L'évolution des systèmes et architectures d'information sous l'influence des données massives : les lacs de données

Cédrine Madera

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

En Informatique

École doctorale Information, Structures, Systèmes

Unité de recherche LIRMM UMR 5506

L'évolution des systèmes et architectures d'information sous l'influence des données massives : Les lacs de données

Présentée par Cédrine MADERA
Le 22 Novembre 2018

Sous la direction de Anne Laurent
Thérèse Libourel et André Miralles

Devant le jury composé de

Jérôme DARMONT, Professeur, Université de Lyon, ERIC, Lyon	Rapporteur
Franck RAVAT, Professeur, Université de Toulouse, IRIT, Toulouse	Rapporteur
Marianne HUCHARD, Professeur, Université de Montpellier, LIRMM, Montpellier	Examinatrice
Claire NOY, Maître de Conférence, Université de Montpellier, ITIC, Montpellier	Examinatrice
Anne LAURENT, Professeur, Université de Montpellier, LIRMM, Montpellier	Directrice
Thérèse LIBOUREL, Professeur Émérite, Université de Montpellier, Espace DEV, Montpellier	Co-encadrante
André MIRALLES, Ingénieur de Recherche HDR, IRISTEA, TETIS, Montpellier	Invité

UNIVERSITÉ DE MONTPELLIER

Abstract

Data is on the heart of the digital transformation.The consequence is an acceleration of the information system evolution , which must adapt. The Big data phenomenon plays the role of catalyst of this evolution.
Under its influence appears a new component of the information system : the data lake. Far from replacing the decision support systems that make up the information system, data lakes come complete information systems's architecture.

First, we focus on the factors that influence the evolution of information systems such as new software and middleware, new infrastructure technologies, but also the decision support system usage itself. Under the big data influence we study the impact that this entails especially with the appearance of new technologies such as Apache Hadoop as well as the current limits of the decision support system .
The limits encountered by the current decision support system force a change to the information system which must adapt and that gives birth to a new component : the data lake. In a second time we study in detail this new component, formalize our definition, give our point of view on its positioning in the information system as well as with regard to the decision support system .

In addition, we highlight a factor influencing the architecture of data lakes : data gravity , doing an analogy with the law of gravity and focusing on the factors that may influence the data-processing relationship. We highlight, through a use case, that taking account of the data gravity can influence the design of a data lake. We complete this work by adapting the software product line approach to boot a method of formalizations and modeling of data lakes. This method allows us :

— to establish a minimum list of components to be put in place to operate a data lake without transforming it into a data swamp,
— to evaluate the maturity of an existing data lake,
— to quickly diagnose the missing components of an existing data lake that would have become a data swamp
— to conceptualize the creation of data lakes by being "software agnostic "

Keywords : Data Lakes, Data Governance, Data gravity, Decision support system, Information Systems, Data Science, data governance

Résumé

La transformation digitale place au coeur de son action la mise en valeur des données. Cela entraîne une accélération de l'évolution du système d'information, qui doit s'adapter. Le phénomène des données massives joue le rôle de catalyseur de cette évolution.
Sous son influence apparaît un nouveau composant du système d'information : le lac de données. Loin de remplacer les systèmes décisionnels qui composent le système d'information, les lacs de données viennent compléter les architectures des systèmes d'information.
Dans un premier temps nous nous intéressons aux facteurs qui influencent l'évolution des systèmes d'information tels que les nouveaux logiciels, les nouvelles technologies d'infrastructure mais aussi l'utilisation qui est faite des systèmes décisionnels.
Sous l'influence des données massives nous étudions l'impact que cela entraîne notamment avec l'apparition de nouvelles technologies comme Apache Hadoop ainsi que les limites actuelles des système décisionnels. Les limites rencontrées par les systèmes décisionnels actuels imposent une évolution au système d 'information qui doit s'adapter et qui donne naissance à un nouveau composant : le lac de données.
Dans un deuxième temps nous étudions en détail ce nouveau composant, formalisons notre définition, donnons notre point de vue sur son positionnement dans le système d'information ainsi que vis à vis des systèmes décisionnels.

Par ailleurs, nous mettons en évidence un facteur influençant l'architecture des lacs de données : la gravité des données, en dressant une analogie avec la loi de la gravité et en nous concentrant sur les facteurs qui peuvent influencer la relation donnée-traitement. Nous mettons en évidence, au travers d'un cas d'usage, que la prise en compte de la gravité des données peut influencer la conception d'un lac de données.
Nous terminons ces travaux par une adaptation de l'approche ligne de produit logiciel pour amorcer une méthode de formalisation et modélisation des lacs de données. Cette méthode nous permet :

— d'établir une liste de composants minimum à mettre en place pour faire fonctionner un lac de données sans que ce dernier soit transformé en marécage,
— d'évaluer la maturité d'un lac de donnée existant,
— de diagnostiquer rapidement les composants manquants d'un lac de données existant qui serait devenu un marécage,
— de conceptualiser la création des lacs de données en étant "logiciel agnostique".

Mots-clés : Lacs de données, Gouvernance de données, Systèmes d'information, Science des données, gravité des données, Système décisionnel.

Remerciements

Ce travail présenté dans ce manuscrit n'aurait jamais pu aboutir sans l'aide et le soutient de nombreuses personnes que je souhaite remercier particulièrement.

Tout d'abord, le coeur de ces travaux : l'équipe de choc constituée par ma directrice et mes co-directeurs, sans qui je n'aurai jamais entrepris cette aventure certes intellectuelle mais surtout humaine : **Anne Laurent, Thérèse Libourel-Rouge et André Miralles.**

En premier lieu, ma directrice de thèse, Anne Laurent, professeur des Universités de Montpellier et au Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), qui a accepté de diriger cette thèse et pris le risque de me prendre dans son équipe. Je la remercie très chaleureusement d'avoir accepté de m'encadrer, mais surtout de m'avoir soutenue, rassurée, faite progresser , tout en m'encourageant et me conseillant tout le long de cette thèse. Sa patience, sa disponibilité, sa bienveillance, ses connaissances et son enthousiasme sans égal ont fait d'elle une directrice extrêmement précieuse.

Ses points de vue originaux et son sens critique m'ont permis d'élargir ma vision du monde industrielle, d'améliorer et de perfectionner toutes les contributions de cette thèse. Sans son support, ses idées et son expérience, ce travail n'aurait pas été fait. Ces années passées à ses côtés ont juste étaient exceptionnelles d'échanges humains et intellectuels mais aussi ouverts sur des domaines de recherche et des équipes de chercheurs qui m'étaient inconnus. Elle a même réussit à me faire apprécier le langage LATEX ;-) !
Pour tout cela merci Anne !

Mes remerciements vont à Thérèse Libourel-Rouge, ma co-directrice plus que précieuse pour ses conseils, son expérience, son sens aigu de la justesse, ses connaissances plurielles, son esprit critique mais surtout son humanité. Je la remercie pour avoir toujours chercher l'excellence dans mes travaux de recherche, d'avoir toujours été à l'écoute, d'avoir gardé l'esprit ouvert sur toutes mes hypothèses et questionnements. Elle a toujours été là pour mes relectures et mes présentations données au dernier moment. Elle a su garder sa disponibilité et sa patience pour m'aider à progresser. Notre contribution sur la gravité des données, nous a passionné et fait partager notre passé de physiciennes, ce sont des instants de partage intellectuels et humains qui sont gravés à jamais dans ma mémoire.
Pour tout cela merci Thérèse !.

Je remercie très chaleureusement André Miralles, mon co-directeur qui a survécu à cette épopée

féminine, qui a fait preuve de patience et d'indulgence. Il m'a fait partagé son expérience et son parcours très enrichissant. Il m'a permis de découvrir la complexité des formalisations sur les lignes de produit, domaine sur lequel il a fait preuve d'une pédagogie et d'une patience extrême. Je lui suis particulièrement reconnaissante pour les intéressantes discussions que nous avons pu avoir, notamment sur les systèmes décisionnels, les échanges sur les systèmes d'information (quelle discussion sur les travaux de Le Moigne ! ! !), pour ses explications toujours claires et précises, sa très grande disponibilité et sa patience pour répondre à mes questions. Il a été d'une aide plus que précieuse pour la rédaction de ce mémoire en gardant un esprit critique et juste.
Pour tout cela merci André !.

Sans la mobilisation sans faille de cette équipe ce manuscrit n'aurait pas vu le jour, en temps et en heure : merci à vous trois pour m'avoir aidé, soutenu, relu, corrigé dans la rédaction de ce mémoire mais aussi préparés pour la soutenance qui en a découlé.

Ce travail n'aurait pas pu se conclure sans l'implication des membres de mon jury, qui ont accepté ce défi.
Je remercie tout d'abord, la présidente du jury, Marianne Huchard, qui m'a fait l'honneur d'accepter de diriger ma soutenance de thèse, d'avoir consacré de son temps précieux à lire ce manuscrit et à m'avoir fait des retours très judicieux pour poursuivre certains de mes travaux sur le parallèle entre les lignes de produits et les lacs de données. Je suis sûre qu'une belle collaboration est à venir.
Pour tout cela merci Marianne !.

Je remercie chaleureusement les autres membres de mon jury, mes deux rapporteurs, Franck Ravat et Jerôme Darmont ainsi que Claire Noy qui en plus du temps passé à lire mes travaux ont fait de ma soutenance un échange d'idées et de points de vues très enrichissants et ouverts des perspectives de collaborations futures très motivantes.
Pour tout cela merci Claire, Franck et Jerôme !.

Je souhaite remercier chaleureusement mes deux collègues Marie-Laure Pessoa et Guillaume Arnould, non seulement d'être venus me soutenir le jour de ma soutenance, mais aussi pour leurs encouragements, tout au long des mes travaux.
Pour tout cela merci Marie-Laure et Guillaume !.

Enfin, je voudrai remercier **ma famille** qui m'a épaulée et soutenue pendant ces années :

Mes parents, papa et maman, qui ont toujours cru en moi, ont fait preuve d'un soutien indéfectible dans cette expérience mais aussi tout au long de ma scolarité et ma vie professionnelle. Merci à vous mes parents ! sans vous, jamais je n'aurai pu y arriver.
Grâce à vous je suis fière du travail accompli et d'être votre fille.
Pour tout cela merci Maman et Papa !.

Sans la patience, le soutien et les encouragements de mon compagnon, Laurent, mes enfants Evan et Louann ,qui ont souffert de mon indisponibilité, je n'aurais pu terminer ce travail de rédaction. C'est au travers leur fierté de me voir soutenir (et défendre !) mes travaux que je mesure la valeur du travail accompli.
Pour tout cela merci Laurent, Evan et Louann !.

À vous, je dédie ce travail, avec tout mon amour.

Table des matières

1 Introduction **1**

1.1 Contexte . . . 2

1.2 Motivations et objectifs . . . 3

1.2.1 Motivations . . . 3

1.2.2 Objectifs . . . 4

1.3 Organisation du mémoire et contributions . . . 6

2 Les systèmes décisionnels du point de vue de l'architecture **9**

2.1 Introduction . . . 9

2.2 Le système d'information . . . 11

2.3 Les architectures des systèmes d'information . . . 14

2.4 Les systèmes décisionnels . . . 14

2.4.1 L'historique . . . 14

2.4.2 L'infocentre versus l'entrepôt de données . . . 18

2.4.3 Les composants d'un système décisionnel . . . 18

2.4.4 Les différences entre l'entrepôt de données et les magasins de données . . . 21

2.4.5 Les architectures des systèmes décisionnels . . . 22

2.4.6 L'architecture de référence . . . 26

3 L'évolution des systèmes décisionnels **33**

3.1 L'évolution des logiciels décisionnels . . . 33

3.1.1 Acquérir . . . 34

3.1.2 Stocker . . . 37

3.1.3 Exploiter . . . 41

3.1.4 Archiver . . . 42

3.2 L'évolution des infrastructures- introduction du concept HTAP . . . 43

3.2.1 Le temps réel . . . 44

3.2.2 En mémoire . . . 46

3.2.3 Hybridation . . . 48

3.2.4 Le stockage Flash . . . 48

3.3 L'évolution des données . . . 50

3.3.1 Données structurées, non structurées et semi-structurées : la différence 52
3.3.2 L'évolution des données sous l'influence des données massives 53
3.4 L'évolution des usages 55
3.4.1 L'analyse descriptive 55
3.4.2 L'analyse de diagnostic 57
3.4.3 L'analyse prédictive 58
3.4.4 L'analyse prescriptive 59
3.5 L'évolution de la modélisation 63
3.5.1 L'approche Data Vault 64
3.5.2 Règles de base d'une modélisation Data Vault : 68
3.5.3 Lien avec la modélisation dimensionnelle : 68
3.5.4 Avantages de Data Vault 68
3.5.5 Inconvénients de Data Vault 69
3.5.6 Conclusion sur les Data Vaults 69
3.6 La technologie Apache Hadoop 70
3.6.1 Historique de Apache Hadoop 70
3.6.2 Les enjeux du marché Apache Hadoop 70
3.6.3 Le lexique Apache Hadoop 71
3.6.4 Les composants d'Apache Hadoop 72
3.6.5 Les acteurs industriels autour de Hadoop 79
3.7 Apache Spark versus Apache Hadoop 84
3.8 L'impact de la technologie Hadoop sur les systèmes décisionnels 87
3.9 L'impact de l'évolution des systèmes décisionnels sur les architectures de référence 90
3.9.1 Les données - data sources 91
3.9.2 L'acquisition 91
3.9.3 Analytique en mouvement - Analytics in motion 92
3.9.4 Système d'exploitation du système décisionnel - Analytics Operating System . . 93
3.9.5 Zone de stockage du système décisionnel - Analytical Data Lake Storage 94
3.9.6 Accès aux données - data access 95
3.9.7 Découverte et exploration-Discovery et Exploration 96
3.9.8 Usages - Actionable Insight 96
3.9.9 Cas d'usage des données - Enhanced Application 97
3.9.10 Gouvernance et gestion de l'information - Information management governance . 97
3.9.11 La sécurité 99
3.9.12 Les plate-formes et infrastructures 100

3.10 Les limites des systèmes décisionnels . . . 101
3.11 Synthèse du chapitre 3 . . . 103

4 Vers un nouveau modèle d'architecture du système d'information intégrant le concept de lac de données 105
4.1 Introduction au lac de données . . . 106
4.2 État de connaissance des lacs de données - Discussions . . . 107
4.3 Notre définition des lacs de données . . . 113
4.4 Les enjeux des lacs de données . . . 114
4.5 Proposition d'un modèle pour les systèmes d'information avec des lacs de données . . . 116
4.6 Les lacs de données vis-à-vis des systèmes décisionnels . . . 118
4.6.1 L'acquisition des données . . . 118
4.6.2 Les données brutes . . . 120
4.6.3 Le temps réel . . . 121
4.6.4 La véracité . . . 121
4.6.5 Les utilisateurs . . . 121
4.7 Démarche d'urbanisation appliquée aux lacs de données . . . 124
4.7.1 L'architecture métier . . . 126
4.7.2 L'architecture fonctionnelle . . . 126
4.7.3 L'architecture applicative . . . 127
4.7.4 L'architecture technique . . . 128
4.8 Les fonctionnalités des lacs de données . . . 129
4.8.1 L'acquisition . . . 130
4.8.2 Le catalogage . . . 131
4.8.3 Le stockage . . . 135
4.8.4 L'exploitation ou l'exploration . . . 135
4.8.5 La gouvernance . . . 136
4.8.5.1 La sécurité . . . 136
4.8.5.2 Le cycle de vie . . . 137
4.8.5.3 La qualité . . . 138
4.9 Notre point de vue sur les lac de données . . . 138
4.10 Synthèse du chapitre 4 . . . 140

5 Influence de la gravité des données dans l'architecture des lacs de données 141
5.1 La gravité des données . . . 142

5.2 La gravité des données dans les lacs de données . . . 143
5.3 Impact de la gravité de la donnée sur les architectures des lacs de données . . . 147
5.3.1 L'impact du volume sur les lacs de données . . . 147
5.3.2 L'impact de la sensibilité sur les lacs de données . . . 149
5.3.3 L'impact du coût sur les lacs de données . . . 150
5.4 Étude de cas : prise en compte de la gravité des données sur un lac de données métrologie 152
5.4.1 L'approche et la méthodologie . . . 152
5.4.2 Description de l'étude de cas industriel . . . 152
5.4.2.1 Le contexte . . . 152
5.4.2.2 L'architecture fonctionnelle . . . 153
5.4.2.3 L'architecture applicative . . . 153
5.4.2.4 L'architecture technique . . . 154
5.4.3 Évaluation initiale du volume, du coût et de la sensibilité . . . 154
5.4.4 Évaluation de la gravité des données sur le lac de données métrologie . . . 156
5.4.4.1 Le volume . . . 157
5.4.4.2 La sensibilité . . . 157
5.4.4.3 Le coût . . . 158
5.4.5 Conclusion du cas d'étude de lac de données métrologie . . . 159
5.5 Synthèse du chapitre 5 . . . 161

6 Contribution à une démarche de formalisation des lacs de données via une approche ligne de produits 163
6.1 Nos attentes . . . 163
6.2 Modélisation des fonctionnalités d'un lac de données . . . 165
6.3 Constitution de la base de connaissance des lacs de données industriels . . . 166
6.4 Notre démarche . . . 167
6.5 L'approche ligne de produit - éléments de vocabulaire . . . 170
6.6 Application de notre démarche . . . 172
6.7 L'analyse des premiers résultats . . . 175
6.8 Synthèse du chapitre 6 . . . 176

7 Conclusion et perspectives 177
7.1 Conclusions . . . 177
7.2 Perspectives . . . 179

8 Annexes **181**
8.1 Questionnaire lac de données . . . 182
8.2 Ligne de Produit treillis . . . 194
8.2.1 Fonction Cataloguer . . . 194
8.2.1.1 Fonction Cataloguer-concept formel . . . 194
8.2.1.2 Fonction Cataloguer-AC-poset . . . 194
8.2.1.3 Fonction Cataloguer-FCA . . . 196
8.2.1.4 Fonction Cataloguer-AOC-poset . . . 196
8.2.2 Fonction Stocker . . . 196
8.2.2.1 Fonction Stocker-concept formel . . . 196
8.2.2.2 Fonction Stocker-AC-poset . . . 196
8.2.2.3 Fonction Stocker-FCA . . . 196
8.2.2.4 Fonction Stocker-AOC-poset . . . 196
8.2.3 Fonction Exploiter . . . 196
8.2.3.1 Fonction Exploiter-concept formel . . . 196
8.2.3.2 Fonction Exploiter-AC-poset . . . 196
8.2.3.3 Fonction Exploiter-FCA . . . 196
8.2.3.4 Fonction Exploiter-AOC-poset . . . 196
8.2.4 Fonction Gérer . . . 196
8.2.4.1 Fonction Gérer-concept formel . . . 196
8.2.4.2 Fonction Gérer-AC-poset . . . 196
8.2.4.3 Fonction Gérer-FCA . . . 196
8.2.4.4 Fonction Gérer-AOC-poset . . . 196

