La compatibilité des API pour supporter diverses charges de données ainsi que pour être intégrées aux systèmes et techniques de la couche de stockage, fait de cette couche un élément décisionnel dans l'architecture du lac de données. De plus, cette couche doit pouvoir contrôler ou synchroniser le taux de consommation et le taux d'ingestion de données afin d'éviter un traitement tardif pour les utilisateurs [Tomcy, 2017]. Les objectifs principaux de cette couche sont d'acquérir tous les types de données (structurées, semi-structurées et non structurées) avec un minimum d'effort et un temps d'ingestion réduit. Le temps d'ingestion est une métrique importante pour évaluer l'efficacité d'une architecture de lac de données. Le temps d'ingestion signifie le temps convenable pour saisir et stocker des données dans le système de stockage et pourrait être défini avec la formule suivante selon [Rangarajan et al., 2015] :

$$T_{in} = T_m - T_a \tag{6.9}$$

Dans cette formule, T_{in} indique le temps d'ingestion qui dérive de la différence entre le temps d'accès aux données T_m et le temps d'arrivée de données T_a.

Afin de considérer la fonctionnalité de la couche d'ingestion sur les performances optimisées du lac de données, il faut déterminer et analyser tous les coûts et avantages pertinents de cette couche en fonction des technologies choisies. Étant donné que nous avons défini les deux technologies rentables comme Apach Tika (pour extraction de métadonnées), Apache Flume (pour

l'acquisition de données) et Apache Atlas (pour getsion de métadonnées). Nous déterminerons les coûts et les avantages remarquables pour la couche d'ingestion comme décrit ci-dessous.

Coûts

- La latence pour ingérer des données à haute vitesse en temps réel et les rendre disponibles pour l'analyse [Nargesian et al., 2019] ;
- Le temps de lancement des jobs de Apache Flume afin de stocker les métadonnées, importer et exporter les commandes et les données [Mătăcuţă and Popa, 2018] ;
- Le temps nécessaire pour charger (ingérer) les données dans l'environnement du lac de données et extraire les métadonnées [Zhao et al., 2021] ;
- Le temps des requêtes API à communiquer aux applications afin d'accéder aux données ou aux services [Rooney et al., 2019] ;
- La latence de capture des données qui entraîne une perte de données [Gupta and Giri, 2018].

Bénéfices

- Ingestion et transfert de données avec haut degré de parallélisme [Nargesian et al., 2019] ;
- La haute capacité de gestion, la fiabilité, et l'évolutivité [Tomcy, 2017, Mătăcuţă and Popa, 2018] ;
- La capacité d'accueillir tous les types de données [Pradeep, 2015] ;
- La vitesse de capture et d'échange des données [Mătăcuţă and Popa, 2018] ;
- La haute compatibilité avec les sources de données externes [Rooney et al., 2019] ;
- La haute flexibilité et extensibilité [Gupta and Giri, 2018].

6.3.2 Couche de stokage

La couche de stockage reçoit et stocke les données dans ses structures d'origine au fur et à mesure qu'elles sont récupérées et ceci sans connaitre a priori les besoins à l'avenir. En effet, cette vertu permet de réduire les surcoûts de transformation, de normalisation, ou de préparation des données avant stockage par rapport aux systèmes traditionnels tels que les entrepôts de données [Llave, 2018]. Cependant, il demeure encore des coûts considérables liés à cette couche qui doivent être pris en compte. Dans l'environnement Hadoop, le système de stockage HDFS est utilisé comme une solution fiable et compatible pour stocker des données semi-structurées et non structurées via des réseaux distribués de nœuds [Holmes, 2017]. Pour cette raison, les coûts pertinents à minimiser et les bénéfices potentiels à maximiser de cette couche sont définis comme décrit ci-dessous.

Coûts

- Le risque de défaillance des nœuds(*NameNode* et *DataNode*), disques, cluster et la perte de fiabilité [Venner, 2009] ;
- Le risque de surcharge de stockage élevée, la latence de lecture dégradée, le temps de reconstruction élevé et le trafic réseau et disque [Xia et al., 2015] ;
- Le coût de la latence d'accès aux données a cause de fournir un débit de données élevé [White, 2012] ;
- Le coût lié à la vitesse de récupération, transfert, et de recherche des données dans les blocs [Mătăcuţă and Popa, 2018] ;
- Le coût (temps) de la réplication de blocs pour accélérer l'exécution des jobs MapReduce dans les blocs répliqués et réduire le risque de défaillance des blocs [Dev and Patgiri, 2014] ;
- Le temps correspondant à la gestion et le stockage de données sur plusieurs nœuds du cluster [Dev and Patgiri, 2014] ;
- Le temps pour stocker, mettre à jour et calculer les données en temps réel qui sont collectées à partir de Apache Flume [Mătăcuţă and Popa, 2018] ;
- Le temps de fusion des résultats des données par lots et en temps réel pour l'étape de recherche des données [Ramakrishnan et al., 2017] ;
- Le temps de déplacer de grandes quantités de données, de gestion et d'extraction de métadonnees [Skluzacek et al., 2021] ;
- Le coût d'un taux déséquilibré de flux de données qui entraîne des files d'attente de stockage inutiles [Oussous et al., 2018] ;
- Le coût de la gravité des données [Madera and Laurent, 2016, Zagan and Danubianu, 2021] ;
- Le coût du nettoyage et de la transformation de la structure des données dans les formats requis par les utilisateurs [Joaquim and dos Santos Mello, 2020] ;
- Temps et coût pour la vérification de l'intégrité, la taille du fichier, le nombre de colonnes et d'enregistrements, le schéma, la duplication, le traçage et la corruption des données [Pradeep, 2015] ;
- Le coût de la perte de valeur des données en raison d'une longue période de conservation [Pradeep, 2015] ;
- Le coût correspondant aux formats de stockage et de compression des données (les différents formats affectent la vitesse de cette couche) [Belov and Nikulchev, 2021] ;
- Le coût de marécage de données (ralentissement de cette couche à cause de l'accumulation de données inutiles) [Paschalidi, 2015, Hai et al., 2016].

Bénéfices

- La haute évolutivité et la tolérance aux pannes pour les nœuds de clusters de machines [Liu et al., 2021] ;
- Les mécanismes flexibles d'accès aux données [Oussous et al., 2018] ;
- La haute compatibilité avec tous les modèles structurels et formats de données [Belov and Nikulchev, 2021] ;
- La vitesse de vérification et de stockage des données [Misaki et al., 2016, White, 2012, Venner, 2009, Oussous et al., 2018] ;
- La grande simultanéité de lecture et d'écriture [Misaki et al., 2016, Pradeep, 2015, Dev and Patgiri, 2014].

6.3.3 Couche de traitement

La couche de traitement est la couche la plus importante de l'architecture des systèmes de gestion de données qui influence grandement la fonctionnalité du système et la vitesse des services. La grande majorité des coûts qui imposent le lac de données, ainsi que les avantages qui peuvent augmenter la fonctionnalité du système, concernent cette couche. Compte tenu des exigences de la couche de traitement, il est évident que ses performances optimales se manifestent dans l'optimisation globale du lac de données. Selon l'architecture définie, on a choisi la technologie MapReduce pour la couche de traitement de lac de données. Pour cette raison, nous nous appuyons sur l'approche générale de cette technique pour traiter les données afin de pouvoir spécifier les tâches pertinentes pour cette couche à optimiser.

MapReduce est un système de traitement de données distribué comme HDFS, un élément important de l'environnement Hadoop qui traite les données volumineuses stokées dans HDFS en parallèle sur de nombreux nœuds de calcul et avec en temps de traitement réduit [Holmes, 2017]. Un *job* en Mapreduce est constitué d'un programme qui effectue des traitements sur des ensembles d'entrées de données par paires (clé, valeurs) et génère des paires intermédiaires et ensuite fusionne les résultats pour créer de nouvelles sorties par paires (clés, valeurs) avec deux composants (opérateurs) principaux *Map* et *Reduce* [Dean and Ghemawat, 2008]. L'anatomie du MapReduce fonctionne sous l'architecture de maitre-esclave comme décrit ci-dessus :

1. Un ***Job*** est soumis par la machine utilisateur dans l'environnement Hadoop.

2. Le Job soumis est transmis à la machine maître et géré avec ***JobTracker*** qui est un élément de MapReduce et est responsable de la gestion des données d'entrée et de sortie.

3. ***JobTracker*** affecte les MapReduce ***TaskTrackers*** appropriés, qui sont situés en machine esclave, pour exécuter le Job en communiquant avec le ***Name Node***

4. Le ***TaskTracker*** démarre les fonctions du MapReduce (Map et Réduire) et actualise du Job en communiquant avec JobTracker.

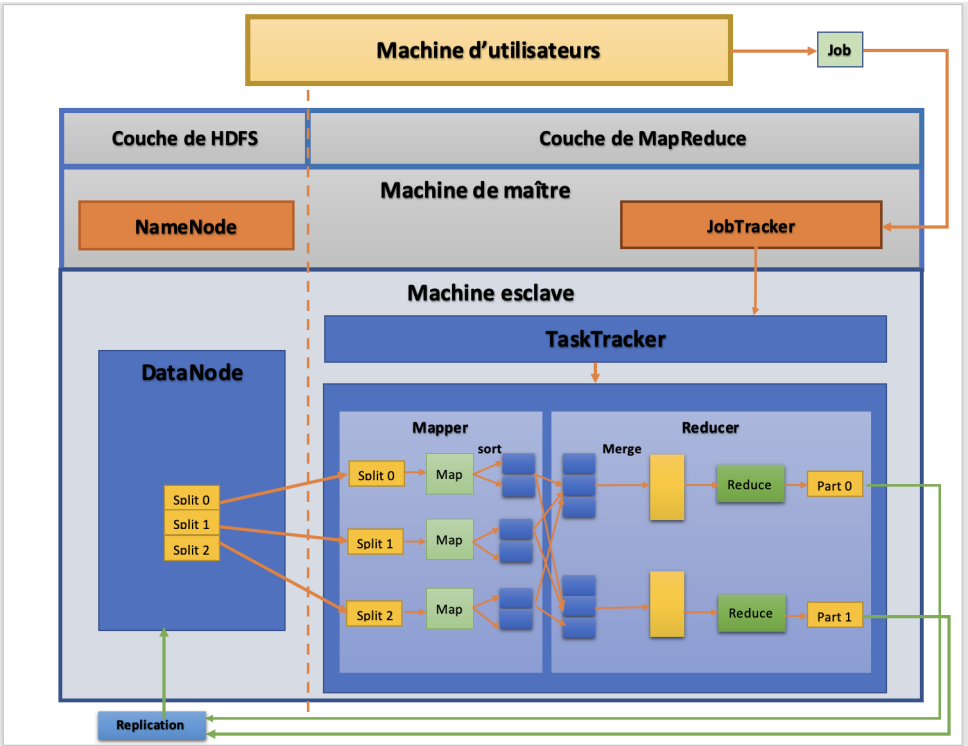


FIGURE 6.3 – Anatomie de MapReduce

5. La fonction ***Map*** commence ses tâches par la lecture des données de HDFS, puis divise, trie, combine, compresse les données avec des ***Mappers*** et fusionne les données triées.

6. La fonction ***Reduce*** commence par les sorties de ***Map***, puis attribue les mêmes valeurs aux clés pertinentes, puis fusionne les résultats avec les ***Reducers***

7. Les sorties de la phase de ***Reduce*** sont écrites sur HDFS.

L'approche ci-dessus est illustrée par la figure 6.3 :

En ce qui concerne la démarche MapReduce, il est évident que l'optimisation de fonctions pertinentes telles que Map et Reduce pourrait augmenter la vitesse de traitement ainsi que la qualité des services de cette couche. A ce titre, les frais relatifs à la réalisation d'un requête et au traitement des données avec MapReduce, incluent les coûts de réalisation d'un Job. De plus, le nombre minimum de Jobs pour effectuer une requête pourrait être un critère remarquable pour améliorer les performances du traitement des jeux de données. À la suite de cette exigence, [Peyravi and Moeini, 2020] travaille sur l'optimisation du pipeline de Mapreduce et les auteurs indiquent que le temps d'exécution d'un Job est le temps d'exécution total des étapes Mapper et Recucer comme l'équation 6.10 remarque :

$$T_{Job} = T_{Mapper} + T_{Reducer} \tag{6.10}$$

Ensuite, sur la base de cette logique et selon l'anatomie de MapReduce, les phases Mapper

et Reducer comprennent plusieurs étapes dont le temps d'exécution de chaque étape génère le temps total d'exécution du Job est calculé comme l'équation 6.11 :

$$T_{Job} = (T_{read} + T_{map} + T_{collect+T_{split}} + T_{merge}) + (T_{shuffle} + T_{reduce} + T_{write}) \tag{6.11}$$

Pour indiquer l'importance des impacts des fonctionnalités des phases Map et Reduce, on pourra se référer aux travaux de [Husain et al., 2011] qui a mené une étude sur la révision de Jobs importants liés aux requêtes SPARQL qui sont répondues avec le framework MapReduce. Selon cette étude, un Job lié à une requête SPARQL contient les coûts concernés aux fonctions MaprReduce pour traiter ce Job. Les coûts pour Job i sont formalisés comme suit :

$$Job_i = MI_i + MO_i + RI_i + RO_i \quad where \;\; i < n \tag{6.12}$$

L'équation 6.12 indique qu'un Job de traitement d'une requête avec MapReduce, contient quatre coûts importants liés à la lecture, au tri et à l'écriture :

- MI_i : Le coût de phase d'entrée de Map qui contient le nombre total de paires *clés-valeurs* saisies ;
- MO_i : Le coût de phase de sorties de Map qui équivaut à MI_i s'il n'existe pas de bornes ;
- RI_i : Le coût de phase d'entrée de Reduce qui contient Le nombre de paires clé-valeur qui sont triées dans RI ;
- RO_i : Le coût de phase de sorties de RI qui est facultatif et est fixé pour une requête.

En achevant ce travail, [Herodotou and Babu, 2011] a proposé les configurations d'optimisation des programmes MapReduce en fonction des coûts liés à l'exécution des Jobs. Les auteurs ont défini le Job i avec des éléments importants comme $i = (p, d, r, c)$ dans lequel p présente le programme MapReduce, d indique les données d'entrée, r présente les ressources, et c montre les choix des paramètres de configuration pour exécuter un Job dans MapReduce. Selon les auteurs, pour optimiser les performances de MapReduce, il est nécessaire d'ajuster et de paramétrer les paramètres liés à l'exécution des Jobs tels que le nombre de Mapper et de Reducer pour le Job i, l'allocation de mémoire à chaque Mapper et Reducer, et l'état de compression ou la combinaison de données pour chaque phase de Map ou de Reduce. Ils ont défini que l'optimisation de MapReduce signifie trouver les meilleurs réglages pour les paramètres d'exécution d'un Job.

La planification des Jobs et la gestion des ressources sont d'autres paramètres influents qui peuvent déclencher l'optimisation des performances de MapReduce. Par conséquent, [Hashem et al., 2020] a révisé des algorithmes pour l'ordonnancement des Jobs et a montré qu'un algorithme optimisé pouvait réduire considérablement le temps d'exécution et les coûts de traitement. En outre, [Rasooli and Down, 2014] a proposé un système de planification des tâches appelé COSHH qui pourrait développer l'optimisation de Hadoop en tenant compte d'importants indicateurs de performance tels que: la localité, l'équité, la satisfaction minimale de la partie et le temps moyen jusqu'à l'achèvement des tâches. En effet, Les mécanismes d'optimisation des tâches et des Jobs

du MapReduce sont les enjeux problématiques qui sont menées par plusieurs études au cours des années dernières et se concentrent sur les coûts pertinents de planification et d'exécution des Jobs [Gu et al., 2014].

Sur la base de ces deux travaux importants, les stratégies d'optimisation qui orientent cette couche vers la minimisation des exécutions spéculatives dans les nœuds et la recherche de la valeur optimale du temps d'exécution pour les Jobs MapReduce. Pour cette raison, nous pouvons nommer les coûts à minimiser et les bénéfices à maximiser correspondants de la couche de traitement comme ci-dessus :

Coûts

- Le coût lié au nombre de Jobs [Peyravi and Moeini, 2020, Herodotou and Babu, 2011, Hashem et al., 2020] ;
- Le coût d'exécution et d'achèvement des Jobs [Gu et al., 2014] ;
- Le temps moyen d'exécution des tâches des fonctions Map et Reduce [Gu et al., 2014] ;
- Les coûts pertinents du partitionnement des données d'entrée, des pannes de machine et de la planification du cluster de machines [Dean and Ghemawat, 2008] ;
- Le temps des calculs à grande échelle [Dean and Ghemawat, 2008, Casado and Younas, 2015] ;
- Le risque de ralentissement de la machine [1] en raison d'un traitement lent qui impose un temps supplémentaire pour terminer les tâches Map et Reduce [Hashem et al., 2020] ;
- Les coûts de la grande quantité de données envoyées sur le réseau et le coût de la consommation de bande passante [Dean and Ghemawat, 2008] ;
- Le coût de latence des Jobs et les temps de service [Gu et al., 2014] ;
- Le coût de la communication dans le système Hadoop et les coûts de recherche d'une allocation de Jobs et de ressources [Rasooli and Down, 2014, Husain et al., 2011].

Bénéfices

- L'évolutivité d'implementer les Jobs en clusters de machines [Dean and Ghemawat, 2008] ;
- La haute disponibilité et la tolérance de pannes des machines [Pradeep, 2015, Casado and Younas, 2015] ;
- La détection élevée des pannes des machines et les tâches lentes ;
- L'utilité maximale du Job [Peyravi and Moeini, 2020] ;
- La haute qualité et la fiabilité des données traitées [Miloslavskaya and Tolstoy, 2016] :
- La capacité élevée de distribution et de parallélisation des données [Zhao et al., 2016] ;

[1] Straggler

- L'utilisation optimale des ressources du système [Munshi and Mohamed, 2018] ;
- Le débit rapide de données [White, 2012, Walker and Alrehamy, 2015] ;
- La grande vitesse de traitement des données [Giebler et al., 2021] ;
- La localité avancée des tâches en resources [Rasooli and Down, 2014] ;
- La flexibilité de traitements hybrides (en lots et en temps réel) [Giebler. et al., 2018].