Bibliographie **211**

Table des figures

2.1 Explosion des données . . . 10
2.2 Position du système d'information - Le Moigne . . . 12
2.3 Position du système décisionnel dans le système d'information . . . 13
2.4 Vue d'une architecture de système décisionnel d'entreprise . . . 19
2.5 Architecture Décisionnelle pour un entrepôt central d'entreprise . . . 23
2.6 Architecture Décisionnelle pour des magasins de données indépendants . . . 24
2.7 Architecture Décisionnelle d'un système décisionnel d'entreprise . . . 25
2.8 Architecture de référence décisionnelle - IBM . . . 27

3.1 Magic Quadrant Gartner des outils d'intégration- outils E.T.L pour 2018 . . . 34
3.2 Comparaison des positions des acteurs du marché des outils d'intégration en 2018 et en 2008 . . . 35
3.3 Comparaison Interface graphique d'un outil ETL et du code pur . . . 36
3.4 Position des acteurs du marché des plateformes analytiques en 2018, Gartner magic quadrant. . . . 37
3.5 Tableau de comparaison des bases de données NoSQL [16] . . . 38
3.6 Système décisionnel avec une architecture hybride des stockages. . . . 41
3.7 Description d'une architecture de type HTAP par le Gartner . . . 45
3.8 Evolution des données structurées et non structurées . . . 50
3.9 les 4 V des Megadonnées . . . 52
3.10 Niveau de maturité de l'utilisation des systèmes décisionnels, vision du Gartner . . . 55
3.11 Positionnement des acteurs majeurs du marché des outils d'analyse descriptive et de diagnostic en 2018, vue du Gartner. . . . 56
3.12 Positionnement des acteurs majeurs du marché des outils d'analyse descriptive et de diagnostic en 2009, vue du Gartner. . . . 57
3.13 Hype cycle Gartner 2017- les 4 types d'analyses . . . 61
3.14 Structure de données du data Vault . . . 63
3.15 Modélisation de type ANCHOR MODELING . . . 64
3.16 Modèle de Data Vault . . . 65
3.17 Synthèse des trois entités d'une modélisation Data Vault . . . 66
3.18 Exemple de Modélisation Data Vault . . . 67

3.19 Historique de Apache Hadoop . . . 70
3.20 Prédiction du marché de la technologie Apache Hadoop en 2017 . . . 71
3.21 Apache Hadoop-Fichier HDFS . . . 72
3.22 Architecture Apache Hadoop . . . 73
3.23 Un environnement Hadoop . . . 74
3.24 Vue de la plateforme Apache Hadoop . . . 78
3.25 Distribution HortonWorks . . . 80
3.26 Distribution Cloudera . . . 81
3.27 MapR . . . 82
3.28 Description et composants de Apache Spark . . . 84
3.29 Courbe 'Hype" du Gartner pour les 2017 . . . 85
3.30 Mode de fonctionnement d'un Resilient Distributed Datasets -RDD avec Spark . . . 86
3.31 Architecture de référence analytique d'IBM . . . 90

4.1 Recherche d'information sur les lacs de données dans le moteur de recherche Google . . . 106
4.2 Architecture de Référence d'un lac de données proposée par IBM [10] . . . 109
4.3 Architecture de Référence d'un lac de données proposée par [73] . . . 112
4.4 Évolution du marché des solutions des lacs de données . . . 115
4.5 Marché des lacs de données : Moteurs, freins, opportunités et défis . . . 115
4.6 Proposition de positionnement des Lacs de données dans le Système d'information . . . 117
4.7 Interaction des lacs de données dans les systèmes d'une organisation . . . 118
4.8 Lac de données versus Système décisionnel : Schema on read versus schema on write . . . 120
4.9 Comparaison lac de données entrepôt de données . . . 123
4.10 Démarche d'urbanisation du système d'information . . . 125
4.11 Architecture fonctionnelle d'un lac de données . . . 126
4.12 Architecture applicative d'un lac de données . . . 127
4.13 Macro composant ou fonctions d'un lac de données . . . 130
4.14 Mutualisation du composant acquisition entre le système décisionnel et le lac de données 131
4.15 Interaction des lac de données dans les systèmes d'une organisation . . . 138
4.16 Vision d'une architecture globale du système d'information, sous l'influence des données massives . . . 139

5.1 Formule de MacCrory sur la gravité des données . . . 142
5.2 Architecture d'un lac de données mixte - en mode fédération et duplication . . . 149
5.3 Plateforme HortonWorks . . . 154

5.4 Évaluation initiale des contraintes non fonctionnelles pour le lac de données métrologie . 156
5.5 Évaluation de la sensibilité par type de serveur, pour le lac de données métrologie 158

6.1 Feature Model de la fonctionnalité cataloguer 165
6.2 Modélisation d'un lac de données par fonctionnalités 167
6.3 Base de connaissance - fonctionnalité Acquérir 168
6.4 Processus de production des lignes de produit appliqués dans ces travaux. 169
6.5 Un concept 171
6.6 Un treillis de concepts 171
6.7 Equivalent Class Feature Diagram - ECFD 172
6.8 Création du contexte formel pour la fonctionnalité Cataloguer, à partir de la base de connaissance 173
6.9 Contexte formel et treillis de concepts associés à la caractéristique "Sécuriser" des lacs de données 173
6.10 Comparaison de trois formats de treillis sur la fonction Sécuriser 174

8.1 Sommaire du questionnaire lac de données 182
8.2 Généralités 1/2 du questionnaire lac de données 183
8.3 Généralités 2/2 du questionnaire lac de données 184
8.4 Fonctionnalité Acquisition- questionnaire lac de données 185
8.5 Fonctionnalité Catalogage 1/2- questionnaire lac de données 186
8.6 Fonctionnalité Catalogage 2/2- questionnaire lac de données 187
8.7 Fonctionnalité du Cycle de vie 1/2- questionnaire lac de données 188
8.8 Fonctionnalité du Cycle de vie 2/2- questionnaire lac de données 189
8.9 Fonctionnalité Exploitation 1/2- questionnaire lac de données 190
8.10 Fonctionnalité Exploitation 2/2- questionnaire lac de données 191
8.11 Fonctionnalité Stockage- questionnaire lac de données 192
8.12 Fonctionnalité Securité- questionnaire lac de données 193
8.13 Concept formel-Cataloguer 194
8.14 AC-Cataloguer-simple 194
8.15 AC-Cataloguer-plein 195
8.16 FCA-Cataloguer-simple 195
8.17 FCA-Cataloguer-plein 196
8.18 AOC-Cataloguer-simple 196
8.19 AOC-Cataloguer-plein 196

8.20 Concept formel-Stocker 196
8.21 AC-stocker-simple 197
8.22 AC-stocker-plein 197
8.23 FCA-stocker-simple 198
8.24 FCA-stocker-plein 198
8.25 AOC-stocker-simple 199
8.26 AOC-stocker-plein 199
8.27 Concept formel-Exploiter 200
8.28 AC-Exploiter-simple 200
8.29 AC-Exploiter-plein 201
8.30 FCA-Exploiter-simple 202
8.31 FCA-Exploiter-plein 203
8.32 AOC-Exploiter-simple 204
8.33 AOC-Exploiter-plein 204
8.34 Concept formel-Gérer 204
8.35 AC-Gérer-simple 205
8.36 AC-Gérer-plein 206
8.37 FCA-Gérer-simple 207
8.38 FCA-Gérer-plein 208
8.39 AOC-Gérer-simple 209
8.40 AOC-Gérer-plein 210

Liste des sigles et acronymes

SI *Système d'Information*

BD *Base de Données*

SensorML *Sensor Model Language*

UML *Unified Modeling Language*

XML *eXtensible Markup Language*

URI *Universal Resource Identifier*

URL *Uniform Resource Locator*

URN *Uniform Resource Name*

ETL *Extract Transform Load*

DWH *Data Warehouse*

Stetl *Streaming ETL*

CSV *Comma Separated Values*

JS *JavaScript*

API Application Programming Interface

DM Data Mart

Data Lake DL

Data Vault DV

RDD Resilient Distributed Datasets

SD Système Décisionnel

ETL Extract Transform Load

ELT Extract Load Transform

MDA Model Driven Architecture

MOF Meta Object Facility

CWM Common Warehouse Metamodel

PIM Platform Independent Model

PSM Platform Specific Model

QVT Query/Views and Tranformations

ATL Atlas Transformation Language

Chapitre 1

Introduction

Préambule

Dans cette introduction, nous présentons d'abord le contexte de nos travaux, qui s'inscrivent dans le domaine des systèmes d'information. Plus précisément, nous proposons d'étudier l'évolution des systèmes d'information, au travers de l'évolution des systèmes décisionnels et l'apparition d'un nouveau système que sont les lacs de données. Nous abordons ensuite les motivations et les objectifs de nos travaux. Enfin, nous présentons le plan de ce manuscrit.

1.1 Contexte

Depuis quelques années la révolution numérique, les objets connectés, les applications mobiles, les smartphones, Internet, les réseaux sociaux, et autres sources multiplient les émissions de données, sous diverses formes et formats, suscitant des convoitises en matière d'information nouvelle. En effet une fois exploitées, ces données pourraient délivrer des informations auxquelles personne n'a encore pensé. Ces données forment désormais un **capital** au sein de ces organisations (notamment), qui les positionnent comme la pierre angulaire de leur projet de transformation numérique.
La conservation de ces données devient donc une priorité pour ces organisations, qui se constituent ainsi un **patrimoine de données**. De la capacité des organisations à tirer parti de ce patrimoine et à le transformer en informations pertinentes dépend, pour une majorité d'entre elles, leur survie face à la compétition [11].
Dans un monde en constante évolution, où les données sont de plus en plus nombreuses, en silos, la nécessité de les regrouper s'est imposée d'elle-même.
D'après une étude de PwC et d'Iron Mountain [60], 75% des dirigeants sont persuadés que le futur de leur entreprise repose sur leur capacité à tirer le meilleur de leurs données. Pour autant, seuls 4% d'entre eux estiment avoir mis en place une approche axée sur la donnée au sein de leur organisation.
Les organisations ont donc comme défi de créer une architecture d'entreprise moderne pour organiser, gérer, exploiter ces larges volumes de données de manière opérationnelle. Le système d'information de ces organisations, et donc son **architecture des données** doivent alors évoluer pour s'adapter à ces nouvelles attentes.
Depuis 2014, l'essor de certaines technologies, telles que l'apparition des systèmes de stockage arborescents comme Apache Hadoop[1] font émerger un nouveau concept pour tirer parti de ce capital de données : les « lacs de données »ou *data lake*.
D'abord assimilés simplement à un nouveau moyen de stocker des données, puis associés à un phénomène marketing, les lacs de données créent un engouement très fort dans le monde industriel, qui les adoptent de façon massive [51]. C'est donc sous l'impulsion, et la vision, très commerciale, du monde industriel que ce nouveau concept des lacs de données se positionne désormais comme incontournable dans le système d'information.
A titre d'exemple, HSBC[2], grande banque internationale, indique, pour réussir sa transformation numé-

1. Hadoop est un projet Open Source géré par Apache Software Fundation basé sur le principe Map Reduce et Google File System, deux produits Google Corp. Le produit est écrit en langage Java.
Hadoop peut être considéré comme un système de traitement de données évolutif pour le stockage et le traitement par lot de très grande quantité de données.

2. HSBC Bank plc est l'une des plus grandes organisations de services bancaires et financiers au monde. Le réseau international de HSBC comprend environ 7 500 bureaux dans plus de 80 pays et territoires en Europe, dans la région Asie-Pacifique, dans les Amériques, au Moyen-Orient et en Afrique (Wikipédia).

rique, travailler sur quatre axes, dont un qui est celui des usages autour de ses données, et pour cela elle compte s'appuyer sur la mise en place d'un lac de données.
Dans une interview de 2016, Darry West, dirigeant de HSBC, évoque le fait que [78] :
"Auparavant, la banque faisait beaucoup d'analyse de données hors ligne. Le but est de passer au temps réel pour détecter rapidement une tentative de fraude ou un changement de comportement de client. "
HSBC, qui emploie 255 000 personnes dans le monde et affiche un chiffre d'affaires de 71 milliards de dollars en 2015, s'est dotée pour cela d'un grand lac de données et met le cap sur le big data.
Les lacs de données sont donc mis en œuvre pour valoriser le patrimoine des données d'une organisation et accélérer sa transformation numérique. Ils s'imposent dans le système d'information des organisations.

1.2 Motivations et objectifs

Dans ce qui suit, nous expliquons pourquoi nous étudions les lacs de données dans le contexte de l'évolution des systèmes d'information et dans quels buts.

1.2.1 Motivations

Architecte en charge de la conception des systèmes d'information, chez IBM, auprès de d'industriels français et étrangers, depuis plus de vingt ans, je me passionne pour les données et leurs usages. De leur naissance à leur transformation en information utile et pertinente, aucun système d'information n'est identique dans sa conception et demande de constantes innovations, notamment au niveau de son architecture, pour s'adapter aux besoins de chaque organisation.
Cette permanente agilité de conception demande d'explorer toutes les nouvelles pistes de solutions, que ce soient les parties logicielles, les parties techniques ou fonctionnelles pour trouver les réponses adéquates, innovatrices et performantes au besoins des organisations. Cette agilité passe par une ouverture sur les deux mondes que sont les mondes académique et industriel.
Les lacs de données sont devenus une réalité dans la plupart des organisations. Si le sujet reste encore immature (moins de quatre ans de pratique dans le monde industriel), le positionnement stratégique dans la transformation numérique des organisations rend sa formalisation très importante dans la cadre des recherches sur l'évolution des système d'information.
De formation universitaire, ayant à plusieurs reprises collaboré à des programmes de recherche, j'ai rejoint une équipe de chercheurs passionnés par les données et décidé d'étudier, de façon plus approfondie, cette évolution des systèmes d'information, d'appréhender et de mieux formaliser ce nouveau composant qu'est le lac de données.
Nos travaux s'inscrivent dans la compréhension des évolutions des systèmes d'information, pour mieux

comprendre ce "phénomène" de lac de données, le positionner dans le système d'information et y apporter un point de vue académique.
Imaginer et appréhender les solutions capables d'exploiter toutes ces données et trouver l'information “de demain" représente un vrai défi, soulève des questions auxquelles ces travaux sont un début de réponse et est l'un des principaux déclencheurs de ce travail de thèse.

1.2.2 Objectifs

L'importance du sujet est certes récente (2014) mais les investissements financiers très conséquents qui sont faits au sein des organisations, comme l'illustre l'exemple de la banque HSBC, démontrent que les lacs de données ne sont pas un simple phénomène de mode.
Si cet engouement est suivi très fortement par tous les acteurs producteurs de logiciels et technologiques certaines questions n'ont pas été soulevées ou abordées, chacun de ces acteurs recherchant plus l'angle qui va avantager sa solution logicielle plutôt que de chercher un consensus sur une définition sur les architectures des lacs de donnée, leurs objectifs, leurs interactions, leurs utilisateurs et leurs positionnements en particulier vis-à-vis des systèmes décisionnels existants.
La littérature scientifique commence à s'intéresser au sujet mais reste encore limitée. D'où la motivation de ce travail qui propose une mise en perspective du concept de lac de données.
Le domaine des systèmes décisionnels a vu le consensus entre le monde industriel et celui de la littérature sur ses définitions, ses objectifs, ses utilisateurs ou sa sémantique, durant les trente dernières années. Il semble alors intéressant au travers d'une vision et approche académique, plus impartiale, d'étudier les lacs de donnée pour tenter d'apporter des éléments de réponses à nos questionnements.
Les travaux exposés dans ce mémoire ont pour objectifs de proposer une définition des lacs de données, d'amorcer une approche conceptuelle du sujet et de donner notre vision, d'architecte, sur le positionnement des lacs de données dans le système d'information ainsi que vis-à-vis des systèmes d'aide à la décision.

Notre objectif est donc d'étudier les composants essentiels constituant un lac de données, leurs objectifs, leurs usages mais aussi de relever l'importance de certains critères, qui peuvent remettre en cause des choix d'implémentation physique de leur architecture.

Ce sujet demande une vision académique pour compléter et enrichir la vision industrielle. Les expériences des organismes qui se sont lancés dans ces évolutions sont très nombreuses, avec des niveaux de maturité et de réussites différentes. Ces "expérimentations" sont aussi en attente d'un point de vue plus impartial, plus formel mais qui intègre aussi les innovations et directions proposées par les

industriels, comme cela a été fait autour des systèmes décisionnels.
C'est dans cet esprit que nous avons amorcé une collaboration intensive avec l'équipe de chercheurs en informatique que j'ai rejoint, y intégrant des collaborations ponctuelles avec des chercheurs d'autres domaines, tel que l'ingénierie des lignes de produits logiciels qui nous ont permis d'évoluer et avancer dans nos travaux. C'est l'apport de nos deux approches, le partage de nos deux modes de réflexions et d'expériences que j'expose dans ce mémoire.

Comment exploiter, acquérir, stocker, intégrer la masse de données disponibles au sein de ces lacs de données ? Y a-t-il des alternatives aux options technologiques prises par les industriels ? Quels en sont les impacts sur les architectures des lacs de données ? Quelles sont les interactions des lacs de données avec les systèmes décisionnels existants ? Quel est leur positionnement ? Comment les définir ? Comment les modéliser ? Comment les positionner dans le système d'information ? Quels sont les éléments clés de leur conception ? Comment vont-ils évoluer ? Quelles sont les perspectives ?
Plusieurs questions que nous nous posons en tant qu'architecte et auxquelles nous voulons contribuer au travers de cette thèse.

1.3 Organisation du mémoire et contributions

Ce mémoire est décomposé en huit chapitres.

Le chapitre 1 est consacré à la présentation du contexte de nos travaux, nos motivations et objectifs de ces travaux pour l'entreprise et à l'organisation du mémoire.

Le chapitre 2 présente un point de vue d'architecte sur les systèmes décisionnels. Il situe nos travaux dans le contexte du système d'information et expose notre vue sur les systèmes décisionnels, de leur naissance à l'état de connaissance que nous avons lors de ces travaux.
Dans ce chapitre les contributions sont :

— Positionnement du système décisionnel comme l'un des composants du système d'information ;

— Synthèse de l'architecture des systèmes décisionnels d'entreprise ;

— Analyse d'une architecture de référence d'un système décisionnel.

Dans le chapitre 3, nous étudions en détails l'évolution des systèmes décisionnels, durant ces vingt dernières années, plus spécifiquement les facteurs qui ont ou qui vont influencer les systèmes décisionnels, notamment les données massives et leurs impacts sur l'architecture de ces systèmes. Ce chapitre a pour objectif d'établir l'état de maturité des système décisionnels pour mieux appréhender le positionnement des lacs de données, au sein du système d'information.
Dans ce chapitre les contributions sont :

— Synthèse sur l'évolution des système décisionnels ;

— Analyse de l'impact des logiciels, des infrastructures, des usages, de la modélisation, de la technologie Apache Hadoop sur les systèmes décisionnels ;

— Introduction du concept d'architecture HTAP ;

— Étude de ces impacts sur une architecture de référence des systèmes décisionnels ;

— Propositions sur les limites des systèmes décisionnels actuels.

Le chapitre 4 est consacré aux lacs de données, les enjeux qu'ils représentent, leur positionnement dans le système d'information et surtout vis à vis des systèmes décisionnels existants. Nous donnons notre point de vue sur les composants et fonctions essentielles qui le composent et amorçons notre définition des lacs de données.
‘ Dans ce chapitre les contributions sont :

— Proposition de définition d'un lac de données ;

— État des lieux des connaissances sur les lacs de données ;

— Proposition d'un modèle pour les systèmes d'information avec les lacs de données ;

— Propositions de fonctionnalités des lacs de données.

Dans le chapitre 5, nous introduisons un facteur d'influence sur la conception des architectures des lacs de données : la gravité des données. Après avoir exposé le rationnel de cette prise en compte, nous en démontrons l'impact, sur l'architecture des lacs de données, au travers un cas d'usage industriel.
Dans ce chapitre les contributions sont :

— Proposition de définition de la gravité des donnés ;

— Analyse de l'impact de la prise en compte de la gravité des données sur une architecture de lac de données.

Dans le chapitre 6, nous adaptons une démarche lignes de produit logiciel pour proposer une ébauche de formalisation des lacs de données. Pour cela, nous constituons une base de connaissance des lacs de données provenant de nos collaborations avec le monde industriel et lui appliquons un processus de transformation semi-automatisé, provenant des lignes de produit. Dans ce chapitre les contributions sont :

— Constitution d'une base de connaissance de lac de données industriels ;

— Adaptation d'une approche ligne de produit et d'une chaîne de transformation semi-automatisée pour une formalisation des lacs de données ;

— Diagnostic de maturité d'un lac de données.

Enfin, le chapitre 7 conclut ces premiers travaux sur les lacs de données et présente quelques perspectives.

Le chapitre 8 rassemble les annexes de ce manuscrit.

La contribution principale de ce manuscrit est d'avoir introduit un composant dans le système d'information, les lacs de données, en ne l'opposant pas aux systèmes décisionnels existants, mais en le positionnant comme une évolution du système d'information, qui complète les systèmes décisionnels existants.

Chapitre 2

Les systèmes décisionnels du point de vue de l'architecture

2.1 Introduction

Selon une étude d'IDC (International Data Corporation) d'ici 2025, plus de 163 ZB[1] de données seront produites dans le monde [33]. Dans le monde industriel, cette "masse" de données est vue par le CIGREF[2] comme le nouvel "or noir" [11]. Cette analogie de la masse des données actuelle ou à venir avec l'ère du pétrole, et son impact sur le vingtième siècle, est intéressante à étudier car elle est reprise par beaucoup de commentateurs et acteurs du secteur du numérique[3]. Chacun y accentue les comparaisons pour appuyer l'impact qu'ont et vont avoir les données dans les organisations mais aussi dans nos modes de vie.

Les travaux de recherche du CIGREF sur le défi du numérique [11] sont basés sur plusieurs entretiens avec les plus grandes entreprises françaises, qui partagent leur expérience, vision et stratégie autour des données. Pour le CIGREF, les données vont :

- impacter l'évolution de notre civilisation ;
- devenir un enjeu économique ;
- devenir un enjeu géostratégique et politique ;
- devenir une ressource vitale du siècle ;
- conduire à la découverte de gisements.

S'il est vrai que cette masse de données va révolutionner la vie de nos entreprises et la façon dont elles opèrent leur "métier" [11], la comparaison avec le nouvel "or noir", même si elle est frappante et reprise par beaucoup de commentateurs du numérique, est à notre avis erronée.

1. 1 Zeta byte de données correspond à 10^{12} GigaOctet.
2. Le Cigref, Association loi 1901 créée en 1970, est un réseau de grandes entreprises et administrations publiques françaises qui se donnent pour mission de réussir le numérique. Il n'exerce aucune activité lucrative (https ://www.cigref.fr/).
3. https ://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data. https ://www.forbes.com/consent/ ?toURL=https ://www.forbes.com/sites/bernardmarr/2018/03/05/heres-why-data-is-not-the-new-oil/

FIGURE *2.1: Explosion des données*

En effet, le pétrole est une ressource difficilement renouvelable, sur le déclin, ce qui n'est en aucun cas la situation des données, dont le volume ne cesse de croître et leur exploitation à peine initiée, comme le montre la figure 2.1.

La figure 2.1 est extraite d'une étude qui montre que 90% des données qui existent aujourd'hui ont été produites durant les deux dernières années, et que ce nombre ne va cesser de croître dans les prochaines années. Nous ne considérons donc pas les données comme le nouvel "or noir" des entreprises ou organisations mais plutôt comme leur patrimoine ou capital, en expansion, qu'elles souhaitent exploiter au mieux en vue d'en tirer de la valeur.

Dans ce point de vue nous sommes rejoints par le CIGREF [12] qui voit les données d'une entreprise comme son capital et c'est au travers la valorisation de ce capital, que les entreprises vont pouvoir tirer de la valeur.

Le point de vue du CIGREF a évolué lui aussi entre son positionnement en 2014 [11], et celui décrit dans leur rapport en 2016 [12] qui repositionne les données n'ont plus comme l'or noir mais comme le

capital d'une organisation.
C'est l'exploitation de ce capital de données qui occupe l'axe central de nos travaux et notamment les solutions et architectures mises en place pour valoriser et tirer parti de ce capital : les lacs de données.
Pour mieux comprendre ce que sont les lacs de données et leur place dans le système d'information d'une organisation, nous nous intéressons tout d'abord aux systèmes d'information déjà en place et qui tirent déjà parti d'un ensemble de données dont ils disposent. Ces systèmes sont des systèmes dits d'aide à la décision (ou systèmes décisionnels) [6]. Au cœur de ces systèmes s'opère la transformation des données en information de valeur permettant la prise de décision.

Nos travaux de recherche portent sur l'évolution de ces systèmes décisionnels au travers de l'augmentation du volume et de la variété des données désormais disponibles dans une entreprise, l'évolution des attentes des entreprises sur l'exploitation de ces données, les innovations technologiques et l'impact sur l'architecture du système d'information.

2.2 Le système d'information

Plusieurs milliers de travaux académiques traitent du système d'information, de sa définition, sa conception, son positionnement, sa couverture, son approche, les domaines qui le composent. Le système d'information a même sa discipline de recherche dédiée, et il serait trop présomptueux d'essayer d'en faire une synthèse dans nos travaux.
Nous adoptons la définition simple d'un système d'information qui est la suivante et extraite des travaux de Servigne [70] :
Le système d'information (SI) est un ensemble organisé de ressources qui permet de collecter, stocker, traiter et distribuer de l'information, en général grâce à un ordinateur. Il s'agit d'un système socio-technique composé de deux sous-systèmes, l'un social et l'autre technique. Le sous-système social est composé de la structure organisationnelle et des personnes liées au SI. Le sous-système technique est composé des technologies (hardware, software et équipements de télécommunication) et des processus d'affaires concernés par le SI.
Les travaux francophones de [64] font une synthèse de vingt cinq ans d'articles sur le sujet et classent les sujets de recherche du SI en cinq grands domaines :

- informationnel, qui recouvre la gestion des données et des connaissances ;
- fonctionnel, qui englobe le traitement des transactions et l'aide aux tâches opérationnelles ;
- décisionnel, qui traite des processus de décision et d'aide à la décision ;

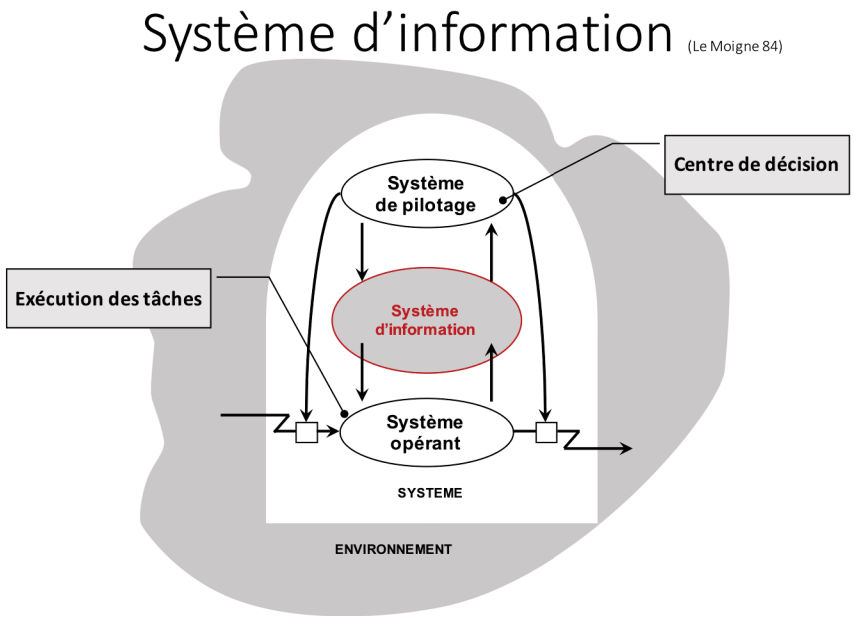


FIGURE 2.2: *Position du système d'information - Le Moigne*

— relationnel, qui rassemble les processus de communication ;

— général, quand le système d'information est traité dans sa globalité.

Dans nos travaux nous nous appuyons sur les travaux de Le Moigne [38], dans le domaine de la systémique, qui ont permis de dégager le modèle constituant la base de la majorité des approches actuelles du système d'information. Ce modèle distingue, dans une organisation, trois sous-systèmes :

— **le système opérant**, qui se compose de l'ensemble des ressources relatives à l'activité de l'entreprise ;

— **le système de pilotage** qui englobe l'ensemble des éléments responsables de la gestion et de la conduite de l'entreprise et de ses moyens ;

— **le système d'information**, vu comme outil de communication entre le système opérant et le système de pilotage.

Le but principal du système d'information, dans cette optique, est de fournir à chaque acteur de l'organisation toutes les informations sur sa situation actuelle, passée ou à venir. Le même agent peut se trouver virtuellement, soit au niveau du pilotage, soit au niveau opérant suivant la situation considérée. Le système d'information automatisé a repris ce modèle en offrant aux utilisateurs une "super" base de données dans laquelle chacun d'eux est susceptible de trouver ce dont il a besoin. C'est en fait autour

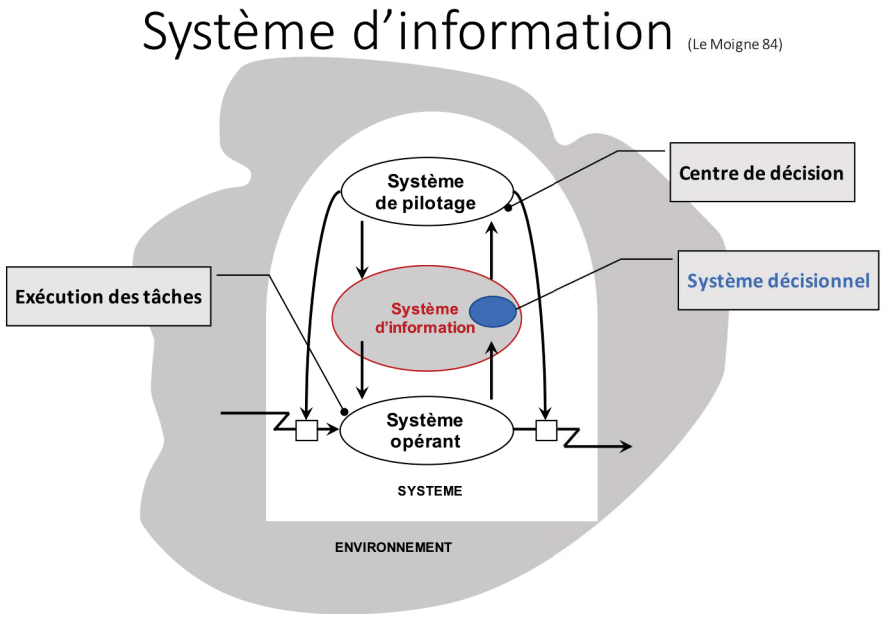


FIGURE 2.3: *Position du système décisionnel dans le système d'information*

d'elle que s'organise l'entreprise. C'est l'évolution de cette "super" base de données, et les facteurs qui en sont les vecteurs qui sont au centre de nos travaux. La figure 2.2 schématise cette vision. C'est au coeur de ce système d'information que sont localisés, les systèmes décisionnels, que nous considérons comme un composant du système d'information d'une organisation. La figure 2.3 illustre ce positionnement, en repartant du modèle de Le Moigne. Dans ce sens, nous rejoignons l'approche de J.Bucki et Y.Pesqueux[36], qui étend la vision de Le Moigne : Ils ne réduisent pas le concept de système d'information (SI) à une "super " base de données mais le considère comme un élément à part entière, plus complexe. Selon J.Bucki et Y.Pesqueux[36] Les données contenues dans le système d'information, orienté structure décisionnelle, sont regroupées autour des activités et des connaissances relatives à leur comportement. L'intégration du système décisionnel avec le système d'information devient dès lors naturelle, comme nous l'illustrons dans la figure 2.3.

Dans le monde industriel se retrouvent les mêmes notions de système opérant (ou transactionnel), système d'information et système de pilotage. Ces notions se retrouvent aussi dans la littérature scientifique anglo-saxonne et industrielle, sous les nominations de *System of Record*, *System of insights*, *System of engagement*.

Dans certaines organisations, le système d'information est parfois assimilé au système décisionnel, car c'est par lui qu'est présentée l'information. Un abus de langage est souvent fait entre système

d'information et système décisionnel.

Dans nos travaux nous distinguons ces deux appellations, et nous nous focalisons sur les systèmes décisionnels et leur architecture.

2.3 Les architectures des systèmes d'information

Dans son sens large, le terme "architecture" désigne un art de construire, de disposer ou de décorer un édifice. Ainsi, l'étude de l'architecture d'un système d'information consiste à examiner la structure d'un ensemble de composants fonctionnels, applicatifs, matériels et logiciels ainsi que le mode de relation qu'entretiennent ces composants.
Dans nos travaux, nous adoptons la démarche d'urbanisation, décrite par Servigne [69] : La démarche d'urbanisation recentre le pilotage de l'évolution du système d'information sur la stratégie et les besoins des métiers de l'entreprise ou organisation concernée. Elle est basée sur un modèle en quatre couches successives : Métier, Fonctionnelle, Applicative et Technique.
L'architecture d'information se décline donc selon quatre couches d'architecture :

- Architecture métier ;
- Architecture fonctionnelle ;
- Architecture applicative ;
- Architecture technique.

2.4 Les systèmes décisionnels

2.4.1 L'historique

Le concept de Systèmes d'Aide à la Décision (SAD), en Sciences de Gestion, a initialement été défini de manière formelle par A. Gorry et M. Scott Morton (1971) [25] : Le SAD est un « système informatisé interactif aidant le décideur à manipuler des données et des modèles pour résoudre des problèmes mal structurés ».
Dans ses travaux, en 2000, Lebraty [44] donne un panorama du concept de SAD, et le positionne comme un élément au sein d'un système décisionnel possédant une architecture spécifique.
Dans ses travaux, Ravat[61], positionne l'entrepôt de données comme élément essentiel du SAD.

Dans le cadre de nos travaux de recherche nous parlons de systèmes décisionnels de façon générique (le SAD y étant inclus) et nous appuyons sur la définition, des travaux de Teste [76] et Ravat[62][4],

suivante :

> **Définition :**
> Un système décisionnel est un système d'information dédié aux applications décisionnelles.

Dans certaines grandes organisations, le système décisionnel peut comprendre plusieurs SAD, on va parler alors de système décisionnel d'entreprise. Pour simplifier, lorsque nous parlons de système décisionnel, même si nous employons le singulier, le pluriel reste une possibilité, qui dépend du système d'information de chaque organisation.

Notre première proposition est une "évolution" de la définition précédente :

> **Proposition :**
> **Un système décisionnel est un COMPOSANT du système d'information, dédié aux applications décisionnelles.**

La figure 2.3 illustre ce positionnement dans le schéma de Le Moigne [38].

Dans la littérature le terme informatique décisionnelle, ou *Business Intelligence,* est lui aussi utilisé pour parler des systèmes décisionnels.

Qu'appelle-t-on Informatique décisionnelle ?

L'informatique décisionnelle[4] est un système informatique conçu pour donner un accès à l'information permettant la prise de décision. Ce système trouve son origine tout naturellement dans des entreprises où la connaissance d'information est primordiale pour la prise de décisions stratégiques ou de pilotage.
Ce sont donc grâce (ou à cause ?) de ces besoins en information, et bien sûr des avancées technologiques, que l'informatique décisionnelle a pu évoluer et se démocratiser au cours du temps.
Informatique destinée à ses débuts à cibler plus une population dite de «décideurs», elle s'est ensuite étendue à toute la population de l'entreprise et a même franchi les portes du monde de l'entreprise pour envahir le quotidien de toute la population. Qui n'a pas entendu parler d'indicateur de suivi d'une activité ? Notre quotidien fourmille de ces besoins de quantifier notre vie et de baser nos décisions sur des informations reçues.
Sans nous en rendre compte tous les jours nous utilisons des informations qui structurées sous une certaine forme et accessibles par différents moyens nous permettent de prendre des décisions : Par

4. Article MADERA Cedrine, Journal INSA LYON 2008. https ://docplayer.fr/4280036-Dossier-16-parcours-18-vie-des-laboratoires-21-vie-au-departement-juin-2008-informatique-decisionnelle.html

exemple regarder un bulletin météorologique va nous permettre de prendre, parfois, selon notre taux de confiance, la décision de porter tel vêtement plutôt qu'un autre ou de décider de partir pour un lieu de villégiature.
Sans nous en rendre compte nous utilisons l'informatique décisionnelle pour prendre notre décision ! En effet ce sont à l'origine des données telles que la température et la pression, qui mises sous une certaine forme nous servent à prendre une décision. Nous ne sommes plus des utilisateurs mais bien des consommateurs d'information.
C'est donc cette informatique mise à la portée de tous, grâce à une structuration, une organisation, une fiabilité et une mise en forme adéquate qui va nous permettre de prendre des décisions. On parle alors d'informatique décisionnelle.