6.3.4 Couche d'accès

La couche d'accès est la couche de consommation de données et la dernière phase du cycle de vie des données. Afin de répondre aux demandes des utilisateurs avec une haute qualité de service, il est impératif de lancer une interface interrogeable capable d'améliorer et d'analyser les données en temps réel et avec une vitesse et une fiabilité acceptables. Le lac de données est un système puissant de valorisation des données hétérogènes et multi-structures est compétent de charger les nombreux outils analytiques et algorithmes dans l'environnement Hadoop [Rangarajan et al., 2015].

En général, on pourrait manifester les coûts et les avantages remarquables associés à cette couche comme décrit ci-dessous.

Coûts

- Le coût lié au temps de recherche de la découverte basée sur les requêtes (recherche d'index), la découverte basée sur la navigation ou l'exploration (recherche basée sur un graphique de liaison ou basée sur une structure hiérarchique) et la découverte basée sur l'analyse [Nargesian et al., 2019] ;
- Le coût associé du temps de la transformation des données en fonction des requêtes effectuées(ELT) [Liu et al., 2021] ;
- Le coût lié au temps des mises en correspondance (mappings) entre modèles de données génériques [Hai et al., 2016].

Bénéfices :

- La rapidité de découverte et d'analyse des connaissances enrichissantes [Nargesian et al., 2019] ;
- La capacité de service aux utilisateurs [White, 2012] ;
- L'aptitude à la découverte sémantique [Giebler et al., 2021].

6.3.5 Couche de la gestion de métadonnées et la gouvernance de données

Comme nous l'avons expliqué, dans l'architecture du lac de données, les couches de gestion des métadonnées et de gouvernance des données sont les couches organisationnelles qui ont des rôles gouvernementaux pour gérer et inspecter les données tout au long du cycle de vie. Ces couches sont responsables de la traçabilité, de la sécurité, de la filiation, de la validité, de l'intégrité, de la purification, de la légitimité, de la justification des accès et de la lisibilité des données.

Sur la base de l'analogie entre l'architecture du lac de données et la structure de la chaîne d'approvisionnement, ces couches sont considérées comme le service d'évaluation et la gestion de la traçabilité des produits dans la chaîne d'approvisionnement. D'autre part, dans les méthodes de la prise de décision conjointe pour optimiser les systèmes logistiques, l'accent est mis sur les problèmes qui ont le plus d'impact sur les coûts et avantages globaux et la fonctionnalité de la chaîne d'approvisionnement. En effet, la présence de ces couches est indispensable pour la mise en place et le stockage des données dans un système de stockage de données. En termes de responsabilités, des systèmes de métadonnées exhaustives et des protocoles complets de gouvernance des données sont les déclencheurs de l'ajout de valeurs au lac de données. En revanche, ces couches n'imposent pas de coûts importants liés au temps d'exécution du système, mais le manque de définition de la réglementation et des politiques globales grâce à ces couches, pourrait remettre en cause la fiabilité et la traçabilité des données. La technologie qui est souvent utilisée pour prendre en charge ces couches dans les écosystèmes Hadoop est Apache Atlas qui couvre toutes les tâches correspondantes de classification, de sécurité, d'audit centralisé, d'intégrité, de lignage et de gestion des ressources, à travers toutes les couches d'ingestion, de stockage, de traitement et d'accès de données [Rangarajan et al., 2015].

On pourrait soutenir que les coûts remarquables de ces couches qui devraient être minimisés sont liés à des opérations chronophages qui ont un impact sur le système, telles que l'indexation, le stockage, la cartographie des schémas et les mécanismes de vérification et d'autorisation des données [Pradeep, 2015, Rangarajan et al., 2015, Tomcy, 2017].

6.4 Description et formulation du problème

L'objectif principal de cette étude est l'optimisation de l'architecture du lac de données. Pour atteindre cet objectif, un parcours analogique a été emprunté de la structure du lac de données avec des systèmes logistiques aux stratégies pratiques de la chaîne d'approvisionnement pour gérer et optimiser l'architecture du lac de données axée sur la logistique. Dans la section 6.2, nous avons abordé une stratégie efficace avec la formulation du modèle pour optimiser une chaîne d'approvisionnement appelée problèmes conjoints de localisation-allocation. Afin d'expliquer le fonctionnement de cette stratégie mimique pour les lacs de données, nous allons reconsidérer l'analogie entre l'architecture des lacs de données et les réseaux de chaîne d'approvisionnement. Dans la chaîne d'approvisionnement, le produit est approvisionné à partir de ressources, est fabriqué par les fabricants, est conservé dans des entrepôts et est distribué et livré aux clients par les distributeurs et les détaillants. La méthode de conception de la chaîne d'approvisionnement recherche les meilleurs moyens de fournir, d'entretenir, de transporter et de servir le produit aux consommateurs dans manière que le temps de service est réduit, les coûts sont minimisés et la

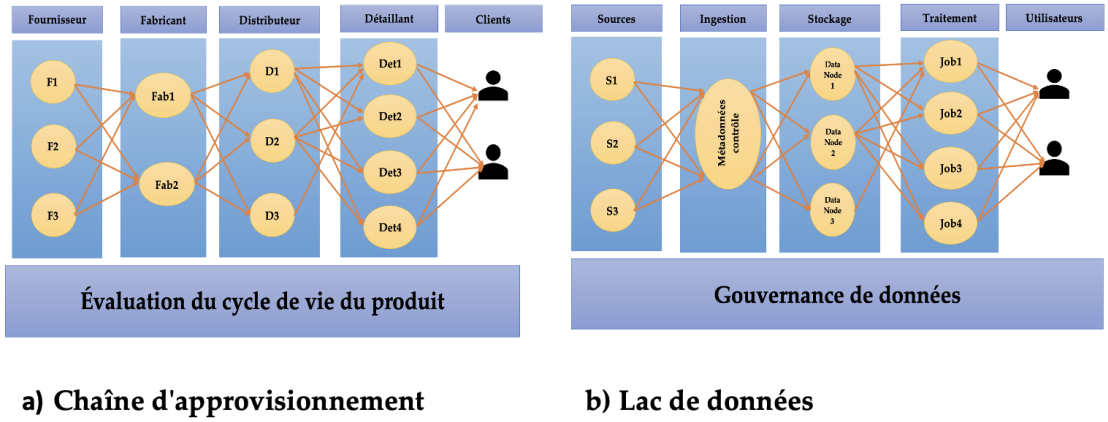


FIGURE 6.4 – Analogie de la structure de la chaîne d'approvisionnement et du lac de données

couverture de la demande est maximisée. Pour atteindre ces objectifs, de nombreuses décisions stratégiques et tactiques sont prises pour être appliquées telles que le nombre de fournisseurs, le nombre et l'emplacement des entrepôts actifs, le nombre et l'emplacement des distributeurs, le nombre et l'emplacement des détaillants, la planification des itinéraires, le contrôle des stocks (le temps et la quantité du réapprovisionnement, de recharge et de livraison des produits) et l'attribution des produits et services aux clients.

Compte tenu de la similitude entre les structures logistiques de la chaîne d'approvisionnement et du lac de données qui est présentée dans la figure 6.4, on pourrait appliquer ces stratégies de décision de la même manière pour le lac de données.

Les données du point de vue du lac de données basées sur les systèmes logistiques sont considérées comme un produit de la chaîne d'approvisionnement qui traverse toutes les couches, de la source aux utilisateurs. Comme la figure 6.4 le montre, dans chaque couche de la chaîne d'approvisionnement, il y a des décisions stratégiques, tactiques et pratiques à mettre en évidence telles que le nombre de fournisseurs, le nombre et l'emplacement des fabricants, les quantités de produit à distribuer, la planification des itinéraires, le nombre et les emplacements des entrepôts, et le nombre et l'emplacement des installations de vente au détail. De même dans le lac de données, chaque couche doit être gérée de manière intégrée et efficace afin d'augmenter les rendements globales du système. Des études menées révélant que la gestion stratégique et tactique de la performance de chaque couche peut améliorer la fonctionnalité du lac de données [Gu et al., 2014, Rasooli and Down, 2014, Hashem et al., 2020, Herodotou and Babu, 2011, Dean and Ghemawat, 2008, Belov and Nikulchev, 2021, Rooney et al., 2019]. Par conséquent, il est envisagé de prendre les décisions stratégiques ou tactiques pour les données selon l'architecture de la figure 6.4 et la stratégie de prise de décision conjointe location-allocation pour optimiser le lac de données comme l'illustre la figure 6.5.

Dans cette étude, nous nous concentrons sur les décisions stratégiques et tactiques du lac de données telles que la gestion de la gouvernance de données, le nombre optimal de Jobs de Mapreduce et la couvertures de demandes d'utilisateurs pour optimiser l'architecture du lac de

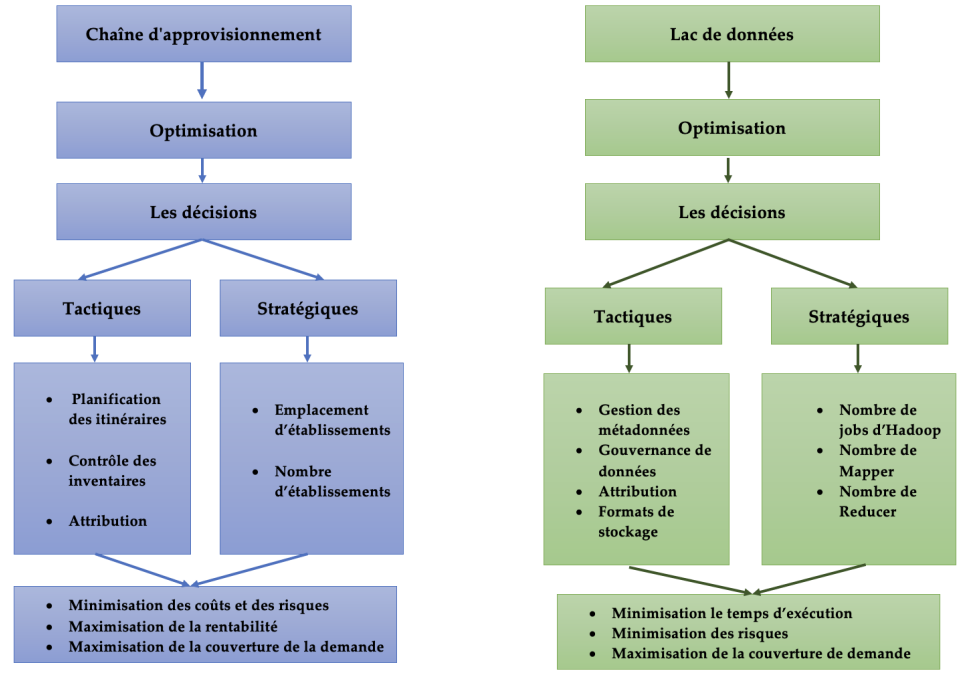


FIGURE 6.5 – L'analogie des approches d'optimisation dans la chaîne d'approvisionnement et le lac de données

données afin d'atteindre le minimisation les coûts liés au temps d'exécution.

D'autre part, dans la section 6.3, nous avons révisé les coûts et les bénéfices des couches principales liées á l'architectures de lac de données qui doivent être respectivement minimisés ou maximisés, afin d'obtenir une infrastructure optimale et des performances de haute qualité. Dans la section qui vient, nous nous appuyons sur le modèle mathématique d'optimisation de la structure des lacs de données basé sur les concepts et le modèle de problèmes conjoints de localisation-allocation en section 6.2 et en considérant les coûts et les avantages des performances et de l'architecture du lac de données en section 6.3.

Avant d'aborder la formulation mathématique du problème, nous expliquons quelques termes principaux en fonction du modèle d'optimisation.

Définition 6.1 (***Optimisation de l'architecture de lac de données***). *Ensembles de stratégies et de décisions prises en fonction de l'infrastructure du lac de données afin de développer les performances et l'efficacité du système tout en minimisant les coûts globaux et en maximisant la rentabilité.*

Selon l'architecture proposée au chapitre 4, un système logistique de lac de données contient des couches telles que des niveaux (membres) dans la chaîne d'approvisionnement qui fournissent, préparent et livrent des données aux utilisateurs finaux. Chaque couche génère les coûts et les avantages pour le lac de données qui doivent être pris en compte pour optimiser le lac de données.

Définition 6.2 (***Les coûts***). *Ensemble de fonctions qui augmentent le temps d'ingestion, de stockage, de traitement et de consommation des données*

Dans notre modèle proposé, le temps d'ingestion des données, le temps d'exécution des Jobs, le temps MapReduce et le temps de servir les utilisateurs sont les coûts les plus importants à minimiser.

Définition 6.3 (***Les avantages***). *Ensemble de fonctions qui ajoutent des valeurs au lac de données ou améliorent les performances du système*

Sur la base de la **Définition** 6.3, la vitesse et la qualité des données sont des avantages plus importants qui devront être maximisés au sein du lac de données.

Définition 6.4 (***La fonction objectif***). *La fonction principale du processus d'optimisation qui cherche la meilleure solution du problème en minimisant les coûts et les risques ou en maximisant les valeurs définies en fonction des contraintes appliquées au problème.*

Dans les problèmes d'optimisation, nous considérons soit la réalisation d'une fonction à objectif unique (minimiser les coûts **ou** maximiser la rentabilité) soit une fonction à objectifs multiples qui recherche un compromis entre les avantages et les coûts (minimiser les coûts **et** maximiser les profits [Savic, 2002]. Dans notre étude, nous utilisons une fonction mono objectif qui minimise les coûts totaux liés à l'architecture du lac de données. En revanche, nous proposons une extension de notre modèle avec une fonction multi-objectif pour de futures recherches.

Définition 6.5 (***Les variables de décision***). *Les principales valeurs du processus d'optimisation qui sont considérées pour trouver leurs valeurs optimales en résolvant la fonction objectif sous les contraintes prédéfinies.*

Dans cette étude, le nombre de Jobs actifs, le nombre de tâches de Map et le nombre de tâches de Reduce ainsi que les demandes couvertes des utilisateurs sont les variables de décision envisagées.

Définition 6.6 (***Le Job actif***). *Un Job Mapreduce actif est un Job qui s'exécute complètement et devient désactivé (Job défectueux) en raison d'une panne de la machine.*

6.4.1 La formulation de modèle mathématique

Dans cette section, nous présentons la formulation du modèle mathématique du problème "localisation-allocation" pour le lac de données avec la différence que "localisation" consiste à trouver le nombre optimal de Job actif et "allocation" est lié à l'affectation du Job actif aux utilisateurs avec un temps d'exécution minimum. Étant donné que l'un des objectifs les plus importants de l'émergence d'un lac de données à l'ère du big data est de réduire le coût total de lancement et les performances du système de stockage de données, l'objectif du modèle est de minimiser le coût global de l'exécution du lac de données qui est entraîné par le réseau.

En se basant sur ces exigences, Nous intégrons maintenant ces deux décisions dans un modèle de programmation mathématique sous les hypothèses susmentionnées. Avant de présenter le modèle, introduisons la notation utilisée dans le modèle.

- Les **hypothèses** sont les suivantes :

- Les coûts de lancement du lac de données sont fixes et ne sont pas pris en compte dans ce modèle ;
- Le lac de données accueille les données par lots ;
- La couche de traitement du lac de données utilise la technologie MapReduce pour traiter les données ;
- Les coûts de stockage sont consideres comme le temps pour stocker les données ;
- Les coûts d'ingestion sont consideres comme le temps pour ingérer les données ;
- Les coûts d'accès sont consideres le temps pour analyser et accéder
- La demande des utilisateurs est déterministe ;
- Toutes les demandes des utilisateurs doivent être couvertes ;
- La qualité des données est garantie par la gouvernance des données ;
- La capacité est définie comme des limitations du CPU pour répondre aux demandes des utilisateurs ;
- Le Job actif est un Job qui s'exécute complètement ;
- Le Job peut être inactif en raison d'une panne de la machine ;
- Les coûts sont définis comme toutes les activités de performance du lac de données qui génèrent du temps supplémentaire pour exécuter les Jobs et répondre aux requêtes.

- Les **variables** associées sont :

I L'ensemble des utilisateurs

J L'ensemble des Jobs candidats de MapReduce

- Les **paramètres et notations** sont listés ci-dessous :

- Ca La capacité d'accès des utilisateurs
- D_i La quantité de la demande (requête) de l'utilisateur i ($\forall i \in I$)
- E Le coût d'ingestion
- S Le coût du stockage des données
- T Le coût du traitement des données qui contiennent :
  - MI_j La phase d'entrée de Map pour Job j
  - MO_j La phase de sortie de Map pour Job j
  - RI_j La phase d'entrée de Reduce pour Job j
  - RO_j La phase de sortie de Reduce de Job j

- Q Le coût de l'opération de requête et du processus analytique
- L Le coût de latence

- Les **Variables de décisions** sont définies comme :

$X_{ij} = la\ demande\ de\ l'utilisateur\ i\ est\ couverte\ par\ le\ job\ j$

$$Y_j = \begin{cases} 1 & si\ le\ job\ j\ est\ actif \\ 0 & sinon\ (\forall j \in J) \end{cases}$$

- Fonction objectif :

Généralement, les problèmes d'optimisation contiennent une fonction objectif (fonction mono-objectif) telle que minimiser les coûts et les risques ou maximiser la rentabilité et la qualité des services, ou plusieurs objectifs (fonction multi-objectif) contradictoires qui minimisent et maximisent les valeurs en même temps. Dans cette étude, nous abordons le problème d'optimisation avec une fonction objectif, cependant, nous proposerons un problème à objectifs multiples comme extension de notre travail. Comme nous l'avons indiqué, nous souhaitons optimiser l'architecture du lac de données à travers une fonction objectif qui minimise les coûts de type de consommation du temps du système. En effet, la fonction objectif contient les coûts d'ingestion, de stockage, de traitement (les coûts des phases d'entrée et de sortie de MapReduce), et les coûts de satisfaction de la requête. Dans ce cas, la fonction objectif contient :

$$\begin{aligned} &\sum_{i \in I} \sum_{j \in J} X_{i,j} D_i (L + Q + E + S) + \\ &\sum_{j \in J} Y_j (MI_j + MO_j + RI_j + RO_j) \end{aligned} \tag{6.13}$$

- Formulation du problème

A partir des hypothèses, des paramètres, des variables de décision, de la fonction objectif et des contraintes potentielles pour ce modèle, la modélisation mathématique est formalisée comme suit :

$$\begin{aligned} Min : &\sum_{i \in I} \sum_{j \in J} X_{i,j} D_i (L + Q + E + S) + \\ &\sum_{j \in J} Y_j (MI_j + MO_j + RI_j + RO_j) \end{aligned} \tag{6.14}$$

Subject to :

$$\sum_{i \in I} X_{i,j} = D_i \qquad \forall j \in J \tag{6.15}$$

$$\sum_{j \in J} X_{i,j} D_i \leq Ca \qquad \forall i \in I \tag{6.16}$$

$$X_{i,j} \leq y_j \qquad \forall j \in J \;\; i \in I \tag{6.17}$$

$$X_{i,j} \geq 0 \qquad \forall j \in J \;\; i \in I \tag{6.18}$$

$$y_j \in 0, 1 \qquad \forall j \in J \tag{6.19}$$

L'équation (6.14) minimise les coûts totaux du lac de données. L'équation (6.15) garantit que chaque requête utilisateur est entièrement attribuée par la Job active. L'équation (6.16) représente la limitation de capacité du lac de données. L'équation (6.17) indique que la demande de l'utilisateur ne peut être affectée à un Job que si ce Job est actif. Enfin, les équations (6.18) et (6.19) indiquent le type de variables de décision.