Les sources de l'informatique décisionnelle : l'infocentre

Cette informatique la, prend ses sources très loin dans le temps, plus de trente années en arrière. Avant de parler d'information nous parlons de données. En effet l'informatique décisionnelle a la "magie" de transformer des données présentes dans les systèmes informatiques, souvent éparses, de formats et de provenances divers, parfois non fiables, en information. Pour constituer un système décisionnel le défi est donc d'avoir à disposition ces "données", les intégrer, les nettoyer et les structurer en vue d'être capable de supporter des processus d'aide à la décision.
Dés les années 70/80, grâce notamment à la percée des bases dites relationnelles (Edgar F. Codd[5] en 1970) les données informatiques présentes dans l'entreprise ont attisé les convoitises des décideurs et de ce fait la sollicitation intense des équipes du département informatique. Ces sollicitations pouvaient consister en des extractions des données opérationnelles, par exemple ou bien différentes "vues" d'une table, pour faciliter son exploitation. Un grand nombre de demandes leur étaient faites pour avoir accès à ces données. Ces demandes devenant récurrentes et perturbantes pour leurs activités quotidiennes, l'idée a germé de mettre à disposition toutes ces données dans un "endroit" déterminé, de donner l'accès en direct à ces données aux demandeurs et ainsi reprendre un peu d'indépendance : le concept de l'infocentre était né ![43].

Définition :
L'infocentre est une collection de données orientées sujet, intégrées, volatiles, actuelles, organisées pour le support d'un processus de décision ponctuel.

L'infocentre possède les caractéristiques suivantes :

— Dans un infocentre, chaque nouvelle valeur remplace l'ancienne valeur, il n'y pas de gestion d'his-

5. Edgar Frank «Ted» Codd (23 août 1923 - 18 avril 2003) est un informaticien britannique. Il est considéré comme l'inventeur du modèle relationnel des SGBDR.

torique des valeurs ;

— Les décisions prises sont des décisions opérationnelles basées sur des valeurs courantes ;

— Les processus d'alimentation de l'infocentre sont simples, ils consistent souvent en une duplication des données.

Si ce concept a un temps répondu aux besoins de ses utilisateurs, il n'a rapidement plus suffit et les limites des premiers systèmes décisionnels sont apparues : des limites technologiques (outils de visualisation, d'analyse, d'extraction) mais aussi de conception (historisation, information primaire).

En effet les données mises à disposition des utilisateurs étaient souvent volatiles dans le temps, peu fiables, non historisées et non structurées. L'accès à cet infocentre à des personnes non "formées" aux technologies informatiques n'était pas envisageable et les données n'étaient accessibles qu'à un nombre restreint de personnes. Avec la percée des ordinateurs personnels et la venue sur le marché informatique de nouveaux logiciels, l'informatique décisionnelle va prendre tout son essor dés le début des années 90. Ce sont alors les concepts, toujours d'actualité, d'un système décisionnel qui s'établissent [35], [41]. La notion d'entrepôt de données (*Data warehouse)* et de magasin de données (*Data Mart)* est introduite.

Définition : Le Data Warehouse est une collection de données orientées sujet, intégrées, non volatiles, historisées, organisées pour le support d'un processus d'aide à la décision.

Les infocentres vont alors évoluer, techniquement avec l'apport de nouveaux logiciels pour leur alimentation (les E.T.L[6]) et leur exploitation (outils d'analyse de type On Line Analyscal Processing OLAP [41][7]). C'est le premier stade marquant de l'évolution des systèmes décisionnels.

Si pour l'entrepôt de données les définitions s'accordent pour les magasins de données, deux visions s'affrontent : celle de Inmon [35] et celle Kimball [41] :

Définition : Définition d'Inmon : Le Data Mart est issu d'un flux de données provenant du Data Warehouse. Contrairement à ce dernier qui présente le détail des données pour toute l'entreprise, il a vocation à présenter la donnée de manière spécialisée, agrégée et regroupée fonctionnellement.

6. L'acronyme ETL (Extract, Transform, Load) est un processus d'intégration des données qui permet de transférer des données brutes d'un système source, de les préparer pour une utilisation en aval et de les envoyer vers une base de données, un entrepôt de données ou un serveur cible

7. OLAP acronyme de Online Analytical Processing, est une technologie permettant d'effectuer des analyses de données multidimensionnelles au sein de bases de données créées à cet effet.

Définition : Définition de Kimball : Le DataMart est un sous-ensemble du DataWarehouse, constitué de tables au niveau détail et à des niveaux plus agrégés, permettant de restituer tout le spectre d'une activité métier. L'ensemble des DataMarts de l'entreprise constitue le DataWarehouse.

Selon l'organisation qui l'utilise et l'architecte d'information qui va concevoir le système décisionnel, c'est l'une ou l'autre de ces définitions (pour les magasins de données) qui sera appliquée.

Dans le cadre de nos travaux nous adoptons la définition de Inmon, représentée graphiquement par le schéma d'architecture de la figure 2.4, que nous commentons en détail dans la section 2.4.3.

Il est important de rappeler que nos travaux s'axent sur l'étude des systèmes décisionnels, avec à la fois un angle académique et industriel. Ces travaux se basent aussi sur une expérience de mise en oeuvre de ces systèmes décisionnels auprès d'organisations de grande envergure, ayant des besoins de mise en oeuvre au niveau global (et donc des moyens et outils de même envergure), ce qui induit un point de vue axé sur les systèmes décisionnels d'entreprise, appelés aussi *Entreprise Data Warehouse*. Certains cas d'entrepôt de données ou systèmes décisionnels de moins grande envergure pourraient ne pas nécessiter ou être concernés par les composants que nous évoquons ci après. Lorsque nous employons le terme système décisionnel ou entrepôt de données, dans ce mémoire c'est bien au niveau "entreprise" que nous nous situons.

2.4.2 L'infocentre versus l'entrepôt de données

Les entrepôts de données deviennent plus complets, tirent parti des avancées technologiques qu'ils intègrent dans leur architecture et prennent alors le pas sur l'infocentre. Le tableau 2.1 compare les deux systèmes. Cependant l'infocentre ne disparaît pas totalement au profit des entrepôts de données et peut notamment être intégré parfois dans une architecture d'entrepôts de données. Dans la section 2.4.3 nous illustrons ce cas de figure.

2.4.3 Les composants d'un système décisionnel

Avec le concept de l'infocentre le système décisionnel primaire était né : un système informatique qui permet de transformer des données en information, capable d'aider à la prise de décision. Ce système décisionnel repose sur trois grandes phases [35] :

— l'acquisition (et le nettoyage !) des données pertinentes pour la prise de décision (phase 1) ;

Infocentre	Entrepôt de données
Collection de données	Collection de données
Orientées sujet	Orientées sujet
Intégrées	Intégrées
Volatiles	Non volatiles
Actuelles	Historisées
Conçu pour un processus de décision ponctuelle	Conçu pour des processus d'aide à la décision
Outils	Architecture

TABLE *2.1: Tableau de comparaison entre infocentre et entrepôt de données*

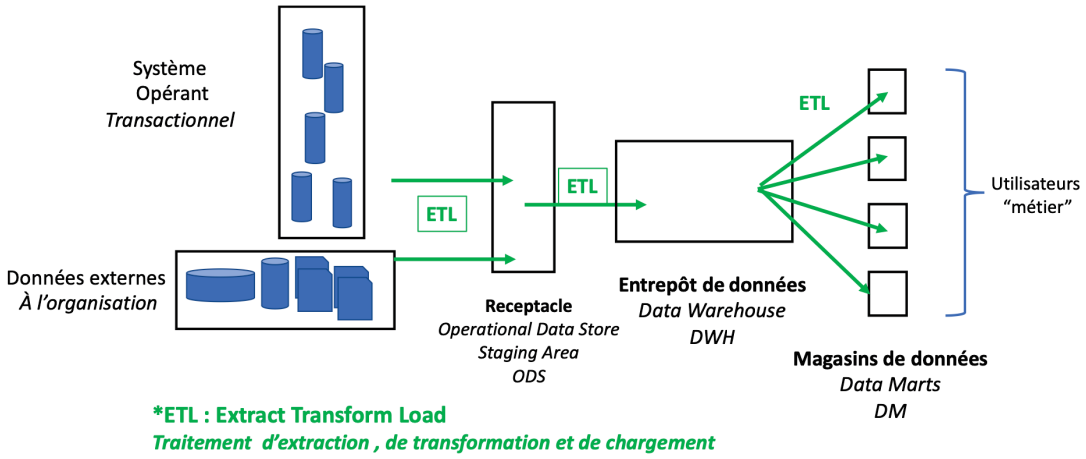


FIGURE *2.4: Vue d'une architecture de système décisionnel d'entreprise*

— La structuration, la mise en forme (indicateurs, agrégations...) et le stockage de ces données (phase 2) ;

— L'exploitation de ces données (phase 3).

A chaque phase décrite ci-dessus correspond un composant dans l'architecture d'un système décisionnel :

— **Phase 1** : nous parlerons de sas de préparation de l'information, de réceptacle ou d'Operational Data Store (ODS) ou Staging area ;

— **Phase 2** : nous parlerons de l'entrepôt de données ou Data Warehouse (DWH) ;

— **Phase 3** : nous parlerons de magasins de données ou de Data Mart (DM).

Ces phases sont illustrées dans la figure 2.4 où l'on retrouve ces trois composants (*cités dans leur version anglo-saxonne*) décrits dans les travaux d'Inmon [35] :

— L'Operational Data Store ou ODS ;

— Le Data Warehouse ou DWH ;

— Les Data Marts ou DM.

Le réceptacle (ou ODS) est le lieu d'intégration, de synchronisation, de normalisation, de reformatage, de "nettoyage" des différentes sources de données. Il est le sas de préparation des données pour l'entrepôt de données.
Parfois, notamment dans les organisations où il n'y a pas d'entrepôt de données central, les utilisateurs peuvent accéder aux tables ou fichiers de l'ODS en mode infocentre traditionnel pour réaliser certains rapports ou extractions de données ponctuelles.
Les données y sont en général volatiles (non historisées), cependant, quand il n'y a pas d'entrepôt de données central mais seulement des magasins de données, l'ODS alimente directement les Data Marts, le maintien de données de détail historisées au niveau de l'ODS devient nécessaire pour constituer des agrégats pertinents dans les Data Marts.
C'est lors de la phase de conception de l'architecture décisionnelle que se décide la configuration du système décisionnel et des composants qui vont y être implémentés.

Le Data Warehouse est d'intérêt pour l'ensemble des utilisateurs du système décisionnel. Sa conception orientée sujet, globalement normalisée pour être évolutive, prend en compte les besoins de l'ensemble des fonctions utilisatrices. Les données y sont historisées et maintenues à leur niveau de détail. Certains agrégats peuvent y être constitués.
Le Data Warehouse est utilisé notamment pour de l'exploitation de données de détail, historisées, ainsi que pour la fouille de données (ou le "data mining").

En synthèse, les caractéristiques principales d'un entrepôt de données sont :

— *Données organisées* : Les données dans l'entrepôt de données sont organisées et orientées par métiers et par applications ;

— *Données non volatiles* : Un entrepôt de données doit être conçu et construit comme un entrepôt où les transactions, les événements et les données métier sont non volatiles ;

— *Information intégrée, consolidée, dérivée et nettoyée* : Les données proviennent de sources de données disparates que l'entrepôt de données rend cohérentes et transforme en information fiable et pertinente ;

— *Réfèrent* : Il doit être la seule source de données pour les utilisateurs ;

— *Conservation de l'historique* : Il est construit sur la base d'un modèle de données 'temporel' : Plusieurs techniques de modélisation peuvent être utilisées pour constituer cet historique et le rendre exploitable.

Le Data Mart est une base de données destinée à quelques utilisateurs d'un département. C'est une petite structure très ciblée et pilotée par les besoins utilisateurs. Il a la même vocation que l'entrepôt de données (fournir une architecture décisionnelle) mais est dédié à un nombre d'utilisateurs plus restreint. Il vise le plus souvent à mesurer la performance et le design le plus adapté est la modélisation multidimensionnelle. Il est le plus fréquemment accédé par les utilisateurs pour leurs besoins d'analyse et compte généralement des agrégats pré calculés afin d'optimiser les temps de réponse des requêtes. Il est alimenté par l'entrepôt de données.

2.4.4 Les différences entre l'entrepôt de données et les magasins de données

Un magasin de données n'est pas un "mini" entrepôt de données. L'objectif d'un magasin de données est de servir un besoin métier spécifique, avec une application dédiée, voire un outil d'analyse ou de "reporting" qui lui est propre. L'entrepôt de données, lui, est plus générique et doit couvrir la vision transverse de son organisation.
Si parfois la "taille" (ou volume de données) d'un magasin de données est effectivement plus petite que celle d'un entrepôt de données, ce n'est pas un critère de différentiation.
Ne pas croire qu'avec une collection de magasins de données on va obtenir un entrepôt de données !
C'est sur ce point que l'approche de Imnon [35] s'oppose notamment à celle de Kimball [41].

Quelques principes à retenir lors de la création d'un magasin de données :

— impliquer les utilisateurs ;

— attention à la cohérence des données ;

— fédérer des Data Marts n'est pas facile ;

— ne pas construire de Data Mart isolé ;

— Attention à la prolifération de Data Marts 'sauvages' ;

— Avoir une vision de l'avenir .....(modèle).

Donc le magasin de données peut préparer à l'entrepôt de données (d'entreprise), mais il faut penser grand, avenir, et adopter des technologies capables d'évoluer. C'est une limitation qui est souvent associée à l'approche de Kimball [2].

2.4.5 Les architectures des systèmes décisionnels

Les trois éléments qui composent un système décisionnel ne sont pas obligatoirement créés lorsqu'un système décisionnel est mis en place, ce sont les besoins fonctionnels, métiers, mais aussi les contraintes techniques (ou non fonctionnelles)[70] qui vont décider quels sont les éléments qu'il faut inclure.

Il existe principalement trois scénarios pour les architectures décisionnelles (d'entrepôts de données entreprise) [47] :

— Architecture entrepôt de données central ;

— Architecture de magasins de données indépendants ;

— Architecture de système décisionnel d'entreprise.

Ces scénarios ne sont pas exhaustifs, d'autres dérivés sont utilisés, nous avons sélectionné les trois scénarios les plus répandus en milieu industriel, en avons tiré les observations et les positions que nous exposons dans la section 3.

Dans le paragraphe suivant nous détaillons les avantages et inconvénients de ces trois scénarios et illustrons chacun par un schéma d'architecture.

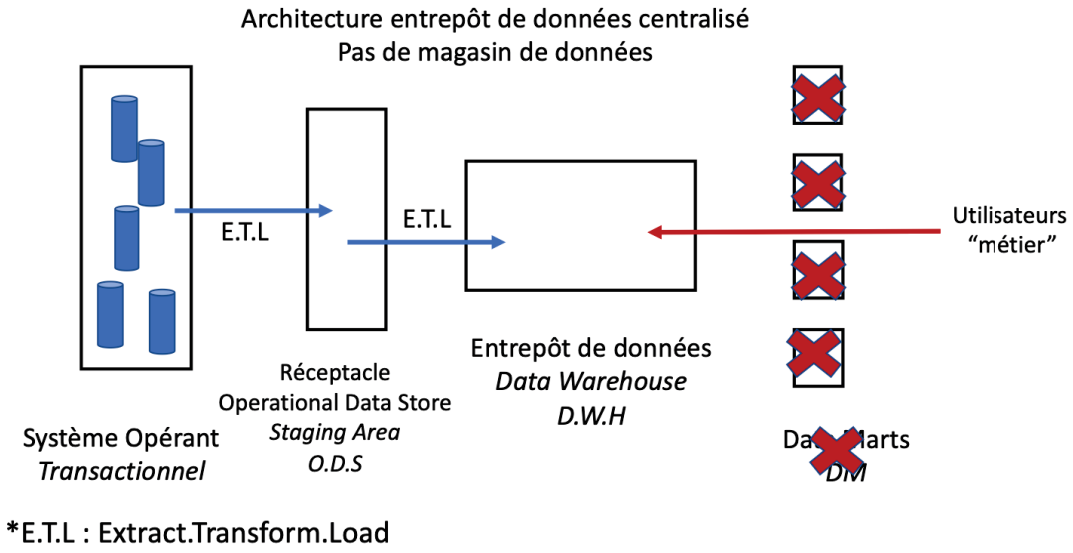


FIGURE 2.5: *Architecture Décisionnelle pour un entrepôt central d'entreprise*

Architecture entrepôt central

Avantages :

— Gain d'espace disque, l'information n'est stockée qu'une fois ;

— Pas de problème de synchronisation liées à la maintenance de plusieurs copies de données ;

— Recommandé lorsque les besoins de l'organisation sont génériques.

Inconvénients :

— Les données ne sont pas organisées pour satisfaire les besoins d'un ensemble d'utilisateurs spécifiques.

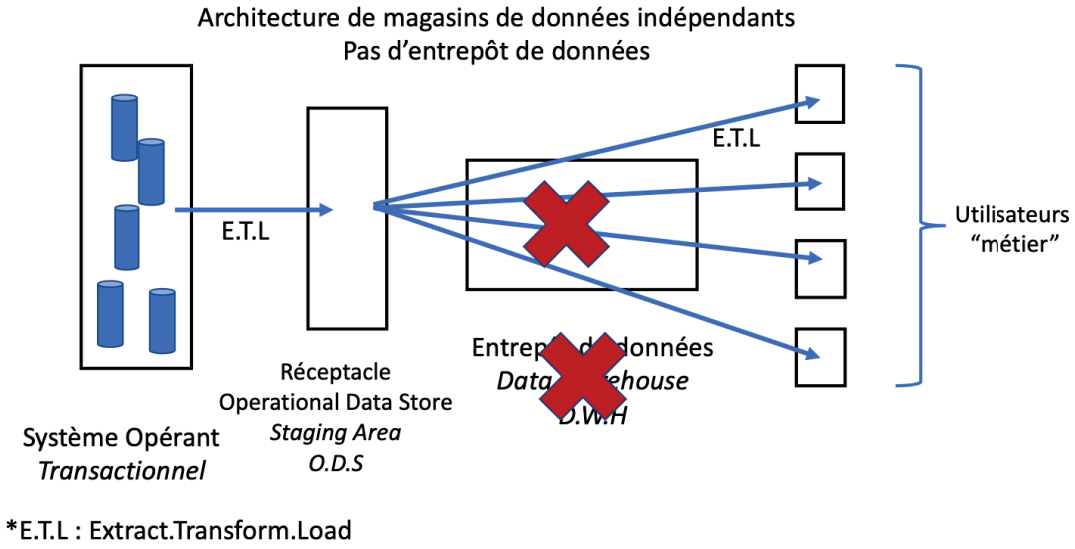


FIGURE *2.6: Architecture Décisionnelle pour des magasins de données indépendants*

Architecture de magasins de données indépendants

Avantages :

- Les Data Marts ont été optimisés pour les besoins (parfois uniques) d'utilisateurs spécifiques ;
- Recommandé lorsque les besoins de l'organisation sont génériques.

Inconvénients :

- Duplication des données (consommation d'espace disque, maintenances multiples) ;
- Complexité de l'alimentation : m sources pour n Data Marts ;
- Risques d'incohérences entre les Data Marts.

Ce type d'architecture est celui qui fait référence à l'approche de Kimball [41].

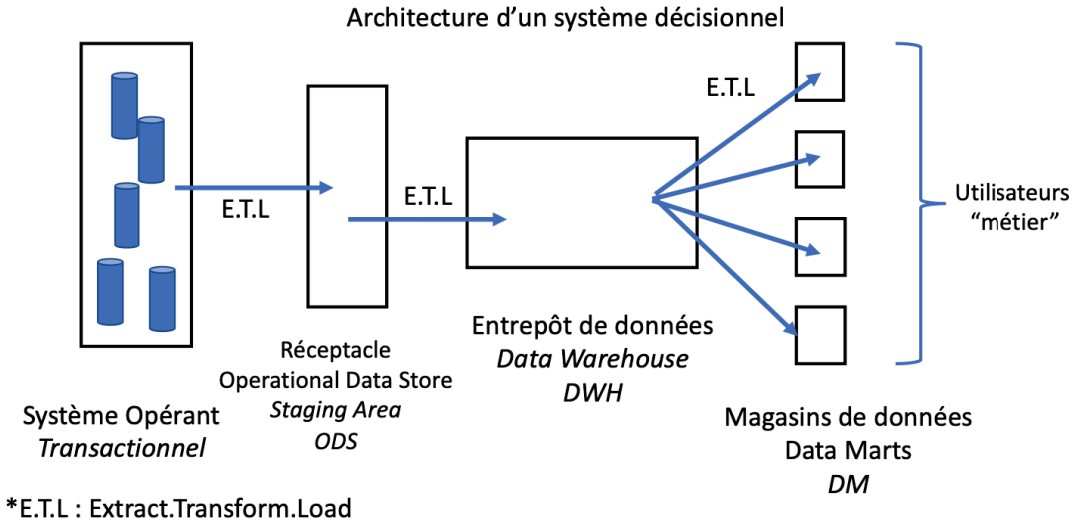


FIGURE 2.7: *Architecture Décisionnelle d'un système décisionnel d'entreprise*

Architecture d'un système décisionnel

Avantages :

- Option la plus "propre" ;
- Alimentation des magasins de données simplifiée : source unique ;
- Séparation claire entre les niveaux.
  - réconciliation / nettoyage ;
  - transformation / dérivation ;
  - propagation.
- Recommandé en cas de nécessité de partage d'informations entre les utilisateurs de différents magasins de données.

Inconvénients :

- Consommateur en espace disque ;
- Gestion plus complexe.

Ce type d'architecture est celui qui fait référence à l'approche de Inmon [35].
Il revient à l'architecte en charge de la conception du système d'information de décider quel scénario s'adapte le mieux au système décisionnel dont il a la charge de conception. Pour cela il s'appuie sur une

méthodologie de conception, son expérience mais aussi sur une architecture de référence [32] dédiée au domaine du décisionnel.

Dans la section suivante nous détaillons ce que nous entendons par architecture de référence.

2.4.6 L'architecture de référence

Définition d'une architecture de référence :

Une architecture de référence comprend un document ou un ensemble de documents qui contient des recommandations sur les structures et les intégrations de produits et services informatiques dans le but d'établir une solution. L'architecture de référence englobe les meilleures pratiques reconnues dans l'industrie et fait des suggestions sur la méthode de livraison optimale pour telle ou telle technologie.
Les architectes qui conçoivent les systèmes d'information ont à disposition plusieurs architectures de référence selon le domaine qu'ils souhaitent couvrir (système décisionnel, infrastructure, réseaux...etc), leurs préférences, les recommandations de leur société ou bien leurs connaissances et formations.
Si les différentes références architectures peuvent différer par leur formalisme, leur représentation graphique, elles ont toutes le même objectif, celui d'aider l'architecte dans sa conception de solution décisionnelle.
Dans ses travaux de recherche Demchenko [14] explore plusieurs architectures de référence dans le cadre des données massives, dont celle d'IBM mais aussi NIST [54] ou Microsoft [52]. Les architectures de référence de NIST ou de Microsoft sont plus récentes et n'existaient pas lors des premiers systèmes décisionnels, notre choix s'est donc porté sur l'architecture de référence des systèmes décisionnel d'IBM [32] afin de pouvoir suivre l'adaptation des architectures de référence à l'évolution des systèmes décisionnels. Pour plus de simplicité nous appelons l'architecture de référence des systèmes décisionnel, **l'architecture de référence décisionnelle.**

Chaque architecture de référence, possède une certaine sémantique pour décrire ses composants. Dans le paragraphe suivant nous détaillons les composants de l'architecture de référence décisionnelle d'IBM.

L'architecture de référence d'IBM :

L'architecture de référence d'IBM est constituée d'un ensemble de **composants** appelés **SBB**

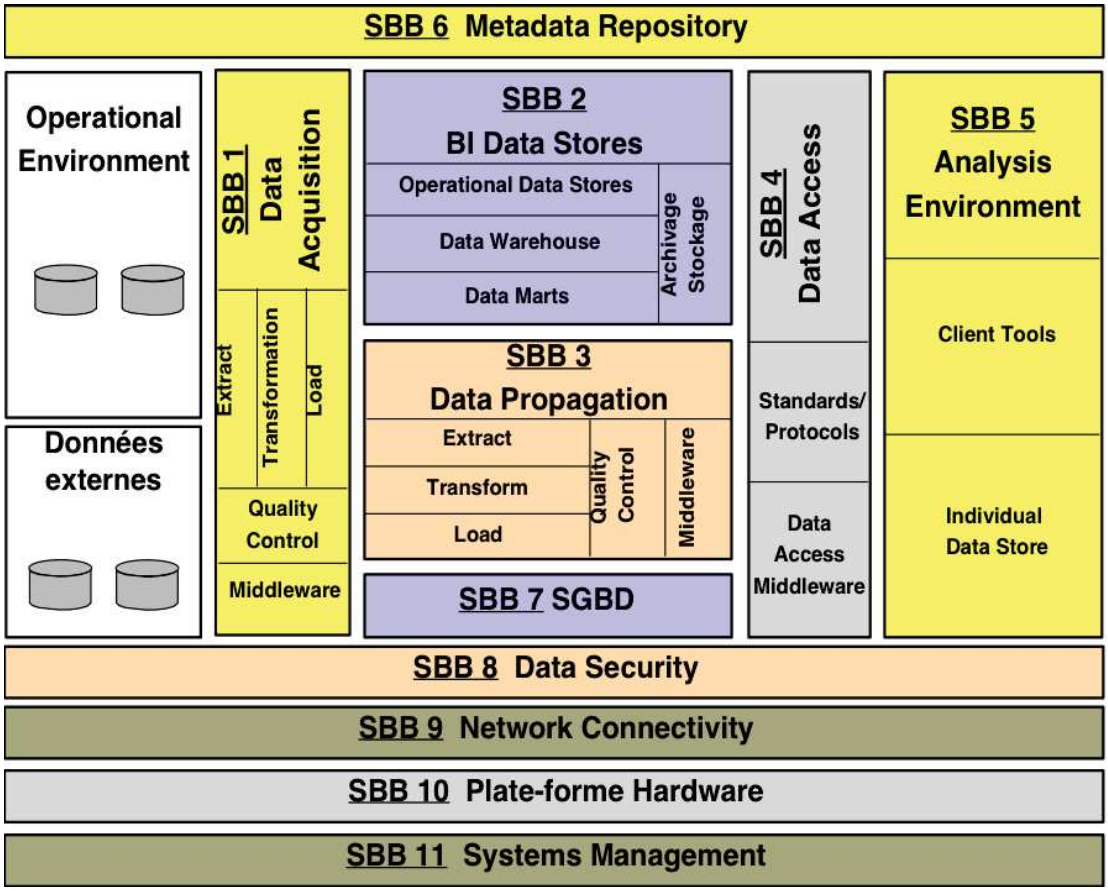


FIGURE *2.8: Architecture de référence décisionnelle - IBM*

(System Building Block) illustrés dans la figure 2.8. La construction de système décisionnel repose sur le modèle d'architecture présenté dans la figure 2.8. Il fournit un cadre permettant de définir précisément les objectifs et le fonctionnement de chacun des composants entrant dans la construction du système décisionnel, et ainsi d'en faciliter la maintenance et les évolutions (prise en compte de nouveaux domaines fonctionnels par exemple ou de nouvelles solutions techniques).

Les composants indiqués sur la figure 2.8 peuvent être répartis en trois sous-ensembles :

— L'environnement de l'entrepôt de données (SBB 1, 2, 3, 4 et 6) ;

— L'environnement d'analyse (SBB 5) ;

— Les services techniques (SBB 7, 8, 9, 10 et 11).

.

L'environnement du système décisionnel (SBB 1, 2, 3, 4, 6)

Les objectifs principaux de ce sous-ensemble sont les suivants :

— L'acquisition des données opérationnelles et des données externes (SBB1) : procédures, programmes et outils "middleware" [8] permettant d'extraire et de sélectionner les données.

— Les transformations et le «nettoyage» des données (SBB1, SBB3) : procédures, programmes et outils "middleware" permettant de fusionner, contrôler, dériver, agréger les données.

— Le chargement des données dans les différents niveaux de stockage des données (SBB1, SBB3) : procédures, programmes et outils middleware permettant de charger les données selon différents modes possibles : «replace», «append», «update or insert», etc ...

— Le transport et la distribution des données entre les différents niveaux de stockage des données (SBB3) ;

— Le stockage des données proprement dit, les procédures de sauvegardes (SBB2) ;

— Les processus d'archivage –ou d'apurement- des données dépassant la profondeur d'historique choisie (SBB2) ;

— La constitution des métadonnées, qui décrit le contenu de l'entrepôt de données (SBB6) sur le plan technique (administration du data warehouse), comme sur le plan métier (guide utilisateurs).

— L'automatisation, le contrôle et le management des flux de données et des processus d'exploitation de l'ensemble de l'environnement de l'entrepôt de données (SBB1, SBB2, SBB3).

— Les interfaces techniques d'accès aux bases d'informations constituant l'entrepôt de données (SBB4) : middleware et passerelles d'accès ;

— Le transport et la distribution des données entre les différents niveaux de stockage des données (SBB3) ;

— Le stockage des données proprement dit, les procédures de sauvegardes (SBB2) ;

— Les processus d'archivage –ou d'épurement- des données dépassant la profondeur d'historique choisie (SBB2) ;

— L'automatisation, le contrôle et le management des flux de données et des processus d'exploitation de l'ensemble de l'environnement de l'entrepôt de données (SBB1, SBB2, SBB3) ;

8. En architecture informatique, un middleware (anglicisme) ou intergiciel est un logiciel tiers qui crée un réseau d'échange d'informations entre différentes applications informatiques. Le réseau est mis en œuvre par l'utilisation d'une même technique d'échange d'informations dans toutes les applications impliquées à l'aide de composants logiciels. Les composants logiciels du middleware assurent la communication entre les applications quels que soient les ordinateurs impliqués et quelles que soient les caractéristiques matérielles et logicielles des réseaux informatiques, des protocoles réseau, des systèmes d'exploitation impliqués.

— Les interfaces techniques d'accès aux bases d'informations constituant le data warehouse (SBB4) : middleware et passerelles d'accès.

Description détaillée : niveaux de stockage (SBB 2)

Le rôle clé de ce sous-ensemble est de constituer, au travers des "éventuels" niveaux de stockage des données, des bases d'informations cohérentes et fonctionnellement fiables permettant aux utilisateurs d'effectuer leurs missions d'analyses sur des sous-ensembles (agrégats) et/ou sur des données détaillées. A chaque niveau peut correspondre un modèle de données spécifique. Selon le contexte et les besoins, les différents niveaux de stockage décrits ci-après seront ou non mis en œuvre (comme décrit dans la section 2.4.5).

— Data Marts (magasins de données) ;

— Le Data Warehouse central (entrepôt de données) ;

— Le réceptacle ou ODS.

Nous avons dans la section 2.4.3 donné les définitions de ces termes, dans l'architecture de référence ils sont considérés comme des "niveaux de stockage".

La propagation des données (SBB 3)

La propagation des données (composant SBB 3) est dans la plupart des cas prise en charge par ce que l'on appelle un outil de type ETL (Extract, Transform and Load), d'extraction de transformation et de chargement des données.

Nous utilisons la définition suivante pour un outil ETL :

Le composant d'ETL (Extract, Transform and Load) a pour but la collecte, la préparation et le chargement des données provenant des systèmes de production en vue de leur stockage dans l'entrepôt de données.

Ce processus s'exécute périodiquement et s'intègre dans l'exploitation informatique de l'entreprise. Les étapes de ce processus sont en général les suivantes :

— L'Etape de collecte des données nécessaires à partir des systèmes opérationnels et de stockage intermédiaire permet d'affranchir la suite du processus des différents rythmes de vie des fichiers de production, et donc de garantir la cohérence dans le temps des données extraites.

— La préparation des données consiste à trier, fusionner ou au contraire diviser les fichiers de données extraites, à effectuer les opérations de nettoyage, de dédoublonnage d'enregistrements, de transformation de formats, de dérivation de valeurs, de concaténation ou d'éclatement des champs de données élémentaires, d'agrégation de valeurs, etc, c'est-à-dire à faire en sorte que les données des systèmes de production deviennent compatibles avec le modèle de données de l'entrepôt.

— L'alimentation consiste à charger les données préparées dans l'entrepôt selon différents modes possibles : remplacement des fichiers cibles, mise à jour des fichiers cibles, ou insertion de nouveaux

enregistrements dans les fichiers cibles ; le choix dépend de besoins fonctionnels comme un souhait d'historisation des données par exemple.

Ce composant peut être réalisé selon diverses méthodes : extractions complètes, extractions incrémentales, réplications, et divers moyens : outils d'extraction, outils de réplication, procédures et programmes développés manuellement.

Il y a en général combinaison de méthodes et de moyens.

Des moyens différents peuvent être choisis, d'une part pour l'alimentation de l'entrepôt central, d'autre part pour celle des magasins de données. Selon les sources de données (tables, fichiers, applications..) ou les tables/fichiers cibles, les critères de choix sont les suivants :

— volumes des sources ou cibles ;

— durée acceptable d'indisponibilité des cibles ;

— taux de mise à jour des sources ;

— capacité du système de transfert de données (réseau, logiciel de transfert) ;

— journalisation des sources ou horodatage des modifications ;

— suppressions logiques ou physiques des enregistrements sources ;

— qualité des données sources ;

— cardinalité des appareillages des sources de données (1-1, 1-N, N-N) ;

— complexité de l'étape de préparation des données.

L'environnement d'analyse (SBB 5)

Cet environnement est composé des moyens, outils et applications mis à la disposition des utilisateurs pour trouver, comprendre, manipuler et analyser l'information contenue dans le système décisionnel. Différentes classes d'outils ou d'applications sont utilisables :

— Requêtes et reporting ;

— Outils d'analyse multidimensionnelle ;

— EIS [9] et tableaux de bord ;

— Fouilles de données ou outils de découvertes.

9. L'Executive Information System (EIS) est un mode de représentation des données décisionnelles, au sein d'un système d'information.

L'objectif fondamental de ce sous-ensemble est de fournir aux utilisateurs les outils et applications dont les fonctionnalités correspondent à leurs besoins d'analyse de l'information et à leur métier.

Les services techniques de base (SBB 7, 8, 9, 10, 11) Les services techniques de mise en œuvre de l'environnement du système décisionnel sont les suivants :

- Le système de base de données (SBB 7) : SGBD Relationnel et/ou multidimensionnel.
- Le système de sécurité d'accès aux informations (SBB 8).
- Le réseau de connexion (SBB 9).
- Les plates-formes hardware et leurs systèmes d'exploitation (SBB 10).
- L'administration de l'ensemble : gestion des évolutions, des performances, des utilisateurs, des incidents, etc (SBB 11).

L'objectif fondamental de ce sous-ensemble est de permettre au système d'information décisionnel d'être :

- Evolutif,
- Disponible et fiable.

Il est intéressant de noter que ce sont ces composants qui prennent de l'importance dans les architectures d'information, en effet la sécurité, la gestion et sauvegarde de gros volumes de données, par exemple, sont de réelles contraintes pour lesquelles l'architecte d'information se doit d'apporter des solutions et donc d'avoir des connaissances, à la fois logicielles mais aussi d'infrastructure informatique.

L'utilisation de cette architecture de référence (quelque soit son créateur) est un outil indispensable à l'architecte. Il lui permet de considérer tous les composants entrant dans la conception d'un système décisionnel, sans rien omettre. Selon les besoins fonctionnels qu'il doit satisfaire et les contraintes auxquelles il est confronté, l'architecte peut décider que certains des composants de l'architecture de référence sélectionnée ne seront pas mis en place. La raison doit être explicitée et permet une traçabilité des choix qui ont été pris. L'architecte justifie ses arbitrages au travers d'un document de "décision d'architecture" (*architectural décision*) qui permet de comprendre ses choix.

L'évolution des systèmes décisionnels entraîne une évolution des architectures d'information qui les supportent et donc des architectures de références. La description détaillée de cette architecture de référence nous sert de repère pour mieux appréhender l'impact de l'évolution des systèmes décisionnels sur le système d'information.

Dans ce chapitre nous avons étudié les systèmes décisionnels du prisme de l'architecture , introduit les principaux composants d'un système décisionnel que sont le réceptacle (ou ODS), l'entrepôt de données (ou Data warehouse) et les magasins de données (ou Data mart).

Dans le chapitre suivant nous détaillons les facteurs qui ont déclenché l'évolution des architectures des systèmes décisionnels.