6.4.2 Solution

Le processus d'optimisation est un ensemble d'outils mathématiques et de stratégies de décision qui tentent de trouver la meilleure solution pour un problème actuel (pas seulement une solution) avec les conditions prédéfinies parmi les solutions potentielles. En général, les méthodes d'optimisation résolvent les problèmes de minimisation, de maximisation ou de minimisation-maximisation de fonctions objectifs pour trouver les solutions optimales en se concentrant sur les étapes qui génèrent l'optimum global au lieu de l'optimum local. En effet, les problèmes de problème d'optimisation sont modélisés par des algorithmes ou des approches qui atteignent les meilleures solutions telles que l'optimum global et évitent de tomber dans les pièges des solutions optimales locales. Les algorithmes d'optimisation sont catégorisés selon le nombre et le type de décisions variables, les contraintes, les méthodes de résolution [Haupt and Haupt, 2004]. Les problèmes d'optimisation continue dans lesquels les décisions variables sont continues, sont répertoriés dans deux sous-ensembles des méthodes d'optimisation linéaire et non linéaire et en outre, les méthodes non linéaires sont divisées en méthodes globales et méthodes locales. De manière générale, pour résoudre des problèmes d'optimisation, des algorithmes exacts ou heuristiques sont normalement utilisés. Les algorithmes exacts conviennent aux problèmes à petite échelle, mais pour les problèmes réels qui sont plus complexes, des algorithmes heuristiques ou méta-heuristiques sont utilisés. Les algorithmes métahusristiques utilisent des approches systémiques pour trouver la solution optimum globale et l'empêcher de se bloquer dans les optimums locaux. Les algorithmes méta-heuristiques sont les meilleures méthodes répétitives et évolutives qui atteignent les résultats pour les problèmes à grande échelle ou en classe NP-difficile [Talbi, 2009].

Dans le cadre du problème de prise de décision hybride, les algorithmes exacts sont mis à disposition pour résoudre les modèles mathématiques et obtenir les bonnes réponses optimales. Cependant, l'efficacité des méthodes exactes sera remise en question lorsque la taille du problème sera augmentée. Pour cette raison, pour ce genre de problème d'optimisation, les algorithmes

méta-heuristiques qui recherchent la meilleure réponse possible, pas nécessairement une réponse exacte, sont plus pratiques à mettre en œuvre. Pour cette raison, nous proposons deux méthodes heuristiques pour résoudre le modèle mathématique dans le chapitre suivant et comparer les fonctionnalités de l'architecture grâce à la structure optimisée proposée.

6.5 Résumé

Dans ce chapitre, nous avons vu que l'optimisation est un processus essentiel pour les systèmes, qu'il s'agisse de systèmes organisationnels ou de systèmes informatiques, afin d'augmenter la fonctionnalité et l'efficacité de manière constante et efficace. Les stratégies et les outils à utiliser pour l'optimisation sont énormes et dépendent de la structure et des objectifs principaux du système, néanmoins dans les cas général, on cherche à réduire et éliminer les coûts et à rentabiliser les systèmes grâce à des méthodes d'optimisation. Dans ce chapitre, nous proposons une méthode d'imitation pour prendre des décisions hybrides (stratégiques et tactiques) au sein de l'architecture du lac de données qui s'inspire des systèmes logistiques. Pour cette raison, nous utilisons un problème de localisation-allocation conjoint qui est utilisé pour concevoir des réseaux de chaîne d'approvisionnement afin de définir et de formuler un modèle d'optimisation mathématique pour le lac de données. Le modèle proposé cherche le nombre optimal de Jobs MapReduce actifs afin de minimiser les coûts totaux associés au temps d'exécution du lac de données en satisfaisant les requêtes générées par les utilisateurs.

Chapitre 7

Méthodologie de la solution et d'expérimentation

"L'homme qui apprend doit croire ; celui qui sait doit examiner."

– Roger Bacon *(1214-1294)*

7.1 Introduction

Les problèmes d'optimisation sont des problèmes omniprésents ou indispensables dans tous les domaines axés sur le développement et l'amélioration des performances. L'optimisation pourrait être obtenue à partir des stratégies managériales et de la modélisation mathématique de la recherche opérationnelle pour soutenir les processus de prise de décision [Savic, 2002]. Les problèmes et l'algorithme d'optimisation sont classés selon le nombre et les types de variables de décision, le nombre de fonctions objectifs, les modèles de programmation, les approches pour trouver la solution optimale et la limitation du problème (contraint et non contraint). La résolution de problèmes d'optimisation signifie qu'on recherche la meilleure valeur pour le problème défini dans un espace de solution faisable [Ross and Jayaraman, 2008]. Selon le type de problème d'optimisation, le modèle de programmation et la complexité temporelle de la résolution, il existe des méthodes exactes ou approximatives pour trouver la réponse optimale. Les méthodes exactes trouvent une réponse optimale exacte au problème comme la méthode Branch and Bound, tandis que les méthodes approximatives recherchent les meilleures solutions (solution quasi optimales[1]) parmi les réponses possibles (pas nécessairement une réponse exacte) telles que les méthodes heuristiques et méta-heuristiques [Thizy et al., 1985, Savic, 2002, Rohde and Vidal, 2020, Talbi, 2009]. Bien que les méthodes exactes puissent être en mesure de trouver la réponse exacte pour le modèle, mais à mesure que la taille du problème augmente, le problème appartiendra à la classe de problèmes NP-Complet ou Np-Difficile et il ne pourra pas être résolu en un temps raisonnable avec les méthodes exactes. Normalement, les problèmes d'optimisation de la vie réelle sont difficiles à résoudre avec les démarches classiques et pour cette raison, dans ces cas, des méthodes approximatives tels que heuristiques et méta-heuristiques sont utilisées pour résoudre le problème à grande taille avec des temps de calcul raisonnables [Haupt and Haupt, 2004, Rothlauf, 2011, Talbi, 2009]. La figure 7.1 présente la typologie des méthodes pour résoudre les problèmes d'optimisation selon le type de modèle et la complexité du temps de calcul.

Les méthodes méta-heuristiques sont un sous-ensemble de méthodes d'optimisation approximatives qui sont mises en œuvre pour résoudre le problème selon lequel sa complexité de temps et d'espace augmentera de manière exponentielle à mesure que la taille du problème serait grande. Selon [Talbi, 2009] :

" Les méta-heuristiques sont des méthodes de calcul de solutions optimales pour des problèmes difficiles qui préparent une solution "acceptable" dans un temps d'exécution raisonnable mais elles ne garantissent pas de manière optimale des solutions obtenues."

En général, les mécanismes des méthodes méta-heuristiques sont dérivés de phénomènes naturels tels que la loi de l'évolution, les concepts gloutons et la colonie de fourmis ou d'abeilles qui offrent des perspectives pour concevoir les algorithmes créatifs pour rechercher et trouver les

[1] Near-optimal solution

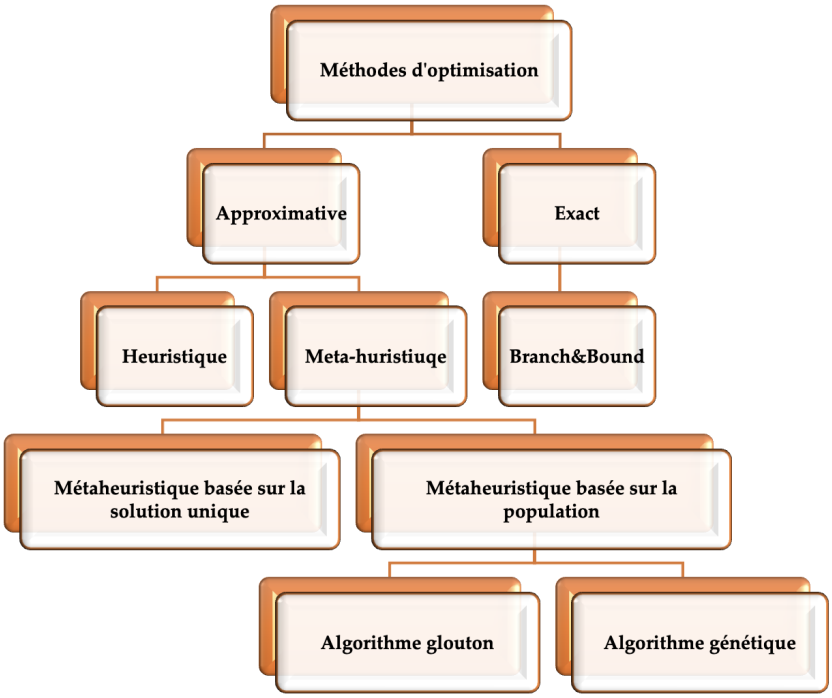


FIGURE 7.1 – Méthodes de résolution de modèles mathématiques d'optimisation

bonnes solutions pour l'optimisation des modèles complexe. Selon la figure 7.1 les algorithmes méta-heuristiques recherchent les réponses optimales dans manière itérative en se basant sur deux mécanismes principaux, la solution unique et la population. Les algorithmes basés sur une solution unique [2] recherchent la réponse optimale à partir d'une solution candidate initiale et à chaque itération, la réponse est évolué jusqu'à trouver la solution optimale, alors que dans les algorithmes basés sur la population [3], ces étapes sont effectuées sur une population candidate initiale tels que l'algorithme génétique et l'algorithme glouton.

La complexité temporelle et d'espace des solutions sont les sujets problématiques pour choisir les méthodes d'optimisation appropriées. Dans le modèle d'optimisation proposé pour le lac de données, lorsque les demandes des utilisateurs et le nombre des tâches des jobs sont augmentés, il y a un risque que le problème ne soit pas résolu avec les méthodes exactes et classiques. Puisque nous définissons la minimisation des coûts d'exécution comme fonction d'objectif pour le modèle d'optimisation de l'architecture du lac de données, le problème devient sensibilisé à tous les critères qui affectent la complexité temporelle du modèle. En remarquant que la technologie de MapReduce est employé pour la couche de traitement de données, *un processus de MapReduce avce M Mapper et N Reducer a besoin de $O(M+N)$ décisions d'ordonnancement et de $O(M*N)$ des états qui sont conservés en mémoire* [Dean and Ghemawat, 2008]. D'autre part selon [Peyravi and Moeini, 2020], le temps d'exécution des Jobs dépend de plusieurs critères tels que l'anatomie de MapReduce, la vitesse des fonctions Map et Reduce, la quantité de données à traiter pour chaque phase et l'état de réseau et est estimé selon la formule ci-après.

[2] Single Solution Metaheuristics
[3] Population–Based Metaheuristics

$$T_{Job} = T_{Map} + T_{Reduce} \tag{7.1}$$

Selon les informations effectuées sur les nombres de Map et Reduce dans MapReduce, les impacts de ces deux phases importantse sur la réalisation des Jobs, et l'équation 7.1, le nombre de Jobs est un élément important qui influence le temps total d'exécution du lac de données.

Comme décrit dans le chapitre 6, nous proposons un modèle mathématique basé sur le problème de localisation-allocation pour optimiser la structure du lac de données. Selon les études menées, les problèmes de décision conjointe qui mettent en évidence plusieurs niveaux de décision simultanément, appartenant normalement à la classe Np-Hard. Le problème de localisation-allocation (en tant que sous-ensemble de problèmes de décision conjointe) qui aborde la prise de décision stratégique et tactique dans le même temps, confronté à ces enjeux dans lesquels lorsque le nombre d'installations ou la demande des clients est augmenté, la taille du problème sera élevée. Par conséquent, trouver la réponse optimale avec des méthodes classiques et exactes est difficile en ce qui concerne les temps de calcul élevés pour ces problèmes. Pour cette raison, la recherche prouve que les problèmes de localisation-allocation hybrides à grande échelle appartiennent à la classe Np-Hard. [Cooper, 1963, Azarmand and Neishabouri, 2009]. De même, dans le modèle proposé pour le pipeline du lac de données, les Jobs de Mapreduce sont considérés comme les établissements dans les problèmes de localisation-allocation que nous envisageons pour prendre ses nombres optimaux. En effet, le modèle d'optimisation du lac de données pourrait appartenir à la classe NP-Hard lorsque le nombre de Jobs sera élevé et pour résoudre le modèle il faut des méthodes efficaces.

Étant donné que dans cette étude :

- Nous avons utilisé les bases des problèmes de localisation-allocation pour modéliser le modèle mathématique afin d'optimiser le lac de données ;
- Le lac de données accueille et traite des ensembles de données à grande échelle ;

Nous pouvons conclure que le modèle d'optimisation proposé pour le lac de données au chapitre 6 appartient à la classe Np-Hard en grande échelle. Pour cette raison, dans ce chapitre nous examinerons notre modèle avec les méthodes exactes et également avec les méthodes méta-heuristiques dans le cas de complexité de problème, pour trouver les valeurs optimales des décisions variables définies.

Dans ce chapitre nous allons nous appuyer sur :

- Les méthodes de résolution pour la modèle mathématique d'optimisation de lac de données (exacte et méta-heuristiques) ;
- Les résultats d'expérimentations obtenus.

Demande	Jobs actifs	Coût (seconde)
1	1	600
2	1	1100
3	1	1700
4	3	2200
5	4	2600
6	4	2900
7	5	3200
8	3	3600
9	3	4100
10	4	4800

TABLE 7.1 – Valeurs optimales du problème d'optimisation avec l'algorithme exact

7.2 Méthodes de résolution

7.2.1 Méthodes exactes

L'une des méthodes courantes pour résoudre les problèmes d'optimisation sous la forme d'un modèle mathématique est la méthode exacte. Les méthodes exactes sont basées sur l'algorithme **Branch and Bound** et sont catégorisées selon les types de fonction objectifs, de contraintes et de variables de décision, telles que *Leanier Programming(LP)*, *Non Leanier Programming(NLP)*, *Quadratic program (QP)*, *Mixed Integer Programming (MIP)*, et ses extensions tels que *Mixed Integer Non Leanier Programming (MINLP)*. Pour résoudre le modèle d'optimisation proposé avec les méthodes exactes, nous avons choisi le logiciel GAMS qui utilise les différents solveurs tels que GAMS/Cplex avec des capacités de modélisation de haut niveau selon les types de modèles mathématiques définis pour résoudre rapidement des problèmes d'optimisation LP, NLP, MIP, et ses extensions, [4].

Pour la première étape, nous choisissons une méthode exacte pour examiner notre modèle.

Notre but est :

" Obtenir les valeurs optimales pour les deux variables de décisions qui sont le nombre optimal des Jobs du Mapreduce Y_j et les demandes couvertes X_{ij} en minimisant le coût total ainsi qu' en effectuant une analyse de sensibilité des paramètres du modèle ".

Dans notre démarche, nous avons utilisé une méthode exacte de Mixed Integer Programming (MIP) dans GAMS pour obtenir les résultats optimaux du modèle.

Pour examiner le problème d'optimisation, nous utilisons les données hypothétiques ainsi que les résultats d'études de [Peyravi and Moeini, 2020] pour les paramètres du modèle mathématique. Le modèle est codé dans un environnement logiciel GAMS 34.3.0 et les expérimentations

[4] https://www.gams.com/35/docs/S_CPLEX.html

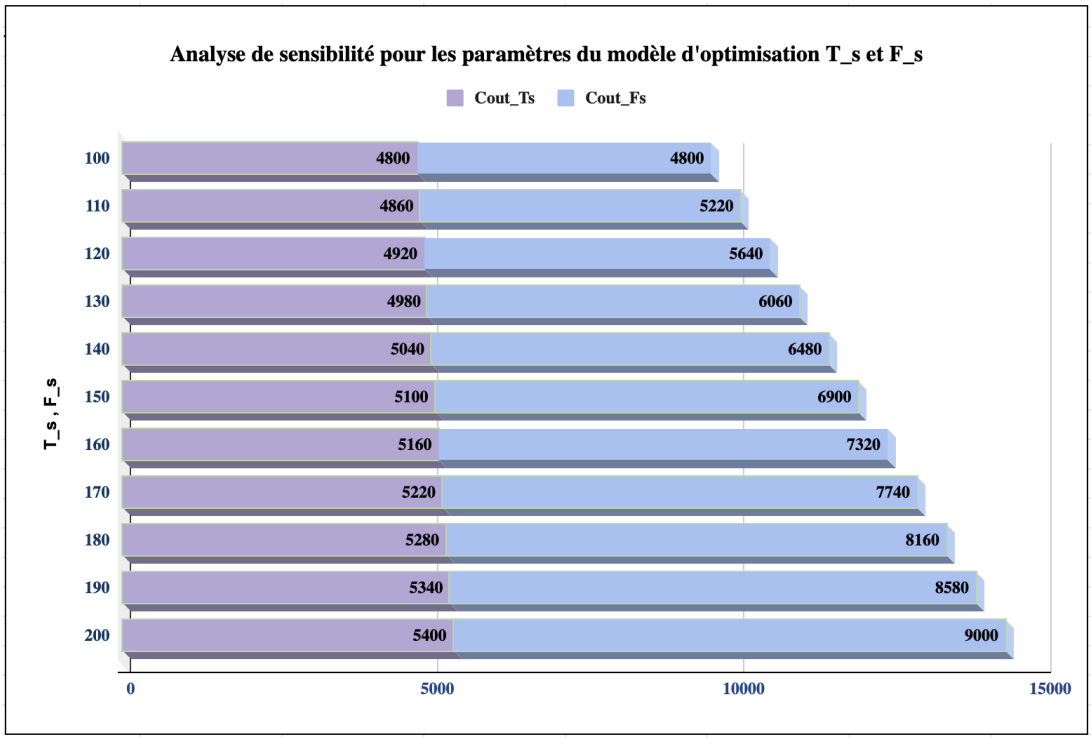


FIGURE 7.2 – Analyse de sensibilité pour les paramètres du modèle d'optimisation $T_{(s)}$ et $F_{(s)}$

sont réalisées dans un ordinateur MacBook Pro M1 et 8 Go de mémoire, pour estimer les fonctions de réponse. Nous considérons 6 Jobs et 10 demandes et nous utilisons IBM ILOG CPLEX 34.3.0 comme solveurs pour examiner le modèle. Les résultats sont obtenus comme 7.1. De plus, pour analyser la sensibilité des fonctions de réponse en fonction des paramètres du modèle tels que les coûts d'ingestion, de stockage, d'accès et de traitement, nous avons effectué une analyse de sensibilité. Dans cette approche, les paramètres du problème sont maintenus constants, et en changeant un paramètre, la quantité de changement dans la fonction objectif par rapport au paramètre variable est vérifiée. Pour cette raison, nous analysons les sensibilités du montant des fonctions objectives (coûts) selon les deux paramètres, on fois pour T (le coût des couches de traitement) et et d'autre fois pour l'ensemble des coûts des autres couches F (les coûts d'ingestion, de stockage, et accès) selon le résultats de table 7.1 pour 10 demandes et 4 jobs actifs.