Chapitre 3

L'évolution des systèmes décisionnels

Les systèmes décisionnels n'ont cessé d'évoluer, se démocratiser et de s'adapter ces dernières années. C'est notamment l'explosion des volumes des données émises (voir figure 2.1) qui accélère cette évolution des systèmes décisionnels (et qui a retenu notre attention). Les marchés du logiciel et des infrastructures matérielles ont aussi contribué à l'évolution des systèmes décisionnels permettant d'accéder, de traiter, de transformer, de stocker, d'exploiter, de visualiser et d'analyser les données de façon plus rapide, plus simple et plus ouverte au point de vue technologique. Cette révolution technologique a permis de rendre les systèmes décisionnels accessibles à un plus grand nombre d'utilisateurs, accélérant leur démocratisation à toute la population d'une organisation et non plus seulement à des utilisateurs avertis.
Dans ce chapitre nous avons recensé, sous l'influence des données massives, les éléments des systèmes décisionnels sujets à évolution tels que :

— l'évolution des logiciels ;

— l'évolution des infrastructures ;

— l'évolution des données ;

— l'évolution des usages ;

— l'évolution de la modélisation.

— la technologie Apache Hadoop

Nous allons détailler les évolutions des systèmes décisionnels relatives à ces divers facteurs dans les paragraphes qui suivent.

3.1 L'évolution des logiciels décisionnels

Chaque composant d'un système décisionnel (voir figure 2.4) est couvert par une couche ou plusieurs couches logicielles.

FIGURE *3.1: Magic Quadrant Gartner des outils d'intégration- outils E.T.L pour 2018*

3.1.1 Acquérir

L'étape d'acquisition des données est couverte par des outils de types *E.T.L* (Extract Transform Load). Ces outils permettent d'acquérir, transformer, nettoyer et charger les données dans les différentes couches de l'architecture décisionnelle [79][7]. Sur la figure 2.4, cette étape est caractérisée par les flèches qui vont d'un composant à un autre. En effet des extractions de données ou des transformations (par exemple) sont nécessaires à chaque construction de composant, qui font appel à ces outils E.T.L.

L'offre du marché des logiciels de type E.T.L est assez étendue, comme l'illustre le graphe 3.1, qui représente un "magic quadrant" [1] du Gartner [2].
Des sociétés telles qu'*IBM*, *Informatica* [3], *Talend* [4] peuvent être citées comme réfèrentes sur le marché des outils E.T.L, dénommés aussi outils d'intégration de données. Le tableau 3.1 fait un état des lieux des acteurs majeurs de l'offre logicielle des outils d'intégration en 2018. Afin d'évaluer l'évolution de ce marché logiciel, il est intéressant de comparer cet état des lieux en 2018, à celui de 2008, comme le

1. le Magic Quadrant du Gartner est une trame graphique de cinq carrés qui représente sous forme de nuage de points le positionnement et la performance des acteurs qui vendent des produits dans une catégorie commune.
2. Gartner Inc. est une entreprise américaine de conseil et de recherche. Elle mène des recherches, fournit des services de consultation et tient à jour notamment différentes statistiques.
3. Société mondiale, dans la gestion des données d'entreprise :https ://www.informatica.com
4. Société mondiale, dans la gestion des données d'entreprise :https ://fr.talend.com/

FIGURE *3.2: Comparaison des positions des acteurs du marché des outils d'intégration en 2018 et en 2008*

montre les deux graphes réunis dans la figure 3.2.
Lorsque l'on compare ces deux graphes de positionnement des acteurs sur le marché des outils d'intégration de données on remarque une stabilité des acteurs principaux, déjà présents dix ans plus tôt sur ce marché. Cela indique une adaptation aux usages, aux données et aux infrastructures qui ont eu lieu ces dix dernières années.
On remarque aussi que les outils provenant du marché "libre" (*monde open*), sont ceux qui ont le plus progressé (Talend) dans ce marché, signe de l'intégration de nouvelles technologies et pratiques dans les architectures des systèmes décisionnels.

L'adaptation aux usages, et donc aussi aux utilisateurs, d'outils d'intégration est l'un des facteurs les plus marquant dans l'évolution de l'offre logicielle des outils d'intégration ces vingt dernières années. En effet lorsque l'on compare ne serait-ce que la partie interface de ces outils on peut comprendre leur adaptation massive dans les architectures des systèmes décisionnels, et leur rôle d'influence dans la mise en place massive de ces systèmes, dans les organisations.
En effet d'outils réservés dans un premier temps à des profils très avertis techniquement, ils ont su évoluer en proposant des interfaces graphiques, très simples, qui rendent leur utilisation plus aisée pour des débutants en informatique décisionnelle. Certes les utilisateurs de ces outils restent des profils techniques mais ils sont plus accessibles et ne demandent pas la maîtrise d'un langage informatique

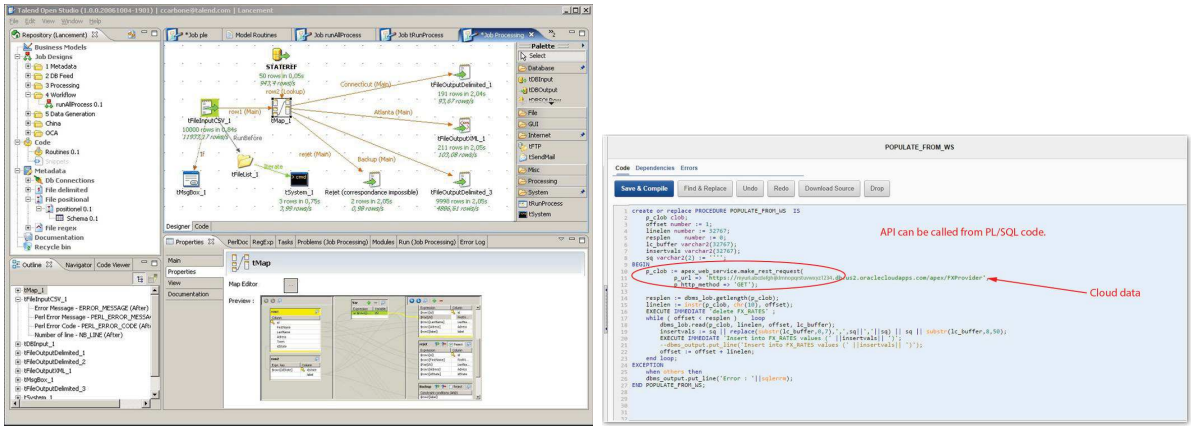


FIGURE 3.3: *Comparaison Interface graphique d'un outil ETL et du code pur*

spécifique.

La figure 3.3 illustre cette évolution au niveau de l'interface. La partie de droite montre l'interface, très orientée "écriture de code informatique" et similaire à une interface de développement de programme informatique, et celle de gauche qui elle est plus intuitive, basée sur des déplacement graphique de type "glisser-déplacer" et beaucoup plus visuelle (outil E.T.L Talend).

L'adaptation de ces outils d'intégration de données s'est faite aussi au niveau technologique. Ils ont su enrichir leur offre de connexions aux différents progiciels, sources de données et infrastructures informatiques (les différents types de serveurs par exemple) présentes dans les organisations, rendant ainsi accessibles et intégrables tous les types de données dont disposent les organisations.

L'évolution la plus marquante concernant les outils d'intégration de données sont donc leur interface graphique, la facilité d'utilisation, leur offre très large en matière de connexions de données et leur grande résistance dans le temps. Le monde "libre" semble toutefois venir déranger l'ordre établi des acteurs historiques du marché.

Ces outils, au-delà de leur pérennité technologique dans le temps, sont un élément crucial, dans la création des systèmes décisionnels car c'est au travers d'eux que s'établissent les règles et contrôles des données qui font de l'information qui en dérive une information de qualité et de confiance.

Lorsque les données sont intégrées via ces outils d'intégration, dans les systèmes décisionnels, elles doivent être stockées, au travers des différents composants du système décisionnel. La section suivante détaille cette étape, son évolution et l'offre logicielle qui en découle.

FIGURE *3.4: Position des acteurs du marché des plateformes analytiques en 2018, Gartner magic quadrant.*

3.1.2 Stocker

Dans la figure 2.4, les zones de stockages des données sont matérialisése par les composants : ODS, Entrepôt de données et magasins de données. Pour ces trois différentes zones, des solutions logicielles existent.

À la naissance de l'informatique, plusieurs modèles de stockage de l'information sont explorés, comme les modèles hiérarchique ou réseau. Mais c'est finalement le modèle relationnel qui l'emporte dans les années 1970 car c'est lui qui permet de mieux assurer le contrôle de l'intégrité des données, grâce à un modèle théorique puissant et simple.

Par ancienneté, mais aussi parce que cela reste encore le système de stockage le plus répandu ce sont les systèmes de gestion de bases de données relationnelles ou RDMS[5] qui stockent les données provenant des systèmes sources pour le système décisionnel. Les acteurs les plus présents sur le marché sont IBM, Oracle et Microsoft.

Comme première solution et évolution dans le stockage des données du système décisionnel nous

5. Relational Database Management Systems

Bases	Année de lancement	Schéma de données	Editeur / prestataire de support	Positionnement
Cassandra	2008	Orienté colonnes	DataSax (ex Riptano)	Adoptée par les géants du web et les start-up, Cassandra permet de gérer de gros volumes de données.
Couchbase	2010	Orienté documents	Couchbase	A la différence de MongoDB, Couchbase dispose d'un outil de requêtage normalisé SQL qui facilite sa prise en mains par des développeurs rompus aux bases SQL.
Elasticsearch	2004	Index inversé	Elasticsearch	Connu avant tout pour son moteur de recherche distribué, Elasticsearch tire sa force de l'indexation et de l'analyse des données.
HBase	2006	Orienté colonnes	Hortonworks	Souvent comparé à Cassandra, HBase joue la carte de la très forte volumétrie. Un produit complexe qui exige un gros travail de structuration.
MongoDB	2007	Orienté documents	MongoDB	Base NoSQL la plus populaire, MongoDB est saluée pour la souplesse de sa structure et sa capacité à répondre à un grand nombre de besoins.
Redis	2009	Clé-valeur	Redis Labs	Base de données en mémoire, Redis privilégie la vitesse d'exécution. En contrepartie, ses capacités de requêtage sont limitées.
Riak	2009	Clé-valeur	Basho Technologies	Riak se présente comme une sorte de Redis évolué en étendant les capacités de requêtage via des index secondaires.

FIGURE *3.5: Tableau de comparaison des bases de données NoSQL [16]*

trouvons les logiciels de stockage dit propriétaires, appelés aussi des "appliance" [74]. Une "appliance" décisionnelle intègre par conception les éléments matériels, serveur, stockage, mémoire mais aussi modélisation des données et gestion de l'ensemble des données. L'appliance Teradata [74] est l'une des plus anciennes appliances décisionnelles, qui est toujours leader sur son marché (voir figure 3.4). On y trouve à ses côtés des acteurs majeurs du marché des plateformes analytiques (autre nomination du Gartner pour regrouper les acteurs du marché des appliances décisionnelles) tels qu'IBM, Oracle ou Microsoft.

Les bases de données de type NoSQL [6] font elles aussi, depuis les année 2000, parties des modes de stockages possibles pour un système décisionnel.
NoSQL signifie Not Only SQL et non pas No SQL, il s'agit de compléments aux SGBDR pour des besoins spécifiques et non de solutions de remplacement. Carlo Strozzi a utilisé le terme NoSQL ou Not Only SQL en premier en 1998 pour désigner la base de données relationnelle "Open Source" qu'il a développée et qui ne disposait pas d'une interface SQL comme ses homologues. Carlo Strozzi a proposé par la suite

6. En informatique et en bases de données, NoSQL désigne une famille de systèmes de gestion de base de données (SGBD) qui s'écarte du paradigme classique des bases relationnelles. L'explicitation du terme la plus populaire de l'acronyme est Not only SQL (« pas seulement SQL » en anglais) même si cette interprétation peut être discutée.

de changer le terme NoSQL en NoRel pour non-relationnelles, vu que ce mouvement a convergé avec le temps vers les bases de données non-relationnelles uniquement. En 2009, le terme NoSQL a été réintroduit par Eric Evans à une échelle plus large, décrivant les nombreuses bases de données s'opposant à la notion relationnelle et possédant les caractéristiques suivantes :

— Elles sont toutes compatibles avec les systèmes distribués ;

— Elles sont de type Open Source ;

— Elles sont de type non-relationnel.

NoSQL est un type spécifique de bases de données, permettant de stocker et de récupérer les données après restructuration, en utilisant des techniques différentes de celles connues dans les bases de données relationnelles. Les développeurs de nos jours ont tendance à utiliser ce type de bases de données pour la simplicité de leur implémentation et leur évolutivité sans limite (horizontalement, à travers de nouvelles colonnes). On distingue quatre catégories de bases de données NoSQL :

— La base orientée clé-valeur. Sur le principe d'un dictionnaire dont la porte d'entrée est le mot, les bases orientées clé-valeur vont accéder à une valeur de données unique. Cela peut être un "compte client" pour valider un panier sur un site en ligne. En plaçant ces données en mémoire, ces bases sont recherchées pour leur vitesse d'exécution. En revanche, elles ne permettent pas de faire des requêtes multicritères sophistiquées.

— La base orientée documents. Évolution des bases clé-valeur, les moteurs orientés documents n'associent plus une clé à une valeur mais à un document dont la structure reste libre. Pour cela, ils s'appuient sur le très populaire format d'échange de données Json (JavaScript Object Notation). Les bases orientés documents sont souvent saluées pour la souplesse de leur structure.

— La base orientée colonnes. Contrairement aux moteurs orientés documents à la structure libre, les bases en colonnes stockent les données par colonnes. Cette structure permet en outre d'ajouter plus facilement une colonne à une table. Ces bases sont plébiscitées pour leur capacité à monter en charge et à accueillir une forte volumétrie de données.

— L'index inversé. Popularisé par le moteur de recherche de Google, les index inversés sont représentés dans le tableau 3.5 par ElasticSearch. Basésur la bibliothèque d'indexation open source Lucene et le format Json, ElasticSearch permet de structurer les données à la manière d'une base orientée documents tout en profitant d'excellentes capacités de requêtage.

Le tableau 3.5 synthétise et compare les différentes bases de données NoSQL et leurs principaux avantages. Selon les besoins en terme d'architecture (la vitesse, la volumétrie, la recherche multicritères, par exemples) la sélection s'orientera vers l'une ou l'autre des grandes familles de base NoSQL.

On peut citer aussi le NewSQL, qui est un stockage distribué et potentiellement entièrement en mémoire et pouvant être requêté classiquement par une interface SQL. NewSQL est tiré du monde NoSQL mais reste différent. Comme NoSQL il s'agit d'une nouvelle architecture logicielle qui propose de repenser le stockage des données. Elle profite des architectures distribuées, des progrès du matériel et des connaissances théoriques depuis 35 ans. Mais contrairement à NoSQL elle permet de conserver le modèle relationnel au cœur du système. Dans ses travaux Hashem [27] présente le modèle de données NoSQL et sa dérivée NewSQL, en termes d'architecture, d'avantages et de limitations. D'après ses travaux, cette technologie reste encore peu usitée dans les systèmes décisionnels.

De la même façon, sans vouloir être exhaustifs, nous pouvons aussi citer les bases de données de type graph (i.e orientées objet utilisant la théorie des graphes, donc avec des nœuds et des arcs) qui peuvent elles aussi être exploitées comme moyen de stockage des données.

D'autres modes de stockage, dont Apache Hadoop[7], sont eux aussi utilisés. Nous y consacrons une section (3.6) dans ce chapitre. Le mode "dans les nuages" (Cloud) est aussi un mode de stockage possible pour les systèmes décisionnels, même si techniquement il va être basé sur une ou plusieurs des technologies que l'on énumère dans ce paragraphe.

Les architectes d'information ont donc à disposition plusieurs modes de stockages possibles pour la conception du système décisionnel : RDMS, NoSQL, Apache Hadoop, Les nuages, etc. Ces modes ne sont pas exclusifs, peuvent tout à fait cohabiter entre eux, c'est d'ailleurs l'évolution la plus importante dans les systèmes décisionnels : **l'hybridation des modes de stockages.**
On va pouvoir en effet adapter exactement le support de stockage de données du système décisionnel aux exigences fonctionnelles et non fonctionnelles en terme d'architecture. On peut donc imaginer un ODS sur un support Apache Hadoop, avec un entrepôt de données sur un SGBD relationnel et des magasins de données utilisant des bases NoSQL, dans les nuages et sur des appliances. La figure 3.6 donne une illustration de ce que ce mode de stockage hybride peut donner sur une architecture décisionnelle.

L'évolution technologique des modes stockages permet aux architectures des systèmes décisionnels d'évoluer et de s'adapter aux différents formats de données désormais disponibles, offrant ainsi de nouvelles possibilités d'extraction d'information à ses utilisateurs. Pour en tirer parti les logiciels permettant de délivrer cette information doivent eux aussi évoluer. Le paragraphe suivant expose ces évolutions autour de la visualisation, l'analyse et l'exploration des données dans les systèmes décisionnels.

7. Hadoop est un framework libre et open source écrit en Java destiné à faciliter la création d'applications distribuées et échelonnables permettant aux applications de travailler avec des milliers de nœuds et des pétaoctets de données

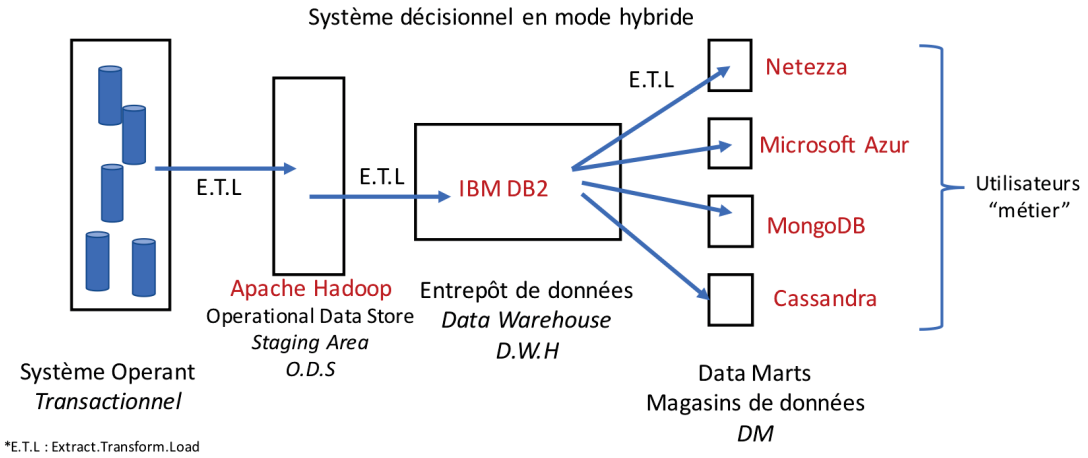


FIGURE *3.6: Système décisionnel avec une architecture hybride des stockages.*

3.1.3 Exploiter

L'exploitation des données transformées en information est l'objectif premier de la mise en place des systèmes décisionnels. On peut classer en trois catégories l'exploitation des données au sein des systèmes décisionnels.