Selon le tableau 7.2 et la figure 7.2, nous constatons que les coûts d'ingestion, de stokage et d'accès($F_{(s)}$) sont un ensemble de coûts importants qui pourraient influencer le coût total du lac de données plus que le coût de traitement $T_{(s)}$ de données. Compte tenu de ces résultats, on peut conclure que la modélisation technique de ces trois couches influentes dans l'architecture du lac de données pourrait améliorer les fonctionnalités et les niveaux de service du lac de données de manière efficace.

Bien que les algorithmes exacts soient la méthode appropriée et efficace pour résoudre ce problème d'optimisation, l'efficacité des modèles pour trouver la solution optimale en temps raisonnable se dégraderont à mesure que le nombre de demandes ou de Jobs augmente. Pour cette raison, nous proposons deux méthodes méta-heuristiques pour résoudre le modèle dans le cas où l'échelle du problème défini sera grande et les modèles exacts seront inefficaces.

$T_{(s)}$ (F est constant)	$Cout_{(s)}$	$F_{(s)}$ (T est constant)	$Cout_{(s)}$
100	4800	100	4800
110	4860	110	5220
120	4920	120	5640
130	4980	130	6060
140	5040	140	6480
150	5100	150	6900
160	5160	160	7320
170	5220	170	7740
180	5280	180	8160
190	5340	190	8580
200	5400	200	9000

TABLE 7.2 – Résultats d'analyse de sensibilité pour les paramètres du modèle d'optimisation $T_{(s)}$ et $F_{(s)}$

7.2.2 Algorithme génétique

Une autre méthode utilisée pour résoudre le modèle d'optimisation est l'algorithme méta-heuristique génétique. L'algorithme génétique est un membre de la famille des algorithmes évolutionnaires qui est basé sur des populations de solutions. Cet algorithme imite la loi de sélection naturelle de Darwin telle que la sélection, la reproduction et la mutation, pour trouver des solutions optimales de manière itérative [Abdelmaguid and Dessouky, 2006]. Le fonctionnement de l'algorithme est plus compliqué et, bien évidemment, plus fiable que les algorithmes exacts et glouton pour trouver des solutions globales optimales. Il part d'une population aléatoire d'individus qui sont les solutions possibles au problème, et il fait évoluer ces solutions initiales à travers des opérateurs naturels évolués comme la combinaison et la mutation jusqu'à l'obtention d'une génération finale qui optimise le problème. L'algorithme génétique simule l'approche de la loi naturelle pour résoudre le modèle d'optimisation avec les étapes suivantes et nous expliquerons comment nous utilisons ces étapes dans notre algorithme selon le modèle défini du chapitre 6 :

Algorithm 1: Génétique

```
1  Begin;
2  Initial population of solutions;
3  Repeat;
4  Compute the fitness value of each chromosome(solution) in initial population;
5  Select the fittest solution according to the fitness value ;
6  Apply the crossover operator ;
7  Apply the mutation operator ;
8  Apply the mutation operator ;
9  Compute the fitness value of new generated chromosome(solution)
10 Until Stop condition;
11 End
```

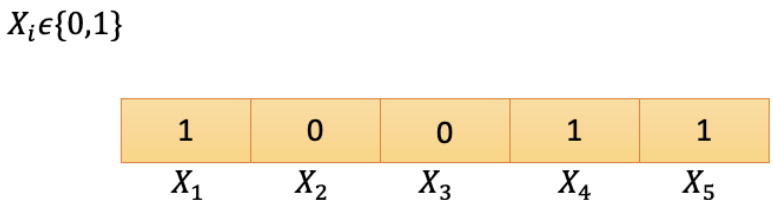


FIGURE 7.3 – Exemple de chromosome de variable de décision binaire

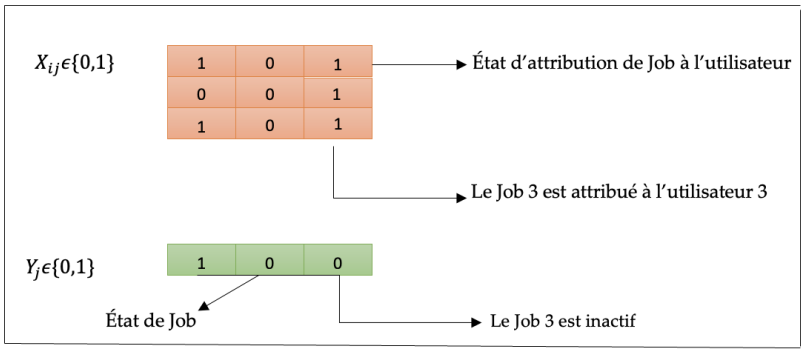


FIGURE 7.4 – Représentation de chromosome du modèle d'optimisation du lac de données

- ***Population initiale***

  L'algorithme commence par une population initiale de solutions de taille fixe qui est générée aléatoirement. Chaque individu de cette population représente une solution possible au problème est présenté par un ensemble de gènes appelé un chromosome. Les chromosomes peuvent être présentés sous plusieurs formes selon les conditions du problème et le type de variables de décision. Par exemple, un chromosome de variable de décision binaire est présent sous la forme [Pasandideh et al., 2013] :

  Dans notre modèle, nous avons défini deux variables de décision y_j et $x_{i,j}$. On considère y_j le nombre de Jobs actifs qui est défini comme une matrice binaire 1 × J où les colonnes représentent le nombre de jobs potentiels et le gène sur ce chromosome accepte les valeurs 0 lorsque le Job est inactif ou 1 lorsque le job c'est actif. De plus, $x_{i,j}$ est considéré comme la demande couverte par les Jobs, on le représente avec une matrice de dimensions I×J où I est le nombre de lignes de la matrice qui représente le nombre d'utilisateurs J est le nombre de colonnes de la matrice qui représente le nombre de Jobs actives dans le lac de données. Chacun des gènes de ce chromosome accepte les valeur entiers (integer values). La figure 7.4 montre la représentation des chromosomes des variables de décision de modèle.

- ***Évaluation d'aptitude des solutions*** [5]

  L'aptitude de chaque population de chromosomes (solution possible) est évaluée selon les critères de compétence qui sont normalement la fonction objective du problème d'optimisation. Dans notre étude, nous avons défini la fonction objectif comme une minimisation des coûts de l'architecture du lac de données, de sorte que les solutions qui ont les va-

[5] Fitness evaluation

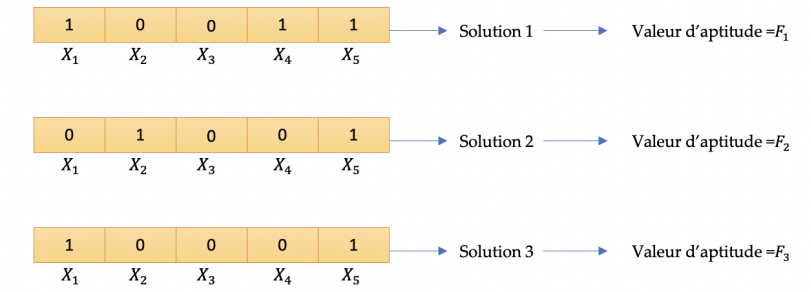


FIGURE 7.5 – Évaluation d'aptitude des solutions

leurs minimales de fonction objectif que les autres solutions sont plus susceptibles d'être sélectionnées en tant que parents pour les étapes suivantes.

- ***Sélection***

  Les solutions les plus adaptées sont classées et sélectionnées pour reproduire la progéniture [6] des nouvelles générations. Différentes méthodes sont utilisées pour sélectionner des parents plus appropriés, mais une approche actuelle pour cette étape est la sélection aléatoire de solutions plus appropriées qui sont classées selon ses valeurs de fonction objectif.

- ***Opérateur de combinaison*** [7]

  Les solutions adaptées sont sélectionnées comme parents pour produire les nouvelles progénitures à travers de opérateur de combinaison et selon le taux de combinaison prédéfinis. Il existe plusieurs méthodes pour opérer la combinaison des parents mais l'une des plus utiles est l'intersection en un point où un point de croisement est choisi aléatoirement pour chaque parent et ils sont combinés (ils échangent leurs gènes) à ce point [Mathew, 2012].

- ***Opérateur de mutation*** [8]

  L'opérateur de mutation, imitant une fonctionnalité présente dans la nature, est effectué sur les différents gènes de chromosomes pour trouver et créer les solutions plus uniques et efficaces qui ne sont pas générées par l'opérateur de combinaison. Cet opérateur choisit un chromosome de population et change aléatoirement les gènes de cette chromosome selon le taux prédéfini de mutation [Eiben et al., 2003].

- ***Évaluation de l'aptitude des nouvelles générations et remplacement***

  L'aptitude des nouvelles générations est évaluée selon la fonction objectif et les solutions les plus aptes seront remplacées par les solutions les moins adaptées dans la population.

- ***Terminus***

  L'algorithme se termine lorsqu'il atteint les critères de terminaison. Différentes conditions peuvent être définies pour arrêter l'algorithme, par exemple le maximum d'itérations, le montant spécifique pour la fonction objectif, le temps d'exécution prédéfini ou le nombre de générations. Dans cette étude, nous considérons l'itération maximale comme une condition de terminaison.

[6] Offspring
[7] Combination/Crossover operator
[8] Mutation operator

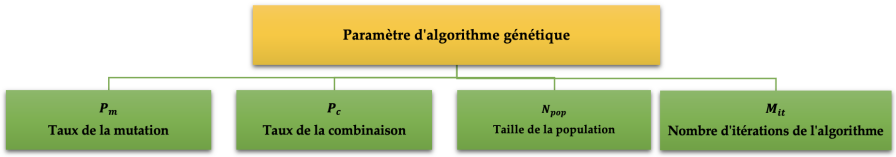


FIGURE 7.6 – Paramètres essentiels de l'algorithme génétique

Afin de mettre en œuvre un algorithme génétique efficace qui produit les nouvelles générations susceptibles de trouver les solutions optimales, il est nécessaire que les paramètres de l'algorithme soient bien déterminés. Les paramètres de l'algorithme génétique sont liés au fonctionnement des procédures de sélection, de reproduction et de terminaison de l'algorithme qui pourraient être classés comme la figure 7.6 tels que le taux de la mutation (P_m), le taux de la combinaison(P_c), la taille de la population(N_{pop}), et le nombre d'itération d'algorithme(M_{it}). Le paramétrage est un processus décisionnel pour lancer l'algorithme qui pourrait affecter les résultats obtenus et doit être mis en œuvre de manière précise [Talbi, 2009, Pasandideh et al., 2013].

Dans notre deuxième méthode heuristique, nous utilisons un algorithme génétique pour résoudre le modèle d'optimisation.

Notre but est :

" Obtenir les valeurs optimales pour les deux variables de décisions qui sont le nombre optimal des Jobs du Mapreduce y_j et les demandes couvertes X_{ij} ".

Notre demarche est :

" Nous avons implémenté un algorithme génétique défini en Python et utilisons la méthode de conception d'expérimentation RSM (Response Surface Methodology) en Minitab pour régler les paramètres principaux d'algorithme génétique tels que N_{pop}, P_c, M_{it} afin de réaliser un algorithme efficace."

Dans la section expérimentale, dans la première étape, nous présentons une méthode pour ajuster les paramètres de l'algorithme génétique, puis dans la deuxième étape, nous relançons l'algorithme génétique avec les paramètres déterminés afin d'obtenir un algorithme efficace qui trouve les meilleures réponses pour le variables de décision prédéfinies du modèle mathématique d'optimisation du lac de données.

Résultats d'expérimentation d'algorithme génétique

Dans cette section, nous implémentons un algorithme génétique qui est codé en `Python 3.9.6` sur ordinateur MacBook Pro M1 basé sur notre modèle heuristique proposé pour résoudre et examiner notre modèle mathématique d'architecture de lac de données. Premièrement, nous utilisons des approches statistiques pour déterminer la valeur optimale des paramètres de l'algorithme génétique et vérifier les performances de l'algorithme génétique dans la résolution du

Itération	$0.10 < P_c < 0.45$	$10 < N_{pop} < 40$	$10 < M_{it} < 40$	Coût (Function objective)
1	0.275	25	27	3630
2	0.100	10	10	4320
3	0.275	25	27	3510
4	0.100	10	45	4955
5	0.275	25	27	3510
6	0.100	40	45	3630
7	0.275	25	45	3510
8	0.275	10	27	3950
9	0.100	25	27	3795
10	0.275	25	27	3630
11	0.275	25	27	3510
12	0.100	40	10	3830
13	0.450	40	45	3510
14	0.275	40	27	3510
15	0.450	10	10	3950
16	0.275	25	10	3510
17	0.450	40	10	3635
18	0.450	10	45	3510
19	0.450	25	27	3510
20	0.275	25	27	3630

TABLE 7.3 – Les résultats RSM pour ajuster les paramètres de l'algorithme génétique

modèle mathématique. De plus, nous évaluons l'efficacité de l'algorithme génétique pour résoudre le problème d'optimisation de l'architecture des lacs de données. Pour ajuster les paramètres de l'algorithme génétique comme N_{pop}, P_c, M_{it}, nous utilisons la RSM [9] qui est une méthode de conception expérimentale. RSM est considéré comme un outil efficace pour identifier les relations entre les variables indépendantes et dépendantes dans les problèmes d'optimisation grâce à l'équation de régression et l'analyse de test de la variance ANOVA pour tester les variables pertinentes en très peu de temps [Said and Amin, 2015]. Cette méthode garantit de trouver les variables qui affectent les résultats du modèle d'optimisation afin de se concentrer sur ces variables pour obtenir les résultats les plus fiables. Pour cette raison, nous lançons RSM pour ajuster les paramètres de l'algorithme afin de trouver les relations et les effets de ces paramètres sur la quantité de fonction objectif (le résultat de l'algorithme génétique). Le tableau 7.3 représente les valeurs du plan expérimental pour les paramètres de l'algorithme génétique en tant que variables explicatives (variables indépendantes) et les résultats de ses effets sur la fonction objectif (variable dépendante) qui sont exécutés dans Minitab. Dans ce tableau, la colonne **Coût** est supposée comme variable de réponse qui est la sortie de l'algorithme génétique.

Sur la base des résultats de RMS, l'équation 7.2 montre la régression qui explique la relation entre les variables explicatives et la variable de réponse et le tableau 7.5 représente les résultats du test d'analyse de variance pour les paramètres de l'algorithme.

$$\begin{aligned} Cout = 5349 - 4600P_c - 78.9N_{pop} + 16.7M_{it} + \\ 4653P_c * P_c + 0.978N_{pop} * N_{pop} + 0.001M_{it} * M_{it} + \\ 71.4P_c * N_{pop} - 41.0P_c * M_{it} - 0.249N_{pop} * M_{it} \end{aligned} \tag{7.2}$$

Afin d'obtenir les meilleures valeurs des paramètres importants analysés, nous allons résoudre la régression de l'équation 7.2 avec une méthode de résolution exacte de modèles d'optimisation qui s'appelle GAMS avec solveur NLP afin de déterminer le valeurs optimaux de paramètres d'algorithme génétique. Le résultat de ce processus est présenté dans le tableau 7.4.

[9] Response Surface Methodology

P_c	N_{pop}	M_{it}
0.45	29	45

TABLE 7.4 – La valeur optimale des paramètres d'algorithme génétique

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Model	9	2218751	246528	10.93	0.000
Linear	3	1250744	416915	18.49	0.000
Pc	1	586673	586673	26.02	0.000
Npop	1	662381	662381	29.38	0.000
Imax	1	1690	1690	0.07	0.790
Square	3	532343	177448	7.87	0.005
Pc*Pc	1	55842	55842	2.48	0.147
Npop*Npop	1	133100	133100	5.90	0.035
Imax*Imax	1	0	0	0.00	0.997
2-Way Interaction	3	441718	147239	6.53	0.010
Pc*Npop	1	281250	281250	12.47	0.005
Pc*Imax	1	126398	126398	5.61	0.039
Npop*Imax	1	34070	34070	1.51	0.247
Error	10	225472	22547	-	-
Lack-of-Fit	5	203872	40774	9.44	0.014
Pure Error	5	21600	4320	-	-
Total	19	2444224	-	-	-

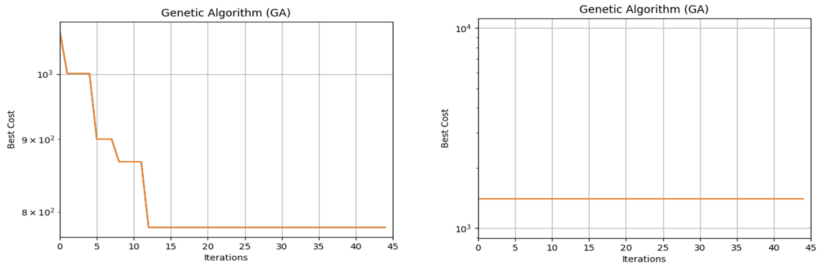
TABLE 7.5 – Résultats du test d'analyse de variance pour paramètres d'algorithme

Pour l'étape suivante, nous exécutons l'algorithme génétique sur la base des résultats optimaux obtenus pour les paramètres de l'algorithme en termes d'évolution des performances de cet algorithme qui sont présentés dans le tableau 7.4. Par conséquent, nous utilisons la méthode expérimentale (RSM) avec le logiciel Minitab mais cette fois, pour générer des valeurs aléatoires pour les variables *Job* et *Demande*. Dans cette recherche, pour éliminer les effets aléatoires et l'incertitude dans la sortie des algorithmes, chaque test a été répété 3 fois. Puisque nous ajustons le nombre maximum d'itérations de l'algorithme génétique à 45, la colonne *itération* représente le nombre d'itérations de l'algorithme pour atteindre les premiers résultats minimum dans chaque implémentation d'algorithme. Le tableau 7.6 et la figure 7.7 montrent les résultats obtenus.

D'après les résultats du tableau 7.6, dans les conditions définies pour ce problème, le coût minimum de l'architecture du lac de données est égal à 780 et il est obtenu avec 6 jobs et 2 demandes couvertes après les itérations initiales de l'algorithme (9/45, 4/45, 11/45). D'autre part, l'algorithme atteint le montant minimum de coût d'architecture de lac de données dans la première itération de chaque implémentation de l'algorithme génétique (1/45) et reste constant.

Itération	$2 < Job < 6$	$2 < Demand < 10$	Coût (Function objective)	$M_{it} = 45$
1	6	2	780	9/45
2	4	6	3360	44/45
3	4	6	3360	45/45
4	2	10	9122	26/45
5	2	2	1400	1/45
6	6	10	5671.25	45/45
7	4	10	6997	44/45
8	2	6	4190	24/45
9	4	6	3360	40/45
10	4	6	3400	34/45
11	6	6	3560	43/45
12	4	2	1050	7/45
13	4	6	3360	37/45
1	4	6	3360	39/45
2	6	10	5702.75	39/45
3	4	6	3400	43/45
4	6	2	780	4/45
5	4	2	1050	4/45
6	4	6	3400	36/45
7	2	6	4190	29/45
8	2	10	8825	41/45
9	4	6	3360	36/45
10	2	2	1400	1/45
11	6	6	3426.6	40/45
12	4	10	7085	43/45
13	4	6	3360	28/45
1	2	10	8900	39/45
2	4	6	3360	33/45
3	4	6	3360	26/45
4	2	6	4190	12/45
5	2	2	1400	1/45
6	6	6	3453.3	44/45
7	4	6	3360	34/45
8	6	10	6145	45/45
9	4	6	3396.6	38/45
10	4	2	1050	4/45
11	4	6	3360	41/45
12	4	10	7013	44/45
13	6	2	780	11/45

TABLE 7.6 – Résultats de l'exécution de l'algorithme génétique



a) Imax=45 Job =6 Demand= 2 b) Imax=45 Job =2 Demand=2

FIGURE 7.7 – Les tracés d'exécution de l'algorithme génétique

7.2.3 Algorithme glouton

Un algorithme glouton est un algorithme qui est souvent utilisé pour résoudre les problèmes d'optimisation [Chen, 2008, Curtis, 2003, Talbi, 2009]. A chaque itération, la solution candidate est construite afin de trouver la solution optimale locale (la meilleure réponse disponible et faisable à ce moment-là) en considérant l'obtention de la solution optimale globale qui satisfait la fonction objectif (la minimisation ou la maximisation des valeurs). L'algorithme glouton est basé sur la méthode consistant à trouver les meilleures valeurs pour les variables de décision à chaque phase de manière gloutonne, jusqu'à trouver la réponse compatible avec les critères définis (la fonction objective et les contraintes) sans revenir sur les décisions précédentes [Curtis, 2003].

Algorithm 2: Gloutonne

```
1 Begin;
2 Initial empty solution;
3 Repeat;
4 Local heuristic research of all candidates in list C;
5 Compute the fitness value of each choice from list C;
6 Test the feasibility of the solution;
7 Update list C according to the feasible solution ;
8 Until global optimal solution found;
9 End
```

Sur la base de cette définition, le mécanisme de l'algorithme glouton est construit sur une recherche heuristique locale et des choix séquentiels en fonction des candidats existants (solutions partielles) pour trouver la solution globale optimale. Les approches montrent qu'à chaque itération de l'algorithme, la faisabilité de la solution locale est vérifiée en fonction de la valeur de fitness (on peut définir la fonction objectif comme fonction de fitness) jusqu'à obtenir la solution optimale globale actuelle et disponible. Bien que cet algorithme soit un algorithme simple à mettre en œuvre et efficace pour résoudre des modèles d'optimisation, il peut être faible de trouver des solutions optimales pour les problèmes de classe NP-difficile sous la forme de base qui est affiché dans la figure 7.8. Pour cette raison, des extensions de cet algorithme sont développées afin d'augmenter la fiabilité de la procédure pour trouver la solution optimale globale pour les problèmes à grande échelle.

Sur la base de cette exigence, nous développons un algorithme glouton pour résoudre le modèle d'optimisation du lac de données et trouver les valeurs optimales des variables de décision prédéfinies. Dans cet algorithme, notre but est de

" déterminer les jobs actifs j les moins chers et les plus rapides pour satisfaire la demande de l'utilisateur i".

Notre démarche est basée sur l'implémentation de notre méthode heuristique en Python avec un échantillon de 4 jobs choisis aléatoirement. Dans cette approche, l'efficacité est définie comme la vitesse du Job actif pour répondre à la demande.

On suppose que la demande de l'utilisateur (i) se produit de manière aléatoire entre un intervalle de temps $(t, t+1)$ et qu'il est automatiquement associé à une *aptitude de job* $(X_{i,j})$.

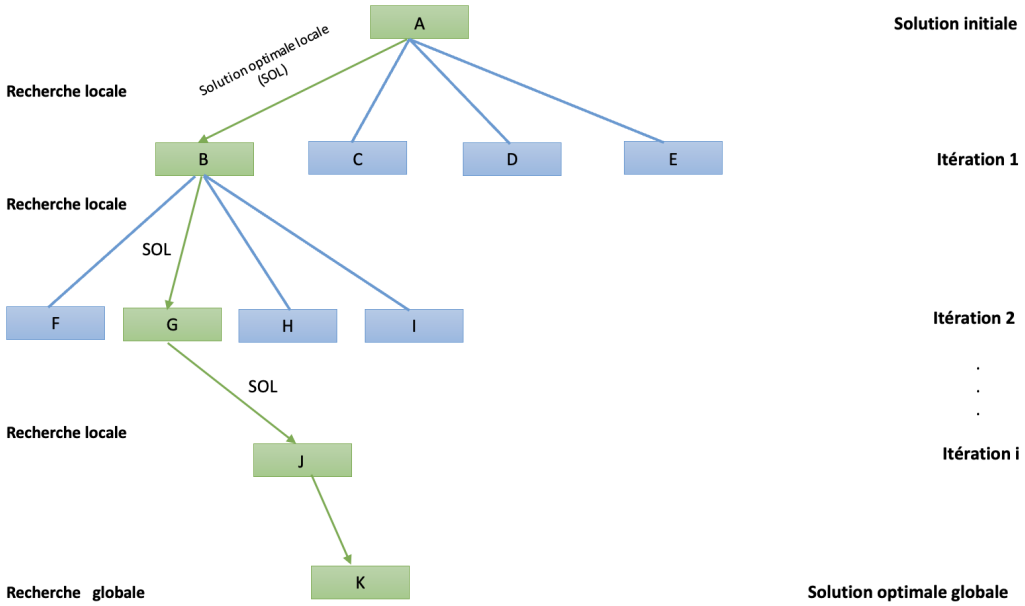


FIGURE 7.8 – Algorithme glouton de référence [Chen, 2008]

Soit $\tau_j(t)$, la vitesse chronologique associée au job j au temps t et, chaque agent au temps t choisit le job qui sera au temps $t+1$. Nous définissons pour tous les choix créés par q l'aptitude des agents dans l'intervalle $(t, t+1)$ par une itération unique de notre algorithme heuristique. Par conséquent, chaque n itération de l'algorithme, chaque agent achève son choix et chaque job démarre instantanément et, à ce stade, la vitesse chronologique de job est mise à jour selon l'équation ((7.3)).

$$\tau_j(t+n) = \tau_j(t) + \varrho \tag{7.3}$$

où ϱ est un facteur dérivé du rapport entre le temps nécessaire pour terminer le job et le temps entre l'intervalle $(t, t+n)$:

$$\varrho = \frac{\Delta t_j}{\Delta(t, t+n)} \tag{7.4}$$

Afin de définir la probabilité de $p^k_{i,j}$ que le $k$ème agent choisisse le job j pour la demande d'utilisateur i, nous définissons *overhead* $\eta_j = 1/c_j$ où c_j est le coût requis pour exécuter le job j). Par conséquent, $p^k_{i,j}(t)$ est décrit dans l'équation (7.5) :

$$p^k_{i,j}(t) = \begin{cases} \frac{\tau_j(t)\cdot\eta_j}{\sum_{k\in allowed_k} \tau_k(t)\cdot\eta_k} & \text{si } j \in allowed_k \\ 0 & sinon \end{cases} \tag{7.5}$$

où $allowed_k$ est l'ensemble de jobs *'actifs'* dont $Y_j = 1$

La probabilité de $p^k_{i,j}(t)$ est un compromis entre *overhead* η_j (qui indique que le job le moins cher doit être sélectionné avec la probabilité la plus élevées) et la vitesse chronologique au temps t $\tau_j(t)$ (qui indique que le job le plus rapide jusqu'à temps t doit être choisi avec la probabilité la plus élevée).

Résultats d'expérimentation

Nous avons implementé notre méthode heuristique en `Python`[10] avec un échantillon des 4 jobs choisis de manière aléatoire. L'efficacité est définie comme étant la vitesse du job actif pour répondre à la requête . La Figure 7.9 représente le résultat de ces expérimentations (mesures de coûts et d'efficacité) .

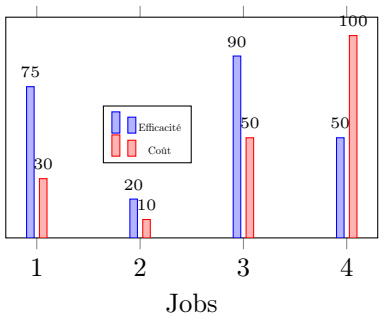


FIGURE 7.9 – Valeurs synthétiques de coût et d'efficacité pour 4 échantillons de jobs

Nous comparons l'allocation de demande à un job en utilisant un modèle équitable et en utilisant le modèle heuristique proposé, comme les Figures 7.10(a) et 7.10(b) le montrent. Le modèle heuristique alloue plus de demandes aux jobs à haute efficacité et à des coûts relativement bas, comme le montre la Figure 7.10(b).

Nous enregistrons le temps d'exécution d'allocation du modèle heuristique (ligne continue grise) et le comparons avec le temps d'exécution d'allocation de modèle équitable (ligne pointillée bleue) dans la Figure 7.11 (a). Nous observons que le modèle heuristique a l'efficacité d'allocation la plus élevée. Nous observons également que les coûts calculés cumulés du modèle heuristique sont inférieurs à ceux du modèle équitable comme le montre la Figure 7.11(b).

[10] https://github.com/owuordickson/dl_opt

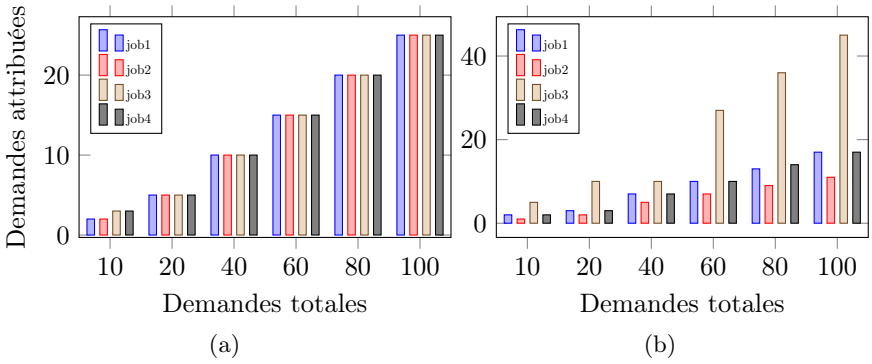


FIGURE 7.10 – Diagrammes à barres (a) Les demandes partagées de manière égale par les jobs disponibles et (b) Les demandes partagées sur la base du modèle heuristique proposé.

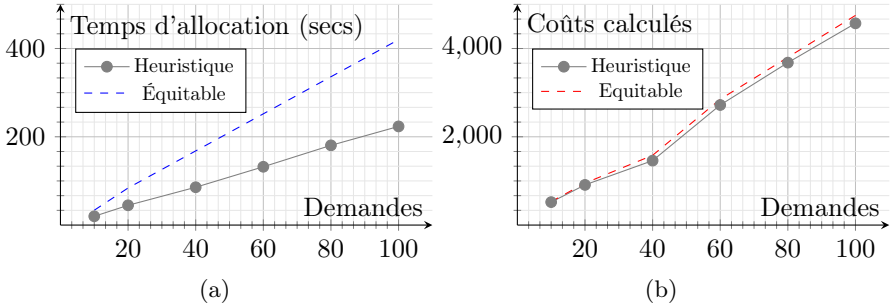


FIGURE 7.11 – (a) Diagramme de tracé de temps d'exécution d'allocation par rapport au nombre de demandes (b) diagramme de tracé de coûts calculés cumulés par rapport au nombre de jobs attribués.

7.3 Résumé

Dans ce chapitre, nous avons vu que parmi les nombreuses approches pour résoudre le modèle mathématique des problèmes d'optimisation, les algorithmes exacts et méta-heuristiques sont plus pratiques grâce à leurs approches efficaces ou évolutionnistes. Dans ce chapitre, nous avons profité de ces deux algorithmes connus pour trouver les valeurs optimales des variables de décision définies pour le modèle d'optimisation de l'architecture du lac de données sur la base des concepts logistiques. Les résultats de toutes les méthodes de résolution montrent que la fonctionnalité et les coûts de chaque couche d'ALLD pourraient affecter de manière significative les coûts totaux du lac de données, tout comme la fonctionnalité des membres du système logistique qui influence la rentabilité totale du système. Grâce à la structure d'ALLD et aux problèmes hybrides de location-allocation, nous avons réussi à définir un modèle mathématique et le résoudre avec différentes méthodes afin d'analyser les impacts de chaque couche du lac de données pour optimiser la performance globale. Compte tenu des résultats obtenus, il est suggéré que la modélisation technique efficace et la sélection de la technologie pertinente de chaque couche pourraient développer les niveaux de service des utilisateurs en minimisant les temps d'exécution du lac de données de manière optimisée.

Chapitre 8

Conclusion et perspectives

"La fin justifie les moyens. Mais qu'est-ce qui justifiera la fin ?"

– Albert Camus 1913 - 1960

8.1 Conclusion

Le lac de données est une nouvelle révolution des systèmes de stockage et de gestion des données qui devient un élément indispensable des systèmes d'information pour faciliter et accélérer le processus de prise de décision en valorisant les données brutes et hétérogènes. L'émergence du concept de lac de données dans l'environnement du Big data oriente les études et recherches scientifiques pour l'organisation, l'architecture et la plateforme performante pour mettre en œuvre un système de stockage centralisé, la gestion de la traçabilité et de l'identification des données et la gouvernance des données multi-structurées. Au regard de la littérature relative à la philosophie des systèmes hétérogènes de stockage de données, les stratégies managériales et les méthodes de conception de l'infrastructure optimisée restent toujours les enjeux essentiels à prendre en compte. Ces besoins nous conduisent à des méthodologies innovantes pour canaliser la structure du lac de données et optimiser sa fonctionnalité globale.

Selon la philosophie d'émergence du lac de données, la gestion de données massives, la fourniture de données FAIR, la qualité et la sécurité des données brutes et non normalisées sont les objectifs principaux de ce référentiel de données post-moderne. Sur la base de ces objectifs, l'urbanisation de l'architecture du lac de données et d'autre part la gouvernance et la gestion de la qualité et de la sécurité des données sont mises en évidence. Pour cette raison, modéliser une structure efficace avec les technologies et la stratégie appropriées qui améliore la qualité de service en minimisant le temps d'exécution, en assurant la traçabilité, la sécurité et la véracité des données, et en augmentant la rentabilité pour les organisations bénéficiaires, est une question à résoudre.

Prenant en compte la revue de la littérature liée aux problématiques du lac de données, nous avons retenu dans cette étude les trois scénarios pour adresser aux enjeux essentiels pour la gestion et l'optimisation du lac de données :

- ***Contribution 1 :*** *Une méthode analogique pour comparer les trois structures systémiques : le lac de données en tant que système de gestion des données, la chaîne d'approvisionnement en tant que système logistique, et l'écosystème de lac naturel. En s'appuyant sur ce scénario, on tire parti des stratégies et des concepts essentiels des systèmes logistiques pour la gestion, la gouvernance et en particulier la conception et l'optimisation de l'architecture du lac de données.*

- ***Contribution 2 :*** *Une méthode évolutionniste basée sur les principes des méthodes MERISE, le cadre DALF et les concepts de systèmes logistiques pour modéliser l'architecture logistique d'un lac de données appelé ALLD. Grâce à ce scénario, nous proposons une architecture logistique pour un lac de données à travers 4 phases de modélisation conceptuelle, logique, physique et optimale.*

- ***Contribution 3 :*** *Une méthode mimétique pour définir un modèle mathématique pour l'optimisation de l'ALLD. En utilisant ce scénario, on tire parti de l'une des stratégies pratiques de gestion et de conception de réseaux de chaîne d'approvisionnement appelée problème de prise de décision hybride (localisation-allocation) pour optimiser le lac de données de manière créative.*

Avec ces trois contributions, nous avons montré que les approches et les méthodes interdisciplinaires sont les bonnes solutions pour orienter la recherche sur les lacs de données vers une vision innovante d'optimisation de la performance des lacs de données. Nous avons également montré que seule l'architecture conceptuelle d'un lac de données n'est pas suffisante pour améliorer les performances d'un lac de données, nous avons plutôt besoin de niveaux plus élevés et plus profonds de conception structurelle et de programmes de la gestion pour développer des fonctionnalités. La conception d'ALLD ainsi que les approches mimiques pour optimiser cette architecture sont les points forts de cette étude qui pourraient ouvrir la voie scientifique pour organiser les systèmes de gestion de données massives plus rentable et de haute qualité de services.