— La production de rapport récurrents (le *reporting*) ;

— L'exploration manuelle ;

— L'analyse de données.

Le principe du *reporting* est d'agréger et de synthétiser des données nombreuses et complexes sous forme d'indicateurs, de tableaux, de graphiques permettant d'en avoir une appréhension globale et simplifiée. Il s'appuie principalement sur les agrégats (GROUP BY en SQL par exemple) afin de faire apparaître des comptages, sommes ou moyennes en fonction de critères d'analyses. Le reporting est généralement récurrent, le même rapport sera produit à intervalles réguliers pour contrôler les variations des indicateurs.
Les nombreuses réglementations pour lesquelles les organisations doivent rendre compte, par exemple les normes de Bâles pour les organismes financiers Européens, nécessitent la mise en place de *reporting* mensuel pour les auditeurs. Cet usage de "reporting" est un des moyens d'exploitation des informations les plus répondus dans les organisations industrielles.

L'exploration manuelle est une autre exploitation des données, en contexte décisionnel, qui consiste à pouvoir explorer les données de façon peu dirigée (heuristique) afin de trouver des réponses à des

questions que l'on ne s'est pas encore posé. C'est un espace de "liberté", dans un environnement métier au préalable défini où les utilisateurs (en général "avertis") peuvent se constituer leur propres tableaux de bord ou rapports d'analyse. Par exemple, dans une organisation industrielle, les contrôleurs de gestion peuvent analyser leurs données au travers de différents angles ou point de vue et se constituer leurs propres rapports ou tableaux de bord.

L'exploration de données s'appuie sur des outils permettant de manipuler et de visualiser les données selon des requêtes dynamiquement produites par des utilisateurs experts du domaine.

L'Analyse de données est plus dense en terme de possibilités et est la façon d'exploiter les données qui a le plus évolué au cours de ces vingt dernières années. Nous détaillons cette évolution dans la section 3.4 qui décrit l'évolution des usages des données dans les systèmes décisionnels. On peut regrouper les possibilités d'analyse des données sous quatre familles qui donnent lieu à quatre familles d'outils :

— descriptifs ;
— diagnostics ;
— prédictifs ;
— prescriptifs.

Ces outils regroupent les différents usages de l'information dans une organisation. Dans la section 3.4, nous détaillons ces quatre types d'analyses et les outils associés.

L'évolution de l'utilisation des données contribue à l'expansion des systèmes décisionnels, l'intégration de nouvelles données de façon régulière accentue le volume de données disponibles dans ces systèmes. Une mémoire, au travers de l'historique de ces données est constituée au sein de ces systèmes. Les notions de données actives (récentes ou chaudes) ou de données moins actives (froides, par opposition) apparaît. Le capital des données d'une organisation se constitue donc aussi au travers de la conservation des données dans ses systèmes décisionnels.
La notion de conservation et donc d'archivage de ces données devient aussi un élément d'architecture dont il faut tenir compte dans l'évolution des systèmes décisionnels.
Le paragraphe suivant traite de cette notion d'archivage, qui donne naissance à de nouveaux usages, que nous détaillons dans la section 3.4.

3.1.4 Archiver

L'utilisation et la maturité des systèmes décisionnels conduisent à la constitution d'un historique de données important. Cet historique est la clé de l'évolution des usages des systèmes décisionnels, notamment au travers l'analytique prédictive (voir section 3.4) qui n'est rendue possible que si un historique des

données, fiable, est disponible. C'est donc un facteur important d'évolution des systèmes décisionnels.
Le défi de l'archivage de ces données est à la fois sur sa profondeur de conservation et sur son état (c'est-à-dire "en ligne", à disposition des utilisateurs), qui impose un accroissement important du volume des données à conserver mais qui est aussi potentiellement un facteur de ralentissement technique lors de l'accès aux données.
En effet le volume à explorer étant plus important, les requêtes d'exploitation des données peuvent être plus lentes à fournir des résultats.
Selon les cas d'usage des organisations il peut y avoir constitution de ces archives sur des supports techniques différents, par exemple, de ceux de l'entrepôt de données tout en restant accessibles pour certains utilisateurs ayant besoin d'accéder à ces données dites "froides" (par comparaison aux données "chaudes", qui sont plus d'actualité). La gestion de l'archivage des données fait partie de la gouvernance des données qui doit être mise en place dans un système décisionnel. C'est la gestion du cycle de vie des données qui y fait référence.
Ce point de gouvernance des données est traité au travers des composants des architectures de référence qui imposent aux architectes des systèmes d'information de traiter cette problématique et d'y affecter une solution. Dans l'architecture de référence présentée en section 2.4.6, c'est le SBB 11 qui traite de cette problématique.
L'archivage des données devient plus aisé lorsque que l'on peut tirer parti des évolutions des infrastructures des plateformes techniques qui supportent les systèmes décisionnels.
Le paragraphe suivant aborde les évolutions technologiques d'infrastructure dont les architectures des systèmes décisionnels peuvent tirer parti et aborder par exemple l'accès de façon rapide à un grand volume de données.

3.2 L'évolution des infrastructures- introduction du concept HTAP

L'évolution technologique des serveurs, au niveau de la puissance, de la mémoire et dans un même temps de la diminution du prix d'achat, permet une évolution des infrastructures qui supportent les systèmes décisionnels et de nouvelles solutions peuvent être envisagées pour répondre aux besoins non fonctionnels.
Ces besoins non fonctionnels peuvent correspondre à la diminution du temps d'accès à l'information que ce soit en terme d'accélération d'exécution des requêtes (lors de la phase d'exploitation), ou à la réduction du temps d'acquisition des données, dans la phase d'acquisition des systèmes décisionnels. Les systèmes décisionnels peuvent être alors considérés comme plus "dynamiques" [8].
Dés 2007, le terme de *Dynamic Data Warehouse, ou DDW* apparaît, cela correspond à une évolution

8. http ://www.dataversity.net/the-dynamic-data-warehouse/

des architectures des systèmes décisionnels qui prennent en compte ces nouvelles capacités techniques des infrastructures, telles que le temps réel ou les calculs en mémoire (*in memory*).
Ces deux grandes évolutions technologiques ont des effets immédiats sur les conceptions d'architecture des systèmes décisionnels. Nous décrivons plus en détails ces impacts dans les sous-sections suivantes.

3.2.1 Le temps réel

La puissance des ordinateurs et des logiciels entraînent une rapidité des traitements des données qui peuvent désormais être réalisés en temps réel. Mettre à disposition, en quasi temps réel des données produites, permet de disposer d'information plus "fraîche" et servir les besoins fonctionnels de certains utilisateurs en ayant la nécessité.
Après avoir été dans une phase de savoir qu'il s'était passé quelque chose (le «reporting»), puis une phase de savoir pourquoi il s'était passé quelque chose (l'analyse) c'est bien une nouvelle étape que d'essayer d'avoir accès à l'information au moment où elle se produit et donc pouvoir prendre une décision dite en temps réel ! Si ce nouveau besoin n'est pas un pré requis pour la plupart des utilisateurs mais pour une poignée d'entre eux, avoir la capacité technologique de le mettre en oeuvre nécessite une forte évolution des architectures décisionnelles.
Dans ce cadre-là c'est bien un défi, non plus aux logiciels auxquels il est demandé de répondre mais bien au système lui-même et c'est dans son infrastructure que se trouve la solution. En effet c'est la composante d'architecture technique [70], dont l'infrastructure dépend, que l'on doit traiter : être capable de délivrer une information, fiable et sécurisée, en temps réel.
Pour cela, il va y avoir remise en question de l'architecture technique du système décisionnel : on va *rapprocher* les données de l'endroit où elles sont produites (les systèmes opérationnels, par exemple) de l'endroit où elles sont transformées en information (dans les systèmes décisionnels). Cela constitue une révolution dans la conception des architectures techniques des systèmes décisionnels.
En effet par tradition le système décisionnel est construit, la plupart du temps sur des infrastructures techniques différentes de celles des systèmes opérationnels. Ces infrastructures ayant chacune des exigences techniques qui n'étaient pas compatibles entre elles, dans le passé. Cependant les évolutions technologiques de certains serveurs permettent d'envisager cette cohabitation au niveau de l'architecture. En 2014, le Gartner nomme de telles architectures, des architectures HTAP [9] [23]..

C'est une architecture dite hybride, qui permet de traiter sur les mêmes ressources physiques et logicielles, à la fois des requêtes de type OLAP (provenant des systèmes décisionnels) et des transactions OLTP (provenant des systèmes opérationnels), a priori incompatibles.
Cette architecture de type HTAP implique donc un partage d'infrastructure entre les données opéra-

9. Hybrid Transactional Analytical Processing

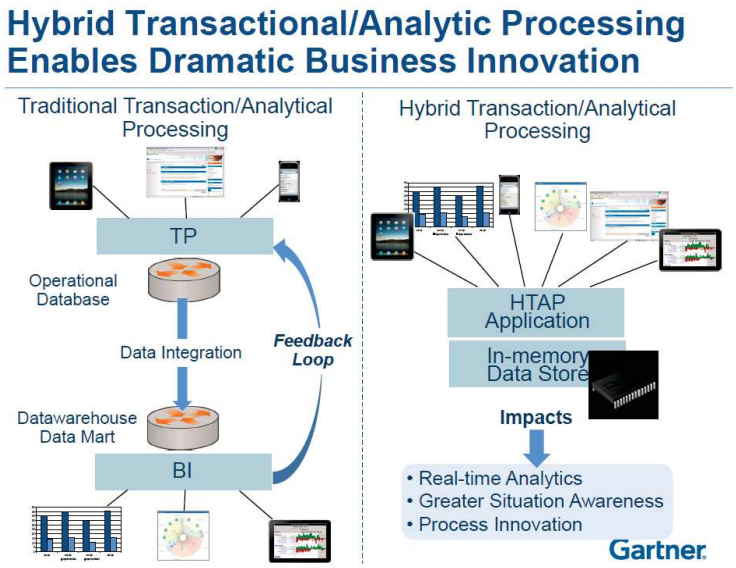


FIGURE *3.7: Description d'une architecture de type HTAP par le Gartner*

tionnelles et décisionnelles en vue de mettre à disposition les données opérationnelles le plus rapidement possible pour réduire le temps de délivrance de l'information. Cela permet aussi une simplification de l'infrastructure et une capitalisation des compétences et ressources existantes.
La partie gauche du schéma de la figure 3.7 illustre l'architecture classique d'un système décisionnel où les données vont quitter l'environnement où elles sont produites (les systèmes transactionnels) pour alimenter le système décisionnel. La partie droite montre la fin de ce déplacement des données à l'extérieur de l'environnement où les données sont produites.

Les avantages d'une architecture HTAP pour les systèmes décisionnels :

— une simplification de son architecture d'information, au plus près de ses données opérationnelles ;

— une accélération dans la délivrance de l'information aux utilisateurs (tant en acquisition qu'en restitution) ;

— une mise à disposition des dernières techniques d'analytiques avancées pour répondre aux nouveaux besoins des utilisateurs ;

— une possibilité de mettre en œuvre des processus complexes d'analyses prédictives en temps réel ;

— une mise à disposition des informations au niveau le plus fin, celui des données transactionnelles ;

— une garantie et une amélioration dans la gouvernance des données (qualité, par élimination de multiples copies des données) ;

— une ouverture à de nouveaux usages permettant l'application de fonctions analytiques, en temps réel, sur des données opérationnelles (la détection de fraude bancaire en temps réel, par exemple) ;

— une colocation des données opérationnelles et décisionnelles gage de fraîcheur et qualité des données ;

— une disponibilité, sécurité, évolutivité et agilité.

Le temps réel, au travers de la mise en place des architectures HTAP notamment permet donc une évolution des architectures des systèmes décisionnels, lorsque les données utilisées par le système décisionnel proviennent **d'un seul système transactionnel**.
Dans la section suivante nous décrivons l'exploitation des améliorations de la mémoire des serveurs, accentuée par la réduction de ses coûts d'acquisitions.

3.2.2 En mémoire

La technologie *in-memory ou IMDB (In Memory DataBase)* ou MMDB (Main Memory DB) consiste à charger des données dans la mémoire vive (RAM) du système informatique afin d'accélérer l'exécution des requêtes qui exploitent ces données. Elle repose sur une infrastructure obligatoirement performante, avec des serveurs puissants et une mémoire vive (RAM) importante, un système d'exploitation et des solutions adaptés.
Utiliser la mémoire pour accélérer les applications très gourmandes en entrées-sorties n'est pas une idée nouvelle, un traitement de données en mémoire est plus rapide (parfois 1000 fois plus) qu'un traitement qui nécessite d'attendre l'écriture et la lecture sur un support plus lent, y compris le stockage flash que nous détaillons dans la section 3.2.4. Depuis les débuts de l'informatique, les solutions positionnées sur la haute performance ont alloué de la mémoire pour cacher les données. La plupart des bases de données ont été conçues pour utiliser la mémoire autant que possible. Aujourd'hui, les traitements en mémoire exploitent ce principe à l'extrême : utiliser la mémoire dynamique (Dynamic RAM) pour l'exécution du code de la base et les structures de données actives, et conserver en mémoire la base persistante. Ces bases de données évitent autant que possible de dialoguer avec des supports de stockages internes ou externes. Au lieu de cela, elles optimisent leurs structures de données pour les traitements en mémoire.
Historiquement, la densité de mémoire disponible par serveur et le coût relativement élevé de la mémoire vive ont limité ce type d'approches, mais aujourd'hui la technologie permet d'appliquer les technologies en mémoire sur des jeux de données de taille bien plus importante. Les serveurs disposent de bien plus de mémoire vive, la déduplication "inline/online" et la compression tirent parti de la puissance des "CPU" modernes pour placer plus de données en mémoire et enfin l'usage de technologie de cluster[10] permet

10. Dans un système informatique, un agrégat, ou « cluster », est un groupe de ressources, telles que des serveurs. Ce groupe agit comme un seul et même système. Il affiche ainsi une disponibilité élevée, voire, dans certains cas, des fonctions

de distribuer les données sur un nombre accru de nœuds disposant eux-mêmes de plus de mémoire.
Les barrettes mémoire sont de moins en moins chères et de plus en plus denses. Les ordinateurs portables modernes sont désormais livrés avec autant de mémoire qu'un ancien mainframe. Aujourd'hui, n'importe qui avec une simple carte de crédit peut louer des serveurs à forte capacité chez les fournisseurs du nuage comme Amazon Web Services (ses instances R3 sont équipées de 244 Go de RAM pour 2, 80$ l'heure). Des serveurs en rack standards comme ceux proposés par HP, Dell, Lenovo ou Fujitsu peuvent héberger de 4 à 6 To de RAM. Et un serveur Unix haut de gamme comme le système Oracle M6-32 peut gérer jusqu'à 32 To de mémoire.
Avec les bases de données transactionnelles traditionnelles, seules les données les plus sollicitées sont placées en mémoire pour optimiser les performances, tandis que des formes de stockage moins coûteuses sont utilisées pour des données plus froides. Le moteur d'une base a pendant longtemps reposé sur ce principe. Aujourd'hui, placer l'ensemble de la base en mémoire peut faire sens.
Les vendeurs de solutions logicielles pour les systèmes décisionnels ne s'y trompent pas et investissent massivement dans ces technologies. Sur le marché, SAP HANA [11] a ouvert le bal. Cette base *In-Memory*, dotée de certaines capacités de passage à l'échelle, est conçue pour héberger les données critiques de l'entreprise. Elle peut fournir des rapports analytiques en temps réel, ou presque, et elle peut également être utilisée pour d'autres formes de données. Les rapports, dont la création prenait des heures avec des bases de données transactionnelles, peuvent ne prendre que quelques minutes avec HANA (d'après SAP).
Les autres fournisseurs de plateformes analytiques (voir figure 3.4), comme Teradata ou HP Vertica, se battent pour optimiser au maximum l'usage de la mémoire. Ils ont également la capacité d'analyser d'importants jeux de données, trop volumineux pour des outils In-Memory. Vertica, par exemple, propose une approche hybride de l'In-Memory qui permet de charger rapidement les données en ayant recours à la mémoire, pour un accès en (presque) temps réel à la fois aux données sur disques et en mémoire.
Oracle Hybrid Columnar Compression (HCC) est un bon exemple de compression de données transactionnelles en fonction de leur date de création, tout en accélérant les capacités d'analyse. A l'automne dernier, Oracle a annoncé une option de traitement de mémoire qui conserve les données dans un format HCC pour accélérer l'analytique, et les lignes transactionnelles en mémoire pour accélérer en parallèle les traitements transactionnels l'OLTP. Récemment, Microsoft a aussi optimisé SQL Server pour que son SGDB puisse supporter davantage de transactionnel en mémoire. IBM a lui aussi doté son SGBD Db2 de capacités de traitement in-memory.
Si ces acteurs majeurs du marché décisionnels font évoluer leur technologie c'est que le *In Memory* est bien une avancée technologique, qui fait évoluer les architectures des systèmes décisionnels.

---

de traitement en parallèle et d'équilibrage de la charge. On parle ainsi de gestion en cluster (clustering).
11. SAP HANE : https ://www.sap.com/france/products/hana.html

Notons cependant que le *In Memory* ne dispense pas de l'usage d'une infrastructure de stockage classique, qui conserve l'ensemble des données de l'entreprise. C'est donc bien une technologie complémentaire que l'on peut utiliser dans les architecture des systèmes décisionnels si les exigences techniques le nécessitent.

Ces deux évolutions technologiques majeures que sont la possibilité d'intégrer du temps réel dans les systèmes décisionnels, d'accélérer les temps d'exécutions des requêtes, ou bien de réduire le temps d'acquisition des donnés à intégrer, impulsent des évolutions sur les architectures techniques qui tirent parti désormais de ces diverses technologies et deviennent plurielles techniquement. L'hybridation technique est désormais inéluctable. Le paragraphe suivant détaille cette conséquence de l'évolution technologique des infrastructures des systèmes décisionnels.