8.2 Limites de l'étude

Alors que les lacs de données l'emportent sur les entrepôts de données à bien des égards, ils ne sont utilisés que par les grandes entreprises qui travaillent avec une énorme quantité de données hétérogènes. La contrainte du budget d'investissement et le manque de connaissances techniques et de ressources humaines professionnelles, sont les causes les plus importantes qui empêchent l'organisation d'utiliser le lac de données et de le remplacer par des technologies anciennes.

L'une des limitations importantes de notre étude, en particulier dans la partie expérimentation du modèle mathématique, était le manque de données réelles et de la littérature associée à l'architecture du lac de données. Ces contraintes sont dues aux raisons principales suivantes :

- *Le lac de données est un sujet inconnu pour de nombreuses entreprises ou elles n'ont pas la possibilité de le mettre en œuvre et de l'utiliser ou de le partager (confidentialité)*
- *Notre sujet de thèse s'inscrit dans des domaines interdisciplinaires et il a été difficile de convaincre les organisations*
- *Manque de données réutilisables dans les revues littératures concernant le fonctionnement des différentes couches de l'architecture du lac de données pour examiner le modèle mathématique en conditions réelles*

Bien que ces limites indiquées ne nous empêchent pas d'aborder notre idée du lac de données logistique et d'utiliser des stratégies et des méthodes mimiques pour optimiser ses performances, elles restent néanmoins les pistes essentielles pour compléter nos travaux qui seront menés dans des études futures.

8.3 Perspectives

Compte tenu de l'étude réalisée et des résultats obtenus, il est évident que nous entamons une voie multidisciplinaire pour organiser et optimiser des systèmes de gestion de données massives et nous sommes au début de ce parcours. Suite aux contributions présentées dans notre étude, plusieurs perspectives pratiques s'ouvrent à nous pour parfaire l'idée discutée. Pour cette raison, plusieurs futures études sont envisagées.

- ***Perspective 1 :*** *Proposition d'autres méthodes et les stratégies managériales des systèmes logistiques pour gérer et développer l'environemment de lac de données.*

  Dans cette étude, nous avons discuté des stratégies plutôt essentielles pour l'organisation du lac de données telles que le ALLD allégée et la conception du réseau logistique du lac de données. En revanche, ils restent plusieurs manières de gérer des systèmes complexes de lacs de données en utilisant des stratégies multidisciplinaires, qu'elles soient naturelles ou logistiques, par exemple :

  - ***ALLD résiliente :*** La résilience est une faculté nécessaire et essentielle pour toutes les structures systémiques qui renforce la capacité du système à prévenir les anomalies et à résister aux événements perturbateurs prévus ou imprévus, internes ou externes. La résilience du système naturel est définie comme une propriété écologique qui protège le système contre les interruptions environnementales. D'autre part, le système logistique résilient signifie une structure optimale qui augmente la capacité de tous les participants à réagir rapidement et efficacement aux scénarios perturbateurs internes et externes. Par conséquent, la résilience des systèmes informatiques ; en particulier tous les systèmes centrés sur les données qui sont potentiellement à risque de scénarios perturbateurs, est une fonctionnalité indispensable.

    Pour cette raison, nous pouvons revenir à la base de la définition du lac de données dans cette étude où la structure systémique du lac de données a été empruntée au système naturel ou au système logistique. Les stratégies multidisciplinaires pré-actives et réactives qui sont utilisées pour développer la résilience d'un système naturel ou logistique pourraient être les bons sources d'inspiration pour concevoir ALLD résiliente. Ces méthodes bio-inspirées et les cadres d'inspiration logistiques pourraient être nommés comme l'analyse du réseau écologique [1], l'analyse des métriques résilientes, la théorie de graphe pour analyser les impacts destructeurs sur les nœuds et les arcs, la gestion et l'atténuation des risques, ingénierie de structures résilientes, et méthode d'augmentation de la capacité d'adaptation, de la visibilité, de la flexibilité et de la redondance du systeème(stratégies de secours d'urgence) [Rezapour et al., 2017, Lee et al., 2014, Chatterjee and Layton, 2020a, Laun et al., 2021].

  - ***ALLD Fuzzy :*** Les incertitudes dans les systèmes informatiques nécessitent toujours des scénarios managériaux pour réduire les conséquences graves générées par les événements à forte et à faible probabilité. Les sources d'incertitude du lac de données pourraient être définies comme la qualité des données, la demande des utilisateurs ou les tentatives de fraude. En général, les incertitudes sont décrites par des termes

[1] Ecological Network Analysis

vagues et imprécis et cette propriété nécessite les méthodes les plus appropriées pour faire face aux cas vagues [Petrovic et al., 1999]. La théorie des sous-ensembles flous est une bonne solution pour analyser et interpréter les incertitudes afin de traiter des événements vagues à détecter dans différentes structures systémiques. Selon la définition de l'ALLD et les sources importantes d'incertitude inhérente, les stratégies de gestion des risques utilisées dans les systèmes logistiques utilisant la logique floue sont applicables à la gestion des risques des lacs de données. Ces stratégies pourraient être définies grâce au cadre de la logique floue pour l'évaluation des risques des lacs de données, modélisation du lac de données à l'aide de sous-ensembles flous, et avec la prise de décision multi-critères flous [2] pour détecter les situations perplexes [Rostamzadeh et al., 2018, Petrovic et al., 1999, Aqlan and Lam, 2015].

- ***ALLD allégée-agile :*** Comme nous l'avons expliqué au chapitre 3, la stratégie allégée-agile est une combinaison de deux stratégies avec les objectifs presque contradictoires. Il s'agit d'un système allégée-agile qui doit être capable de réduire les activités et les coûts supplémentaires tout en ayant la capacité de répondre rapidement aux besoins variés des clients [Goldsby et al., 2006]. Par conséquent, la gestion de l'ALLD allégée-agile nécessite une modélisation plus complexe par rapport au cas où une seule stratégie est appliquée pour atteindre les objectifs prédéfinis. Cette stratégie hybride nécessite un modèle d'optimisation multi-objectifs équitable dans lequel un objectif est défini comme la minimisation des coûts totaux du lac de données (comme la minimisation du temps d'exécution) et un autre objectif en même temps maximise la qualité de service et la rapidité de réponse aux attentes de utilisateurs (comme maximiser le nombre de Jobs ou de nœuds).

- ***Perspective 2 :*** *Développer l'architecture technique du lac de données.*

  Avec le changement évolutif des environnements Big Data, les attentes en matière d'architectures compatibles avec les nouvelles technologies sont améliorées. Les utilisateurs demandent des infrastructures plus adaptées, plus rentables et plus évolutives, mais en revanche plus équitables et économiques. Dès lors, la plate-forme technique du lac de données peut être développée avec les technologies les plus actuelles telles qu'une architecture hybride pour accueillir des données en lots et en temps réel ainsi que l'utilisation de technologies compatibles avec ces caractéristiques telles qu'Apache Spark au lieu de la technologie proposée dans cette étude. De plus, l'architecture en Cloud du lac de données ou l'architecture hybride (en Cloud et en On-premises) sont les structures rentables pour mettre en œuvre un lac de données évolutif, à faible coût et avec une haute qualité de services [Zagan and Danubianu, 2021]. Pour cette raison, nous pouvons améliorer notre architecture prédéfinie vers une architecture Cloud (ALLD en Cloud) et utiliser les stratégies que nous avons abordées dans cette étude pour optimiser la structure envisagée.

- ***Perspective 3 :*** *Fournir une extension du modèle mathématique multi-objectifs*

  Comme nous l'avons discuté, l'objectif d'optimisation de l'architecture du lac de données dans cette étude est basé sur la stratégie allégée pour concevoir un réseau optimisé de systèmes de stockage de données brutes. Pour cette raison, nous avons proposé un modèle d'optimisation mathématique à mono-objectif basé sur la stratégie allégée qui minimise

[2] fuzzy multi-criteria decision-making (MCDM)

tous les coûts totaux supplémentaires du lac de données. D'autre part, les enjeux et problématiques de conception d'un lac de données rentable et favorable aux attentes des organisations, nécessitent également de prendre en compte un compromis entre les coûts et les avantages (bénéfices) de ce système informatique [Goldsby et al., 2006]. Par conséquent, des modèles de conception et d'optimisation multi-objectifs basés sur la stratégie allégée-agile qui minimisent les coûts totaux du lac de données tout en maximisant simultanément les avantages et la rentabilité, pourraient être envisagés pour les recherches futures.

Le modèle mathématique simplifié de cette perspective pourrait être défini comme suit :

$$\begin{aligned} Max : \sum Avantages \\ Min : \sum Couts \\ Subjectto : Contraints \end{aligned} \tag{8.1}$$

- ***Perspective 4 :*** *Réaliser le lac de données proposé dans cette étude et examiner les modèles d'optimisation avec les données réelles*

  L'une des principales limites de cette étude était le manque de jeux de données réelles pour tester la véracité des stratégies utilisées. Comme nous l'avons indiqué, la nature multidisciplinaire inhérente à cette étude, rend difficile l'expérimentation du modèle dans le monde réel. Cependant, avec le développement de l'utilisation des lacs de données en tant que technologie appropriée pour le stockage et l'analyse de données brutes massives, il y a de l'espoir que cette architecture optimisée puisse être réalisée et que des stratégies proposées soient exprimées dans de futures recherches.

Bibliographie

[Abadi et al., 2003] Abadi, D. J., Carney, D., Cetintemel, U., Cherniack, M., Convey, C., Lee, S., Stonebraker, M., Tatbul, N., and Zdonik, S. (2003). Aurora: a new model and architecture for data stream management. *the VLDB Journal*, 12(2):120–139.

[Abdelmaguid and Dessouky, 2006] Abdelmaguid, T. F. and Dessouky, M. M. (2006). A genetic algorithm approach to the integrated inventory-distribution problem. *International Journal of Production Research*, 44(21):4445–4464.

[Abraham et al., 2019] Abraham, R., Schneider, J., and vom Brocke, J. (2019). Data governance: A conceptual framework, structured review, and research agenda. *International Journal of Information Management*, 49:424–438.

[Ahmadi Javid and Azad, 2010] Ahmadi Javid, A. and Azad, N. (2010). Incorporating location, routing and inventory decisions in supply chain network design. *Transportation Research Part E: Logistics and Transportation Review*, 46(5):582 – 597.

[Al-Othman et al., 2008] Al-Othman, W. B., Lababidi, H. M., Alatiqi, I. M., and Al-Shayji, K. (2008). Supply chain optimization of petroleum organization under uncertainty in market demands and prices. *European Journal of Operational Research*, 189(3):822–840.

[Ambe, 2009] Ambe, I. M. (2009). Agile supply chain: strategy for competitive advantage. In *THE PROCEEDINGS OF 5 th INTERNATIONAL STRATEGIC MANAGEMENT CONFERENCE*, page 659.

[Anderson et al., 2007] Anderson, D. L., Britt, F. F., and Favre, D. J. (2007). The 7 principles of supply chain management. *Supply Chain Management Review*, 11(3):41–46.

[Aqlan and Lam, 2015] Aqlan, F. and Lam, S. S. (2015). A fuzzy-based integrated framework for supply chain risk assessment. *International journal of production economics*, 161:54–63.

[Asif et al., 2012] Asif, F. M., Bianchi, C., Rashid, A., and Nicolescu, C. M. (2012). Performance analysis of the closed loop supply chain. *Journal of Remanufacturing*, 2(1):1–21.

[Avison, 1991] Avison, D. (1991). Merise: A european methodology for developing information systems. *European Journal of Information Systems*, 1(3):183–191.

[Ayers, 2000] Ayers, J. B. (2000). *Handbook of supply chain management*. CRC Press.

[Azarmand and Neishabouri, 2009] Azarmand, Z. and Neishabouri, E. (2009). Location allocation problem. In *Facility location*, pages 93–109. Springer.

[Baca, 2016] Baca, M. (2016). *Introduction to metadata*. Getty Publications.

[Backlund, 2000] Backlund, A. (2000). The definition of system. *Kybernetes*.

[Ballou, 2001] Ballou, R. H. (2001). Unresolved issues in supply chain network design. *Information Systems Frontiers*, 3(4):417–426.

[Bamrara, 2015] Bamrara, A. (2015). Evaluating database security and cyber attacks: A relational approach. *The Journal of Internet Banking and Commerce*, 20(2).

[Baptiste, 2009] Baptiste, J.-L. (2009). *Merise Guide pratique: Modélisation des données et des traitements, langage SQL*. Editions ENI.

[Barbierato et al., 2013] Barbierato, E., Gribaudo, M., and Iacono, M. (2013). Modeling apache hive based applications in big data architectures. In *Proceedings of the 7th International Conference on Performance Evaluation Methodologies and Tools*, pages 30–38.

[Barbot et al., 2019] Barbot, N., Miclet, L., and Prade, H. (2019). Analogy between concepts. *Artificial Intelligence*, 275:487–539.

[Beamon, 1998] Beamon, B. M. (1998). Supply chain design and analysis:: Models and methods. *International journal of production economics*, 55(3):281–294.

[Beamon, 1999] Beamon, B. M. (1999). Designing the green supply chain. *Logistics information management*.

[Belov and Nikulchev, 2021] Belov, V. and Nikulchev, E. (2021). Analysis of big data storage tools for data lakes based on apache hadoop platform. *International Journal of Advanced Computer Science and Applications*, 12:551–557.

[Benoît et al., 2010] Benoît, C., Norris, G. A., Valdivia, S., Ciroth, A., Moberg, A., Bos, U., Prakash, S., Ugaya, C., and Beck, T. (2010). The guidelines for social life cycle assessment of products: just in time! *The international journal of life cycle assessment*, 15(2):156–163.

[Berg et al., 2012] Berg, K., Seymour, D. T., and Goel, R. (2012). History of databases. *International Journal of Management & Information Systems (IJMIS)*, 17:29.

[Berntson et al., 2017] Berntson, G. G., Cacioppo, J. T., and Bosch, J. A. (2017). *From Homeostasis to Allodynamic Regulation*, page 401–426. Cambridge University Press.

[Bertinelli and Black, 2004] Bertinelli, L. and Black, D. (2004). Urbanization and growth. *Journal of Urban Economics*, 56(1):80–96.

[Beyer and Schwefel, 2002] Beyer, H.-G. and Schwefel, H.-P. (2002). Evolution strategies–a comprehensive introduction. *Natural computing*, 1(1):3–52.

[Bešinović, 2020] Bešinović, N. (2020). Resilience in railway transport systems: a literature review and research agenda. *Transport Reviews*.

[Borgström, 2010] Borgström, B. (2010). *Supply chain strategising: Integration in practice*. PhD thesis, Jönköping International Business School.

[Boucher et al., 1982] Boucher, D. H., James, S., and Keeler, K. H. (1982). The ecology of mutualism. *Annual Review of Ecology and Systematics*, 13(1):315–347.

[Bush et al., 2001] Bush, A. O., Fernandez, J. C., Esch, G. W., and Seed, J. R. (2001). *Parasitism: the diversity and ecology of animal parasites*. Cambridge university press.

[Cachon and Fisher, 2000] Cachon, G. P. and Fisher, M. (2000). Supply chain inventory management and the value of shared information. *Management science*, 46(8):1032–1048.

[Casado and Younas, 2015] Casado, R. and Younas, M. (2015). Emerging trends and technologies in big data processing. *Concurrency and Computation: Practice and Experience*, 27(8):2078–2091.

[Castet and Saleh, 2012] Castet, J.-F. and Saleh, J. H. (2012). On the concept of survivability, with application to spacecraft and space-based networks. *Reliability engineering & system safety*, 99:123–138.

[Chatterjee and Layton, 2020a] Chatterjee, A. and Layton, A. (2020a). Mimicking nature for resilient resource and infrastructure network design. *Reliability Engineering & System Safety*, 204:107142.

[Chatterjee and Layton, 2020b] Chatterjee, A. and Layton, A. (2020b). Mimicking nature for resilient resource and infrastructure network design. *Reliability Engineering & System Safety*, 204:107142.

[Chen and Lee, 2004] Chen, C.-L. and Lee, W.-C. (2004). Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. *Computers & Chemical Engineering*, 28(6-7):1131–1144.

[Chen, 2008] Chen, M. (2008). *A greedy algorithm with forward-looking strategy*. INTECH Open Access Publisher.

[Chopra and Meindl, 2007] Chopra, S. and Meindl, P. (2007). Supply chain management. strategy, planning & operation. In *Das summa summarum des management*, pages 265–275. Springer.

[Chow and Heaver, 2007] Chow, G. and Heaver, T. (2007). Logistics in north america. In *GLOBAL*, volume 12, page 403.

[Closs and McGarrell, 2004] Closs, D. J. and McGarrell, E. F. (2004). *Enhancing security throughout the supply chain*. IBM Center for the Business of Government Washington, DC.

[Combes, 2001] Combes, C. (2001). *Parasitism: the ecology and evolution of intimate interactions*. University of Chicago Press.

[Contreras et al., 2012] Contreras, I., Fernández, E., and Reinelt, G. (2012). Minimizing the maximum travel time in a combined model of facility location and network design. *Omega*, 40(6):847 – 860. Special Issue on Forecasting in Management Science.

[Cooper, 1963] Cooper, L. (1963). Location-allocation problems. *Operations research*, 11(3):331–343.

[Cornuéjols et al., 1983] Cornuéjols, G., Nemhauser, G., and Wolsey, L. (1983). The uncapicitated facility location problem. Technical report, Cornell University Operations Research and Industrial Engineering.

[Cottingham et al., 2001] Cottingham, K., Brown, B., and Lennon, J. (2001). Biodiversity may regulate the temporal variability of ecological systems. *Ecology Letters*, 4(1):72–85.

[Curtis, 2003] Curtis, S. A. (2003). The classification of greedy algorithms. *Science of Computer Programming*, 49(1-3):125–157.

[Czarnecka et al., 2017] Czarnecka, A., Butor, A., and Halemba, M. (2017). Lean supply chain management. *World Scientific News*, 72:177–183.

[Daskin, 2011] Daskin, M. (2011). *Network and Discrete Location: Models, Algorithms, and Applications*. John Wiley & Sons, 2011.

[Daskin et al., 2005] Daskin, M. S., Snyder, L. V., and Berger, R. T. (2005). Facility location in supply chain design. In *Logistics systems: Design and optimization*, pages 39–65. Springer.

[Dean and Ghemawat, 2008] Dean, J. and Ghemawat, S. (2008). Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113.

[Dessalles et al., 2016] Dessalles, J.-L., Gaucherel, C., and Gouyon, P.-H. (2016). *Le fil de la vie: la face immatérielle du vivant*. Odile Jacob.

[Dev and Patgiri, 2014] Dev, D. and Patgiri, R. (2014). Performance evaluation of hdfs in big data management. In *2014 International Conference on High Performance Computing and Applications (ICHPCA)*, pages 1–7. IEEE.

[Drohomeretski et al., 2012] Drohomeretski, E., Gouvea da Costa, S., Pinheiro de Lima, E., and Wachholtz, H. (2012). Lean supply chain management: Practices and performance measures. *62nd IIE Annual Conference and Expo 2012*, pages 1869–1880.

[Duval, 2001] Duval, E. (2001). Metadata standards: What, who & why. *Journal of Universal Computer Science*, 7(7):591–601.

[Eiben et al., 2003] Eiben, A. E., Smith, J. E., et al. (2003). *Introduction to evolutionary computing*, volume 53. Springer.

[Eichler et al., 2021] Eichler, R., Giebler, C., Gröger, C., Schwarz, H., and Mitschang, B. (2021). Modeling metadata in data lakes—a generic model. *Data & Knowledge Engineering*, 136:101931.

[Eskandarpour et al., 2015] Eskandarpour, M., Dejax, P., Miemczyk, J., and Péton, O. (2015). Sustainable supply chain network design: An optimization-oriented review. *Omega*, 54:11–32.

[Falasca et al., 2008] Falasca, M., Zobel, C. W., and Cook, D. (2008). A decision support framework to assess supply chain resilience. In *Proceedings of the 5th International ISCRAM Conference*, pages 596–605. Washington, DC.

[Fang, 2015] Fang, H. (2015). Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem. *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 820–824.

[Farahani et al., 2014] Farahani, R. Z., Rezapour, S., Drezner, T., and Fallah, S. (2014). Competitive supply chain network design: An overview of classifications, models, solution techniques and applications. *Omega*, 45:92–118.

[Fath et al., 2007] Fath, B. D., Scharler, U. M., Ulanowicz, R. E., and Hannon, B. (2007). Ecological network analysis: network construction. *Ecological modelling*, 208(1):49–55.

[Fernandes et al., 2014] Fernandes, D. R., Rocha, C., Aloise, D., Ribeiro, G. M., Santos, E. M., and Silva, A. (2014). A simple and effective genetic algorithm for the two-stage capacitated facility location problem. *Computers & Industrial Engineering*, 75:200 – 208.

[Fister Jr et al., 2013] Fister Jr, I., Yang, X.-S., Fister, I., Brest, J., and Fister, D. (2013). A brief review of nature-inspired algorithms for optimization. *arXiv preprint arXiv:1307.4186*.

[Foulonneau and Riley, 2014] Foulonneau, M. and Riley, J. (2014). *Metadata for digital resources: implementation, systems design and interoperability*. Elsevier.

[Frank and Chou, 1972] Frank, H. and Chou, W. (1972). Topological optimization of computer networks. *Proceedings of the IEEE*, 60(11):1385–1397.

[Fu and Meng, 2020] Fu, L. and Meng, F. (2020). A human disease transmission inspired dynamic model for closed-loop supply chain management. *Transportation Research Part E: Logistics and Transportation Review*, 134:101832.

[Gaur et al., 2017] Gaur, J., Subramoniam, R., Govindan, K., and Huisingh, D. (2017). Closed-loop supply chain management: From conceptual to an action oriented framework on core acquisition. *Journal of Cleaner Production*, 167:1415–1424.

[Giannoccaro, 2018] Giannoccaro, I. (2018). Centralized vs. decentralized supply chains: The importance of decision maker's cognitive ability and resistance to change. *Industrial Marketing Management*, 73:59–69.

[Giebler et al., 2019] Giebler, C., Gröger, C., Hoos, E., Schwarz, H., and Mitschang, B. (2019). Leveraging the data lake: current state and challenges. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 179–188. Springer.

[Giebler et al., 2021] Giebler, C., Schwarz, H., Mitschang, B., and und Web, T. (2021). The data lake architecture framework: A foundation for building a comprehensive data lake architecture. *Datenbanksysteme für Business, Technologie und Web (BTW 2021) 13.–17. September 2021 in Dresden, Deutschland*, page 351.

[Giebler. et al., 2018] Giebler., C., Stach., C., Schwarz., H., and Mitschang., B. (2018). Braid - a hybrid processing architecture for big data. In *Proceedings of the 7th International Conference on Data Science, Technology and Applications - DATA,*, pages 294–301. INSTICC, SciTePress.

[Gill et al., 2019] Gill, P. E., Murray, W., and Wright, M. H. (2019). *Practical optimization*. SIAM.

[Goldsby et al., 2006] Goldsby, T. J., Griffis, S. E., and Roath, A. S. (2006). Modeling lean, agile, and leagile supply chain strategies. *Journal of business logistics*, 27(1):57–80.

[Gorelik, 2019] Gorelik, A. (2019). *The enterprise big data lake: Delivering the promise of big data and data science*. O'Reilly Media.

[Gu et al., 2014] Gu, R., Yang, X., Yan, J., Sun, Y., Wang, B., Yuan, C., and Huang, Y. (2014). Shadoop: Improving mapreduce performance by optimizing job execution mechanism in hadoop clusters. *Journal of Parallel and Distributed Computing*, 74(3):2166–2179.

[Guide Jr and Van Wassenhove, 2009] Guide Jr, V. D. R. and Van Wassenhove, L. N. (2009). Or forum—the evolution of closed-loop supply chain research. *Operations research*, 57(1):10–18.

[Gunasekaran et al., 2004] Gunasekaran, A., Patel, C., and McGaughey, R. E. (2004). A framework for supply chain performance measurement. *International journal of production economics*, 87(3):333–347.

[Gunasekaran et al., 2001] Gunasekaran, A., Patel, C., and Tirtiroglu, E. (2001). Performance measures and metrics in a supply chain environment. *International journal of operations & production Management*.

[Gupta and Giri, 2018] Gupta, S. and Giri, V. (2018). Data lake ingestion strategies. In *Practical Enterprise Data Lake Insights*, pages 33–85. Springer.

[Hai et al., 2016] Hai, R., Geisler, S., and Quix, C. (2016). Constance: An intelligent data lake system. In *SIGMOD '16: Proceedings of the 2016 International Conference on Management of Data*, pages 2097–2100.

[Hall and Fagen, 2017] Hall, A. D. and Fagen, R. E. (2017). Definition of system. In *Systems Research for Behavioral Sciencesystems Research*, pages 81–92. Routledge.

[Han et al., 2011] Han, J., Haihong, E., Le, G., and Du, J. (2011). Survey on nosql database. In *2011 6th international conference on pervasive computing and applications*, pages 363–366. IEEE.

[Hashem et al., 2020] Hashem, I. A. T., Anuar, N. B., Marjani, M., Ahmed, E., Chiroma, H., Firdaus, A., Abdullah, M. T., Alotaibi, F., Ali, W. K. M., Yaqoob, I., et al. (2020). Mapreduce scheduling algorithms: a review. *The Journal of Supercomputing*, 76(7):4915–4945.

[Haslhofer and Klas, 2010] Haslhofer, B. and Klas, W. (2010). A survey of techniques for achieving metadata interoperability. *ACM Computing Surveys (CSUR)*, 42(2):1–37.

[Haupt and Haupt, 2004] Haupt, R. L. and Haupt, S. E. (2004). *Practical genetic algorithms*. John Wiley & Sons.

[Hauschild et al., 2018] Hauschild, M. Z., Rosenbaum, R. K., and Olsen, S. I. (2018). *Life cycle assessment*, volume 2018. Springer.

[Heilala et al., 2014] Heilala, J., Ruusu, R., Montonen, J., Vatanen, S., Bermell-Garcia, P., Astwood, S., Iwhiwhu, C., Kavka, C., Asnicar, F., Ricco, L., Scholze, S., Einramhof-Grama, C., Kotte, O., and Armijo, A. (2014). Epes white paper: Product concept collaborative manufacturability and sustainability assessment with (epes) eco process engineering system epes white paper product concept collaborative manufacturability and sustainability assessment with (epes) eco process engineering system corresponding. *EPES White paper, 2014*.

[Heinberg and Lerch, 2010] Heinberg, R. and Lerch, D. (2010). What is sustainability. *The post carbon reader*, pages 11–19.

[Herodotou and Babu, 2011] Herodotou, H. and Babu, S. (2011). Profiling, what-if analysis, and cost-based optimization of mapreduce programs. *Proceedings of the VLDB Endowment*, 4(11):1111–1122.

[Hillmann et al., 2008] Hillmann, D. I., Marker, R., and Brady, C. (2008). Metadata standards and applications. *The Serials Librarian*, 54(1-2):7–21.

[Ho et al., 2015] Ho, W., Zheng, T., Yildiz, H., and Talluri, S. (2015). Supply chain risk management: a literature review. *International Journal of Production Research*, 53(16):5031–5069.

[Holling, 1973] Holling, C. S. (1973). Resilience and stability of ecological systems. *Annual review of ecology and systematics*, 4(1):1–23.

[Holmes, 2017] Holmes, D. E. (2017). *Big data: a very short introduction*. Oxford University Press.

[Huan et al., 2004] Huan, S. H., Sheoran, S. K., and Wang, G. (2004). A review and analysis of supply chain operations reference (scor) model. *Supply chain management: An international Journal*.

[Husain et al., 2011] Husain, M., McGlothlin, J., Masud, M. M., Khan, L., and Thuraisingham, B. M. (2011). Heuristics-based query processing for large rdf graphs using cloud computing. *IEEE Transactions on Knowledge and Data Engineering*, 23(9):1312–1327.

[Inmon, 2016] Inmon, B. (2016). *Data Lake Architecture: Designing the Data Lake and avoiding the garbage dump*. Technics publications.

[Janvier-James, 2012] Janvier-James, A. M. (2012). A new introduction to supply chains and supply chain management: Definitions and theories perspective. *International Business Research*, 5(1):194–207.

[Joaquim and dos Santos Mello, 2020] Joaquim, J. L. M. and dos Santos Mello, R. (2020). An analysis of confidentiality issues in data lakes. In *Proceedings of the 22nd International Conference on Information Integration and Web-based Applications & Services*, pages 168–177.

[Kachaoui and Belangour, 2020] Kachaoui, J. and Belangour, A. (2020). From single architectural design to a reference conceptual meta-model:an intelligent data lake for new data insights. *International Journal*, 8(4).

[Kane, 1997] Kane, D. L. (1997). The impact of hydrologic perturbations on arctic ecosystems induced by climate change. In *Global change and arctic terrestrial ecosystems*, pages 63–81. Springer.

[Karp et al., 2011] Karp, D. S., Ziv, G., Zook, J., Ehrlich, P. R., and Daily, G. C. (2011). Resilience and stability in bird guilds across tropical countryside. *Proceedings of the National Academy of Sciences*, 108(52):21134–21139.

[Kenneth, 2013] Kenneth, L. (2013). *Optimization*. Springer Texts in Statistics 95. Springer, New York, NY, 2nd ed. 2013.. edition.

[Khine and Wang, 2018] Khine, P. P. and Wang, Z. S. (2018). Data lake: a new ideology in big data era. In *ITM web of conferences*, volume 17, page 03025. EDP Sciences.

[Kim et al., 2015] Kim, Y., Chen, Y.-S., and Linderman, K. (2015). Supply network disruption and resilience: A network structural perspective. *Journal of operations Management*, 33:43–59.

[Kovac, 2013] Kovac, M. (2013). Lean supply chain management. *Lean supply chain management*, pages 3–7.

[Kreps, 2014] Kreps, J. (2014). Questioning the lambda architecture. the lambda architecture has its merits, but alternatives are worth exploring. *O'Reilly Media. https://www. oreilly. com/ideas/questioning-the-lambda-architecture. Zugegriffen am*, 21:2020.

[L, 2013] L, L. R. (2013). *The metadata manual : a practical workbook*. Chandos Publishing, Oxford Cambridge New Delhi.

[Ladley, 2012] Ladley, J. (2012). Chapter 2 - definitions and concepts. In Ladley, J., editor, *Data Governance*, MK Series on Business Intelligence, pages 7–20. Morgan Kaufmann, Boston.

[Lambert and Cooper, 2000] Lambert, D. M. and Cooper, M. C. (2000). Issues in supply chain management. *Industrial Marketing Management*, 29(1):65–83.

[LaPlante, 2016] LaPlante, A. (2016). *Architecting data lakes*. O'Reilly Media.

[Laun et al., 2021] Laun, A., Mazzuchi, T. A., and Sarkani, S. (2021). Conceptual data model for system resilience characterization. *Systems Engineering*.

[Laurent et al., 2020] Laurent, A., Libourel, T., Madera, C., and Miralles, A. (2020). The gravity principle in data lakes. *Data Lakes*, 2:187–199.

[Lee and Billington, 1993] Lee, H. L. and Billington, C. (1993). Material management in decentralized supply chains. *Operations research*, 41(5):835–847.

[Lee et al., 2014] Lee, Y. H., Mari, S. I., and Memon, M. S. (2014). Sustainable and resilient supply chain network design under disruption risks. *Sustainability*, 2014:6666–6686.

[Lerner et al., 1954] Lerner, I. M. et al. (1954). Genetic homeostasis. *Genetic homeostasis*.

[Li et al., 2019] Li, G., Shao, S., and Zhang, L. (2019). Green supply chain behavior and business performance: Evidence from china. *Technological Forecasting and Social Change*, 144:445–455.

[Linkov and Palma-Oliveira, 2017] Linkov, I. and Palma-Oliveira, J. M. (2017). *Resilience and risk: Methods and application in environment, cyber and social domains*. Springer.

[Liu et al., 2021] Liu, P., Loudcher, S., Darmont, J., and Noûs, C. (2021). Archaeodal: A data lake for archaeological data management and analytics. In *25th International Database Engineering & Applications Symposium*, pages 252–262.

[Llave, 2018] Llave, M. R. (2018). Data lakes in business intelligence: reporting from the trenches. *Procedia Computer Science*, 138:516–524.

[Lo Giudice et al., 2018] Lo Giudice, P., Musarella, L., Sofo, G., and Ursino, D. (2018). An approach to extracting complex knowledge patterns among concepts belonging to structured, semi-structured and unstructured sources in a data lake. *Information Sciences*, 478.

[Lummus and Vokurka, 1999] Lummus, R. R. and Vokurka, R. J. (1999). Defining supply chain management: a historical perspective and practical guidelines. *Industrial management & data systems*.

[Maccioni and Torlone, 2018] Maccioni, A. and Torlone, R. (2018). Kayak: a framework for just-in-time data preparation in a data lake. In *International Conference on Advanced Information Systems Engineering*, pages 474–489. Springer.

[Madera, 2018] Madera, C. (2018). *L'évolution des systèmes et architectures d'information sous l'influence des données massives : les lacs de données*. Theses, Université Montpellier.

[Madera and Laurent, 2016] Madera, C. and Laurent, A. (2016). The next information architecture evolution: the data lake wave. In *Proceedings of the 8th International Conference on Management of Digital EcoSystems*, pages 174–180.

[Mahnam et al., 2009] Mahnam, M., Yadollahpour, M. R., Famil-Dardashti, V., and Hejazi, S. R. (2009). Supply chain modeling in uncertain environment with bi-objective approach. *Computers & Industrial Engineering*, 56(4):1535–1544.

[Martin, 1998] Martin, C. (1998). *Logistics and supply chain management: strategies for reducing cost and improving service*. Financial Times.

[Martínez-Jurado and Moyano-Fuentes, 2014] Martínez-Jurado, P. J. and Moyano-Fuentes, J. (2014). Lean management, supply chain management and sustainability: A literature review. *Journal of Cleaner Production*, 85:134–150. Special Volume: Making Progress Towards More Sustainable Societies through Lean and Green Initiatives.

[Marz and Warren, 2013] Marz, N. and Warren, J. (2013). *Big Data: Principles and best practices of scalable real-time data systems*. Manning.

[Mason-Jones et al., 2000] Mason-Jones, R., Naylor, B., and Towill, D. R. (2000). Engineering the leagile supply chain. *International journal of agile management systems*.

[Mătăcuţă and Popa, 2018] Mătăcuţă, A. and Popa, C. (2018). Big data analytics: Analysis of features and performance of big data ingestion tools. *Informatica Economica*, 22(2):25–34.

[Mathew, 2012] Mathew, T. V. (2012). Genetic algorithm. *Report submitted at IIT Bombay*.

[McClanahan et al., 2012] McClanahan, T. R., Donner, S. D., Maynard, J. A., MacNeil, M. A., Graham, N. A., Maina, J., Baker, A. C., Alemu I, J. B., Beger, M., Campbell, S. J., et al. (2012). Prioritizing key resilience indicators to support coral reef management in a changing climate. *PLOS ONE*.

[Mehmood et al., 2019] Mehmood, H., Gilman, E., Cortes, M., Kostakos, P., Byrne, A., Valta, K., Tekes, S., and Riekki, J. (2019). Implementing big data lake for heterogeneous data sources. In *2019 ieee 35th international conference on data engineering workshops (icdew)*, pages 37–44. IEEE.

[Mell and Grance, 2011] Mell, P. and Grance, T. (2011). The nist definition of cloud computing.

[Mentzer et al., 2001] Mentzer, J. T., DeWitt, W., Keebler, J. S., Min, S., Nix, N. W., Smith, C. D., and Zacharia, Z. G. (2001). Defining supply chain management. *Journal of Business logistics*, 22(2):1–25.

[Miloslavskaya and Tolstoy, 2016] Miloslavskaya, N. and Tolstoy, A. (2016). Application of big data, fast data and data lake concepts to information security issues. In *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*.

[Min and Kim, 2012] Min, H. and Kim, I. (2012). Green supply chain research: past, present, and future. *Logistics Research*, 4(1):39–47.

[Min and Zhou, 2002] Min, H. and Zhou, G. (2002). Supply chain modeling: past, present and future. *Computers & industrial engineering*, 43(1-2):231–249.

[Ming et al., 2007] Ming, S., FU, R.-x., Chen, Z., and XIN, Z.-h. (2007). Study on the agile supply chain management based on agent. *The Journal of China Universities of Posts and Telecommunications*, 14:115–118.

[Mirjalili, 2019] Mirjalili, S. (2019). Introduction to evolutionary single-objective optimisation. In *Evolutionary Algorithms and Neural Networks*, pages 3–14. Springer.

[Misaki et al., 2016] Misaki, M., Tsuda, T., Inoue, S., Sato, S., Kayahara, A., and Imai, S.-i. (2016). Distributed database and application architecture for big data solutions. In *2016 International Symposium on Semiconductor Manufacturing (ISSM)*, pages 1–4. IEEE.

[Morgan Ernest and Brown, 2001] Morgan Ernest, S. and Brown, J. H. (2001). Homeostasis and compensation: the role of species and resources in ecosystem stability. *Ecology*, 82(8):2118–2132.

[Munshi and Mohamed, 2018] Munshi, A. A. and Mohamed, Y. A.-R. I. (2018). Data lake lambda architecture for smart grids big data analytics. *IEEE Access*, 6:40463–40471.

[Muralikrishna and Manickam, 2017] Muralikrishna, I. V. and Manickam, V. (2017). Chapter five - life cycle assessment. In Muralikrishna, I. V. and Manickam, V., editors, *Environmental Management*, pages 57–75. Butterworth-Heinemann.

[Murdoch and Oaten, 1975] Murdoch, W. W. and Oaten, A. (1975). Predation and population stability. In *Advances in ecological research*, volume 9, pages 1–131. Elsevier.

[Muñoz et al., 2012] Muñoz, E., Capón, E., Laínez, J. M., Moreno-Benito, M., Espuña, A., and Puigjaner, L. (2012). Operational, tactical and strategical integration for enterprise decision-making. In Bogle, I. D. L. and Fairweather, M., editors, *22nd European Symposium on Computer Aided Process Engineering*, volume 30 of *Computer Aided Chemical Engineering*, pages 397 – 401. Elsevier.

[Myerson, 2012] Myerson, P. (2012). *Lean supply chain and logistics management*. McGraw-Hill Education.

[Nargesian et al., 2019] Nargesian, F., Zhu, E., Miller, R. J., Pu, K. Q., and Arocena, P. C. (2019). Data lake management: challenges and opportunities. *Proceedings of the VLDB Endowment*, 12(12):1986–1989.

[Nogueira et al., 2018] Nogueira, I. D., Romdhane, M., and Darmont, J. (2018). Modeling data lake metadata with a data vault. In *Proceedings of the 22nd International Database Engineering & Applications Symposium*, pages 253–261.

[Oliver et al., 2015] Oliver, T. H., Heard, M. S., Isaac, N. J., Roy, D. B., Procter, D., Eigenbrod, F., Freckleton, R., Hector, A., Orme, C. D. L., Petchey, O. L., et al. (2015). Biodiversity and resilience of ecosystem functions. *Trends in ecology & evolution*, 30(11):673–684.

[Oussous et al., 2018] Oussous, A., Benjelloun, F.-Z., Lahcen, A. A., and Belfkih, S. (2018). Big data technologies: A survey. *Journal of King Saud University-Computer and Information Sciences*, 30(4):431–448.

[Pagh and Cooper, 1998] Pagh, J. D. and Cooper, M. C. (1998). Supply chain postponement and speculation strategies: how to choose the right strategy. *Journal of business logistics*, 19(2):13.

[Panahi and Navimipour, 2019] Panahi, V. and Navimipour, N. J. (2019). Join query optimization in the distributed database system using an artificial bee colony algorithm and genetic operators. *Concurrency and Computation: Practice and Experience*, 31(17):e5218.

[Panayides, 2006] Panayides, P. M. (2006). Maritime logistics and global supply chains: towards a research agenda. *Maritime Economics & Logistics*, 8(1):3–18.

[Pasandideh et al., 2013] Pasandideh, S. H. R., Niaki, S. T. A., and Hajipour, V. (2013). A multi-objective facility location model with batch arrivals: two parameter-tuned meta-heuristic algorithms. *Journal of Intelligent Manufacturing*, 24(2):331–348.

[Paschalidi, 2015] Paschalidi, C. (2015). Data governance : A conceptual framework in order to prevent your data lake from becoming a data swamp. Master's thesis, Luleå University of Technology Department of Computer science, Electrical and Space engineering.

[Perea-Lopez et al., 2003] Perea-Lopez, E., Ydstie, B. E., and Grossmann, I. E. (2003). A model predictive control strategy for supply chain optimization. *Computers & Chemical Engineering*, 27(8-9):1201–1218.

[Petrovic et al., 1999] Petrovic, D., Roy, R., and Petrovic, R. (1999). Supply chain modelling using fuzzy sets. *International journal of production economics*, 59(1-3):443–453.

[Petrović et al., 2019] Petrović, G., Mihajlović, J., Ćojbašić, Ž., Madić, M., and Marinković, D. (2019). Comparison of three fuzzy mcdm methods for solving the supplier selection problem. *Facta Universitatis, Series: Mechanical Engineering*, 17(3):455–469.

[Peyravi and Moeini, 2020] Peyravi, N. and Moeini, A. (2020). Estimating runtime of a job in hadoop mapreduce. *Journal of Big Data*, 7(1):1–18.

[Pimm et al., 1991] Pimm, S. L., Lawton, J. H., and Cohen, J. E. (1991). Food web patterns and their consequences. *Nature*, 350(6320):669–674.

[Plotkin, 2020] Plotkin, D. (2020). *Data stewardship: An actionable guide to effective data management and data governance*. Academic Press.

[Power et al., 2001] Power, D. J., Sohal, A. S., and Rahman, S.-U. (2001). Critical success factors in agile supply chain management-an empirical study. *International journal of physical distribution & logistics management*.

[Pradeep, 2015] Pradeep, P. (2015). *Data lake development with big data : explore architectural approaches to building Data Lakes that ingest, index, manage, and analyze massive amounts of data using Big Data technologies*. Packt Publishing, Birmingham.

[Protocol, 2011] Protocol, G. G. (2011). Product life cycle accounting and reporting standard. *World Business Council for Sustainable Development and World Resource Institute*.

[Ramakrishnan et al., 2017] Ramakrishnan, R., Sridharan, B., Douceur, J. R., Kasturi, P., Krishnamachari-Sampath, B., Krishnamoorthy, K., Li, P., Manu, M., Michaylov, S., Ramos, R., et al. (2017). Azure data lake store: a hyperscale distributed file service for big data analytics. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 51–63.

[Rangarajan et al., 2015] Rangarajan, S., Liu, H., Wang, H., and Wang, C.-L. (2015). Scalable architecture for personalized healthcare service recommendation using big data lake. In *Service research and innovation*, pages 65–79. Springer.

[Rasooli and Down, 2014] Rasooli, A. and Down, D. G. (2014). Coshh: A classification and optimization based scheduler for heterogeneous hadoop systems. *Future Generation Computer Systems*, 36:1–15. Special Section: Intelligent Big Data Processing Special Section: Behavior Data Security Issues in Network Information Propagation Special Section: Energy-efficiency in Large Distributed Computing Architectures Special Section: eScience Infrastructure and Applications.

[Ravat and Zhao, 2019] Ravat, F. and Zhao, Y. (2019). Data lakes: Trends and perspectives. In *International Conference on Database and Expert Systems Applications*, pages 304–313. Springer.

[Rezapour et al., 2017] Rezapour, S., Farahani, R. Z., and Pourakbar, M. (2017). Resilient supply chain network design under competition: A case study. *European Journal of Operational Research*, 259(3):1017–1035.

[Rohde and Vidal, 2020] Rohde, P. D. and Vidal, M.-E. (2020). Optimizing federated queries based on the physical design of a data lake. *arXiv preprint arXiv:2002.08102*.

[Rooney et al., 2019] Rooney, S., Bauer, D., Garcés-Erice, L., Urbanetz, P., Froese, F., and Tomic, S. (2019). Experiences with managing data ingestion into a corporate datalake. In *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*, pages 101–109. IEEE.

[Ross and Jayaraman, 2008] Ross, A. and Jayaraman, V. (2008). An evaluation of new heuristics for the location of cross-docks distribution centers in supply chain network design. *Computers & Industrial Engineering*, 55(1):64–79.

[Rostamzadeh et al., 2018] Rostamzadeh, R., Ghorabaee, M. K., Govindan, K., Esmaeili, A., and Nobar, H. B. K. (2018). Evaluation of sustainable supply chain risk management using an integrated fuzzy topsis-critic approach. *Journal of Cleaner Production*, 175:651–669.

[Rothlauf, 2011] Rothlauf, F. (2011). Optimization methods. In *Design of Modern Heuristics*, pages 45–102. Springer.

[Ruan, 2017] Ruan, G. (2017). Closed-loop big data analysis with visualization and scalable computing. *Big Data Research*, 8.

[Russom, 2017] Russom, P. (2017). Data lakes: Purposes, practices, patterns, and platforms. *TDWI white paper*.

[Saed et al., 2018] Saed, K. A., Aziz, N., Ramadhani, A. W., and Hassan, N. H. (2018). Data governance cloud security assessment at data center. In *2018 4th International Conference on Computer and Information Sciences (ICCOINS)*, pages 1–4. IEEE.

[Said and Amin, 2015] Said, K. A. M. and Amin, M. A. M. (2015). Overview on the response surface methodology (rsm) in extraction processes. *Journal of Applied Science & Process Engineering*, 2(1):8–17.

[Santoso et al., 2005] Santoso, T., Ahmed, S., Goetschalckx, M., and Shapiro, A. (2005). A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1):96–115.

[Savic, 2002] Savic, D. (2002). Single-objective vs. multiobjective optimisation for integrated decision support. *1st international congress on environmental modelling and software - lugano, Switzerland - June 2002*.

[Sawadogo and Darmont, 2021] Sawadogo, P. and Darmont, J. (2021). On data lake architectures and metadata management. *Journal of Intelligent Information Systems*, 56(1):97–120.

[Sawadogo et al., 2019a] Sawadogo, P., Scholly, É., Favre, C., Ferey, É., Loudcher, S., and Darmont, J. (2019a). Metadata systems for data lakes: Models and features. *CoRR*, abs/1909.09377.

[Sawadogo et al., 2019b] Sawadogo, P. N., Scholly, E., Favre, C., Ferey, E., Loudcher, S., and Darmont, J. (2019b). Metadata systems for data lakes: models and features. In *European conference on advances in databases and information systems*, pages 440–451. Springer.

[Sefraoui et al., 2012] Sefraoui, O., Aissaoui, M., and Eleuldj, M. (2012). Openstack: toward an open-source solution for cloud computing. *International Journal of Computer Applications*, 55(3):38–42.

[Servigne, 2008] Servigne, S. (2008). Conception, architecture et urbanisation des systemes d'information. *Encyclopædia Universalis, Encyclopædia Britannica*.

[Singh and Verma, 2018] Singh, D. and Verma, A. (2018). Inventory management in supply chain. *Materials Today: Proceedings*, 5(2):3867–3872.

[Skluzacek et al., 2021] Skluzacek, T. J., Wong, R., Li, Z., Chard, R., Chard, K., and Foster, I. (2021). A serverless framework for distributed bulk metadata extraction. In *Proceedings of the 30th International Symposium on High-Performance Parallel and Distributed Computing*, pages 7–18.

[Sundaram and Vidhya, 2016] Sundaram, D. and Vidhya, M. (2016). Data lakes-a new data repository for big data analytics workloads. *International Journal of Advanced Computer Research*, 7.

[Talbi, 2009] Talbi, E.-G. (2009). *Metaheuristics: from design to implementation*, volume 74. John Wiley & Sons.

[Thizy et al., 1985] Thizy, J.-M., van Wassenhove, L. N., and Khumawala, B. M. (1985). Comparison of exact and approximate methods of solving the uncapacitated plant location problem. *Journal of Operations Management*, 6(1):23–34.

[Thomas, 2006] Thomas, G. (2006). The dgi data governance framework; data gov. *Institute: Orlando, FL, USA*, 20.

[Tomcy, 2017] Tomcy, J. (2017). *Data Lake for enterprises*. Packt Publishing, Birmingham.

[Trivers and Dawkins, 1976] Trivers, R. and Dawkins, R. (1976). The selfish gene.

[Tschirhart, 2000] Tschirhart, J. (2000). General equilibrium of an ecosystem. *Journal of Theoretical Biology*, 203(1):13–32.

[Tylianakis et al., 2008] Tylianakis, J. M., Didham, R. K., Bascompte, J., and Wardle, D. A. (2008). Global change and species interactions in terrestrial ecosystems. *Ecology letters*, 11(12):1351–1363.

[Ulanowicz, 2004] Ulanowicz, R. E. (2004). Quantitative methods for ecological network analysis. *Computational biology and chemistry*, 28(5-6):321–339.

[Vachon, 2007] Vachon, S. (2007). Green supply chain practices and the selection of environmental technologies. *International Journal of Production Research*, 45(18-19):4357–4379.

[Van den Hof, 1997] Van den Hof, P. (1997). Closed-loop issues in system identification. *IFAC Proceedings Volumes*, 30(11):1547–1560.

[Van Hoek et al., 2001] Van Hoek, R. I., Harrison, A., and Christopher, M. (2001). Measuring agile capabilities in the supply chain. *International Journal of Operations & Production Management*.

[Vemuganti, 2013] Vemuganti, G. (2013). Metadata management in big data. *Big data: Countering tomorrow's challenges*, 3:3–77.

[Venner, 2009] Venner, J. (2009). Hdfs details for multimachine clusters. *Pro Hadoop*, pages 97–126.

[Walker and Alrehamy, 2015] Walker, C. and Alrehamy, H. (2015). Personal data lake with data gravity pull. In *2015 IEEE Fifth International Conference on Big Data and Cloud Computing*, pages 160–167.

[Wende, 2007] Wende, K. (2007). A model for data governance–organising accountabilities for data quality management. *18th Australasian Conference on Information Systems*.

[Westman, 1978] Westman, W. E. (1978). Measuring the inertia and resilience of ecosystems. *BioScience*, 28(11):705–710.

[WH, 2019] WH, I. (2019). *Data architecture : a primer for the data scientist*. Elsevier, San Diego, CA.

[White, 2012] White, T. (2012). *Hadoop: The definitive guide.* " O'Reilly Media, Inc.".

[Wilkinson et al., 2016] Wilkinson, M., Dumontier, M., Aalbersberg, I., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L., Bourne, P., Bouwman, J., Brookes, A., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C., Finkers, R., Gonzalez-Beltran, A., Gray, A., Groth, P., Goble, C., Grethe, J., Heringa, J., 't Hoen, P., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S., Martone, M., Mons, A., Packer, A., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.-A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., and Mons, B. (2016). The fair guiding principles for scientific data management and stewardship. *Scientific Data*, 3.

[Wu and Zhang, 2014] Wu, T. and Zhang, K. (2014). A computational study for common network design in multi-commodity supply chains. *Computers & Operations Research*, 44:206 – 213.

[Xia et al., 2015] Xia, M., Saxena, M., Blaum, M., and Pease, D. A. (2015). A tale of two erasure codes in {HDFS}. In *13th USENIX conference on file and storage technologies (FAST 15)*, pages 213–226.

[Yang and He, 2020] Yang, X.-S. and He, X.-S. (2020). *Nature-Inspired Computation in Data Mining and Machine Learning.* Springer International Publishing.

[Yazdani, 2014] Yazdani, M. (2014). An integrated mcdm approach to green supplier selection. *International Journal of Industrial Engineering Computations*, 5(3):443–458.

[Yebenes and Zorrilla, 2019] Yebenes, J. and Zorrilla, M. (2019). Towards a data governance framework for third generation platforms. *Procedia Computer Science*, 151:614–621. The 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops.

[Yu and Gen, 2010] Yu, X. and Gen, M. (2010). *Introduction to evolutionary algorithms.* Springer Science & Business Media.

[Yusuf et al., 2004] Yusuf, Y., Gunasekaran, A., Adeleye, E., and Sivayoganathan, K. (2004). Agile supply chain capabilities: Determinants of competitive objectives. *European Journal of Operational Research*, 159(2):379–392. Supply Chain Management: Theory and Applications.

[Zagan and Danubianu, 2021] Zagan, E. and Danubianu, M. (2021). Cloud data lake: The new trend of data storage. In *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–4. IEEE.

[Zhang et al., 2020] Zhang, Q., Wang, Y., and Zhang, L.-J. (2020). *Cloud Computing - CLOUD 2020: 13th International Conference, Held As Part of the Services Conference Federation, SCF 2020, Honolulu, HI, USA, September 18-20, 2020, Proceedings.* Lecture Notes in Computer Science. Springer International Publishing AG, Cham.

[Zhang et al., 2014] Zhang, Y., Qi, M., Miao, L., and Liu, E. (2014). Hybrid metaheuristic solutions to inventory location routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 70:305 – 323.

[Zhao et al., 2016] Zhao, L., Chen, L., Ranjan, R., Choo, K.-K. R., and He, J. (2016). Geographical information system parallelization for spatial big data processing: a review. *Cluster Computing*, 19(1):139–152.

[Zhao et al., 2021] Zhao, Y., Megdiche, I., and Ravat, F. (2021). Data lake ingestion management. *arXiv preprint arXiv:2107.02885*.

[Zikopoulos, 2015] Zikopoulos, P. (2015). *Big data beyond the hype: A guide to conversations for today's data center*. McGraw-Hill Education.

[Zsidisin and Siferd, 2001] Zsidisin, G. A. and Siferd, S. P. (2001). Environmental purchasing: a framework for theory development. *European Journal of Purchasing & Supply Management*, 7(1):61–73.