3.2.3 Hybridation

Sous l'influence des innovations technologiques vues dans les paragraphes précédents, le système décisionnel n'est plus mono technologique (un seul système de stockage ou serveurs par exemple), mais bien pluriel. A ces innovations technologiques s'ajoutent l'influence du "cloud" (technologie dans les nuages) privé ou public et de la virtualisation. Selon certaines contraintes des organisations, telles que le coût par exemple, il peut être imposé d'incorporer ces techniques dans les architectures des systèmes décisionnels, renforçant le côté hybride de leur architecture.
Les résultats attendus sont d'améliorer l'efficience des serveurs, de réduire l'empreinte environnementale de l'infrastructure, de favoriser l'automatisation et les nouveaux modèles de prestation de services et, point essentiel, de rendre les entreprises plus agiles.
La virtualisation des serveurs quant à elle a un impact positif sur leur consommation énergétique et permet aux organisations de regrouper voire de rationaliser quelques centres de données. Le temps de provisionnement des serveurs baisse et de nouveaux serveurs peuvent souvent être mis en place en l'espace de quelques minutes.
Ces deux facteurs, "cloud" et virtualisation se rajoutent aux facteurs d'innovation technologique pour influencer, via l'aspect technique, l'évolution des systèmes décisionnels. L'aspect stockage des données quant à lui n'est pas en reste en terme d'évolution et même de révolution technique.
Dans le paragraphe suivant nous abordons le stockage "flash" qui révolutionne lui aussi les architectures décisionnelles, lorsqu'il y est intégré.

3.2.4 Le stockage Flash

Le support technique de stockage des données d'un système décisionnel, est souvent peu considéré lors de la conception d'architecture. Dans l'architecture de référence que nous avons détaillée dans la section 2.4.6 c'est le SBB10 qui traite du sujet, mais il n'existe pas de SBB dédié. En effet parfois

peu de question (ou contraintes) se posent sur l'utilisation d'une baie de disques par rapport à une autre et souvent le sujet stockage est traité dans sa globalité dans l'organisation sans tenir compte des spécificités de certains systèmes, si ce n'est l'aspect volumétrie. Or les innovations technologiques de ces dernières années en matière de stockage méritent d'être étudiées, pour en faire bénéficier les architectures des systèmes décisionnels. On attend du support de stockage qu'il soit stable, sécurisé, peu onéreux, de grande capacité, qu'il permette la compression des données et qu'il soit performant aussi bien en lecture, qu'en écriture.
Les technologies de types FLASH [12] prennent tout leur sens dans ce contexte. Le stockage Flash est une technologie de stockage des données qui repose sur une mémoire ultra rapide à programmation électrique. L'écriture des données et les opérations d'entrée-sortie aléatoires sont réalisées à la vitesse de l'éclair.Le stockage Flash utilise un type de mémoire non volatile appelée mémoire Flash, qui n'a pas besoin d'une alimentation électrique pour préserver l'intégrité des données stockées. Donc, même en cas de coupure de courant, rien n'est perdu. Le stockage Flash utilise des cellules de mémoire pour stocker les données. Vous devez effacer le contenu de ces cellules avant d'y écrire de nouvelles données.
Cette nouvelle technologie, qui reste encore un peu chère, permet notamment grâce à des techniques de compression de pointe, de stocker avec un ratio de 4 pour 1 les données. D'autres avantages tels que la réduction du temps de latence, la rapidité de production de la donnée, la diminution de la consommation énergétique, le gain d'espace (grâce au ratio de compression) ou sa simplicité d'administration, rendent ce type de stockage très intéressant dans le cadre des architectures décisionnelles, gourmandes en volumétrie et en accès lecture et écriture notamment.

Dans cette section nous avons détaillé, une partie, des innovations technologiques, au niveau de l'infrastructure qui ont ou vont faire évoluer les architectures des systèmes décisionnels, en permettant d'y intégrer des capacités d'accès à des informations mises à jour presque en temps réel, de façon plus rapide et moins onéreuse potentiellement.
Ces nouveautés techniques engendrent des modifications dans les architectures des systèmes décisionnels, les poussant vers un modèle de plus en plus hybride technologiquement.

L'intérêt accru autour de l'intelligence artificielle et ses dérivées, impact même la partie stockage qui au niveau des laboratoires de recherche étudient comment rendre plus "intelligent" les systèmes de stockage, en minimisant par exemple les temps d'arrêt ou bien en diagnostiquant automatiquement des problèmes techniques [13]. Cela n'est que le prémisse à de nouvelles possibilités technologiques.

12. https ://www.ibm.com/fr-fr/it-infrastructure/storage/flash
13. https ://www.usine-digitale.fr/article/quand-l-ia-colonise-massivement-les-baies-de-stockage.N779664

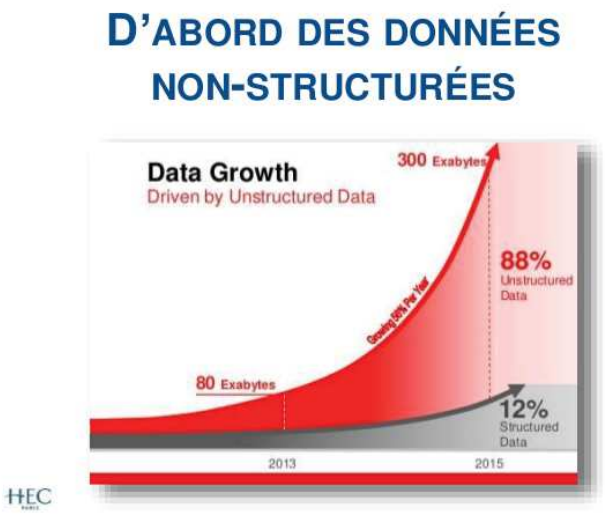


FIGURE *3.8: Evolution des données structurées et non structurées*

Cette hybridation technique ouvre la porte à de nouvelles possibilités d'exploitation des données, sous toutes leurs formes et donc à de nouveaux usages autour de l'information, que nous décrivons dans les deux prochaines sections.

3.3 L'évolution des données

On ne peut étudier l'évolution des systèmes décisionnels, sans étudier l'évolution de ce qui constitue son coeur : les données. Ces dernières ne cessent d'exploser en terme de volume comme le montre la figure 2.1, où l'on prévoit d'atteindre plus de quarante zeta bytes de données produites d'ici 2020. Au delà de leur volume c'est aussi au travers de leur diversité qu'elles évoluent. Désormais les données les plus "volumineuses" ne proviennent plus des systèmes transactionnels des organisations mais bien d'autres sources. Elles se retrouvent sous de nouveaux formats, structurés différemment. On parle alors de données non structurées par opposition aux données structurées, émises par les systèmes informatiques traditionnels.
Le graphe de la figure 3.8 montre l'inflexion de la courbe de répartition entre les données structurées et non structurées. Ce graphe montre que sur cent données émises, quatre vingt huit sont non structurées, l'émission de données structurées restant relativement stable.

Ces données non structurées recoupent les données d'organisation tels que les courriels, documents,

historiques de processus métiers..etc, aussi bien que des données issues de capteurs, des contenus publiés sur le web (images, vidéos, sons, textes), des transactions de commerce électronique, des échanges sur les réseaux sociaux, des données transmises par les objets connectés (étiquettes électroniques, compteurs intelligents, smartphones...), des données géolocalisées, etc.

L'intégration de ces données non structurées dans les systèmes d'information en vue d'en tirer de la valeur représente avant tout un défi technologique pour les conceptions d'architectures de part la variété de leurs formats mais aussi par leur volume et la rapidité, pour certaines, à laquelle elles sont émises.
C'est ce que l'on appelle le phénomène des données massives ou megadonnées ou big data. Nous employons les trois termes dans ces travaux, avec une préférence pour le terme "données massives " pour designer ce phénomène.

C'est Gartner qui en premier (2001) a défini les caractéristiques des megadonnées, avec le fameux principe des trois V :

— le Volume de données de plus en plus massif ;

— la Variété de ces données qui peuvent être brutes, non structurées ou semi-structurées ;

— la Vélocité qui désigne le fait que ces données sont produites, récoltées et analysées en temps réel.

Certaines entreprises[14] ajoutent un quatrième "V" (voire cinq ou plus) à cette définition pour la Véracité qui évoque la nécessité de vérifier la crédibilité de la source et la qualité du contenu afin de pouvoir exploiter ces données. Dans la figure 3.9 nous résumons ces "4V".

Les Données Massives, si l'on se réfère à la définition du Gartner, sont définies comme des données qui deviennent tellement volumineuses, rapides et variées qu'elles deviennent difficiles à travailler avec des outils classiques de gestion de base de données ou de gestion de l'information. Les données massives exigent donc des formes innovantes de traitement de l'information pour permettre une meilleure compréhension, prise de décision et automatisation des processus.
Dans leur travaux Vitari et Raguseo [80] ont étudié comment les technologies de l'information et de la communication (TIC) peuvent offrir de nouvelles opportunités aux entreprises, en mettant l'accent sur le rôle de Données Massives et des TIC sous jacentes dans la création de valeur économique. Ils proposent un classement des données massives selon leur origine : celles produites par les humains et celles produites par les machines.
Dans nos travaux nous adoptons la distinction des données selon leur structure et non leur origine : données structurées, non structurées et semi-structurées.

14. IBM fait partie des entreprises ajoutant un quatrième V au phénomène de méga données. https ://www.lebigdata.fr/infographie-quatre-v-big-data-expliques-ibm.

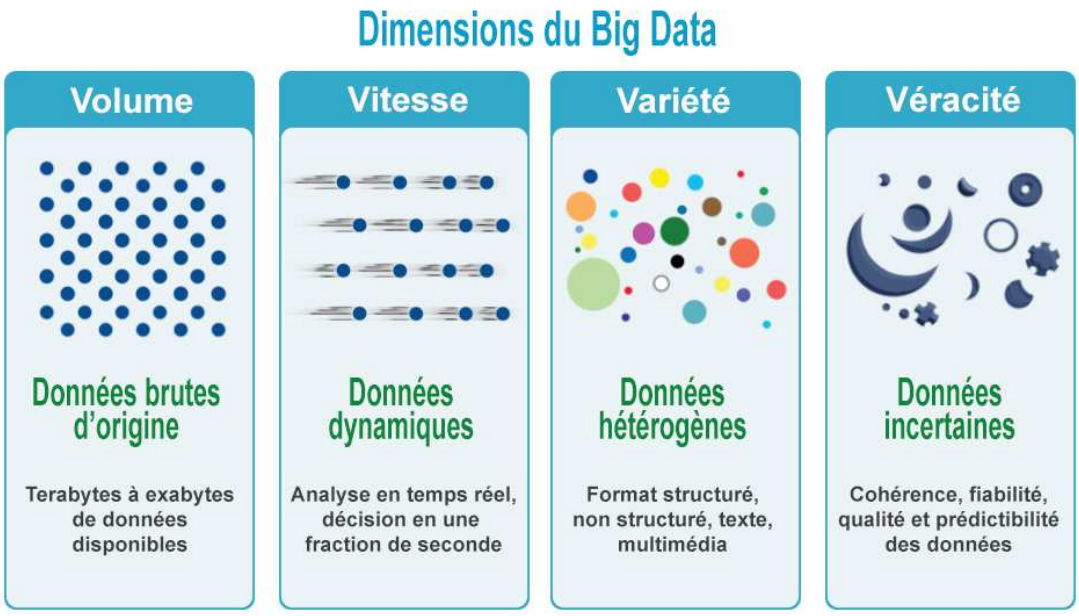


FIGURE *3.9: les 4 V des Megadonnées*

3.3.1 Données structurées, non structurées et semi-structurées : la différence

Les données non structurées ne sont pas organisées dans un format qui permet d'y accéder et de les traiter plus facilement. En réalité, très peu de données sont complètement non structurées. Même des éléments souvent considérés comme non structurés, tels que des documents et images, sont structurés dans une certaine mesure.

Les données structurées sont peu ou prou le contraire des données non structurées : elles ont été reformatées et leurs éléments, réorganisés, selon une structure permettant à chacun d'être traité, organisé et manipulé selon diverses combinaisons, afin de mieux exploiter les informations. Les données semi-structurées constituent une forme intermédiaire. Elles ne sont pas organisées selon une méthode complexe rendant possible un accès et une analyse sophistiqués ; cependant, certaines informations peuvent leur être associées, telles que des balises de métadonnées, qui permettent l'adressage des éléments qu'elles renferment.

Par exemple un document Word est généralement considéré comme un ensemble de données non structurées. Cependant, vous pouvez lui ajouter des métadonnées sous la forme de mots-clés qui représentent le contenu du document et qui permettent de le retrouver plus facilement lorsqu'une

recherche est effectuée sur ces termes. Les données sont alors semi-structurées. Cependant, le document n'est pas organisé de façon aussi complexe qu'une base de données, et ne se compose donc pas à proprement parler de données structurées.
En réalité, les limites entre les trois catégories sont extrêmement floues. Considérées dans leur ensemble, ces catégories sont parfois appelées le continuum des données.

3.3.2 L'évolution des données sous l'influence des données massives

L'évolution des données avec l'apparition du phénomène des données massives fait l'objet de plusieurs travaux de recherche, chacun s'attachant à étudier l'impact, notamment du volume de ces données sous divers angles. Dans ses travaux Sansen [68] s'attache à la problématique de visualisation de ces données massives, en soulevant les problématiques, techniques, générées par l'accumulation de données : stockage, temps de traitement, hétérogénéité, vitesse de captation/génération, etc.
Ces problématiques techniques sont d'autant plus impactantes que les données sont massives, complexes et variées. Il n'aborde pas cependant pas ces points de façon détaillée.
Dans nos travaux nous ne focalisons pas sur cet aspect visualisation, déjà traité dans les travaux de Sansen (et d'autres) mais bien sur les problématiques techniques évoquées par Sansen, auxquelles sont confrontées les systèmes d'information des organisations.
D'autres travaux académiques sur le domaine des données massives s'orientent autour de l'exploitation des données massives, comme ceux de Sansen, ceux de Gillet [24] qui se focalisent sur l'optimisation des requêtes sur ces données massives ou les travaux de Perrot, connexes à ceux de Sansen, sur l'aspect visualisation des données massives [57].
Ces extraits de travaux sur les données massives ont tous en commun d'être récents (2017), le phénomène des données massives n'ayant encore que peu de recul, comparé à la maturité des connaissances sur les systèmes décisionnels. Le nombre de ces travaux est en augmentation constante, signe de l'intérêt du sujet au niveau académique.

Dans nos travaux nous nous concentrons sur l'impact de ces données massives au niveau de l'architecture d'information, et son influence sur l'évolution des systèmes décisionnels.

Nos travaux de recherches, nous amènent à observer que le volume de données émises (des données massives) corrélé à la variété des formats des données impacte les architectures des systèmes décisionnels. C'est l'association de ces deux facteurs (**volume-variété**) qui accélèrent l'adoption des évolutions d'infrastructure (décrites dans la section 3.2) dans la conception des architectures des

systèmes décisionnels.
La prise en compte du facteur **vitesse** des caractéristiques des données massives, sollicite la partie infrastructure des architecture des systèmes d'information. C'est au travers de la partie acquisition de données, (émises rapidement et massivement) que les évolutions technologiques du stockage des données (que nous avons décrit dans la section 3.2.4) mais aussi celles des outils d'intégration des données (décrits dans la section 3.1) vont être le plus sollicitées, dans les conceptions des architectures décisionnelles.

Il y a cependant une autre problématique, peu soulevée dans la littérature, autour de la vitesse, dans les données massives, qui n'est pas celui de l'acquisition, mais celui de l'accès aux données.
En effet plus un volume de données est important, plus techniquement il est difficile d'accéder, rapidement, à ces données. Nous avons aussi abordé ce point dans la partie archivage des données section 3.1.4, qui présente la même problématique.

C'est donc l'accélération du temps des requêtes, par exemple, qui va devoir être traitée dans la conception de l'architecture d'information. Plus précisément, en reprenant la démarche d'urbanisation de Servigne [70] c'est l'architecture technique des systèmes décisionnels qui va devoir proposer une solution pour résoudre cette problématique.

Les "appliances" décisionnelles, comme Teradata, par exemple, traitent ce point, pour les données structurées essentiellement. Les données non structurées ou semi-structurées doivent pouvoir exploiter, via leur mode de stockage, des "accélérateurs" techniques. Ces derniers peuvent être purement liés à une caractéristique physique d'un serveur ou à une propriété logicielle de la base de données.
Nous pouvons illustrer cela au travers d'un exemple de produit industriel, celui d'IBM[15], "DB2 Analytics Accelerator "qui est un dispositif haute performance étroitement intégré à la base de donnée "Db2 for z/OS". Il assure un traitement ultra-rapide des requêtes DB2 complexes qui prennent en charge les rapports critiques et les charges de travail analytiques. Ce "produit" étroitement lié à l'infrastructure est une réponse à un besoin d'accélération des requêtes.

L'évolution des données, sous l'influence des données massives, impacte donc la partie infrastructure des architectures d'information. Les systèmes décisionnels pour s'adapter doivent donc intégrer les nouveautés technologiques et faire évoluer la partie technique de leur architecture.

C'est au travers des usages de ces données massives, qu'il faut traiter dans les systèmes décisionnels, que les choix technologiques sont mis en place dans les architecture d'information et induisent leur

15. https ://www.ibm.com/fr-fr/marketplace/db2-analytics-accelerator

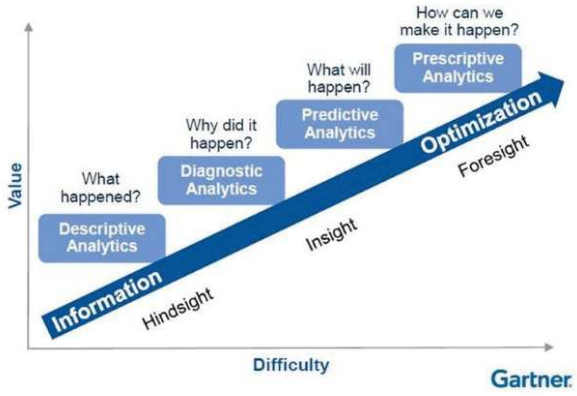


FIGURE *3.10: Niveau de maturité de l'utilisation des systèmes décisionnels, vision du Gartner*

évolution.
La section suivante traite les différents usages autour de ces données variées et volumineuses.

3.4 L'évolution des usages

La maturité des systèmes décisionnels, qui sont présents dans les organisations depuis plus de trente ans, s'accompagne aussi d'une maturité au niveau des utilisateurs et des usages qu'ils font de l'information qui leur est mise à disposition. Selon Le Gartner 3.10, on distingue quatre niveaux de maturité dans l'utilisation des données dans les systèmes décisionnels :

— analyse descriptive ;

— analyse de diagnostic ;

— analyse prédictive ;

— analyse prescriptive.

3.4.1 L'analyse descriptive

Ce type d'analyse répond à la question : *Que s'est-il passé ?*
Cela implique d'analyser un ensemble de données et de décrire ses dispositifs et ses caractéristiques. Ceci est principalement utilisé pour décrire et caractériser des événements passés. Les rapports financiers, les analyses des points de vente, les tableaux de pilotage d'une centrale d'achat sont par exemple les informations qui sont délivrées au travers l'analyse descriptive. Les acteurs du marché sont très nombreux, le graphe 3.11 qui est un "magic quadrant" du Gartner, expose le positionnement des

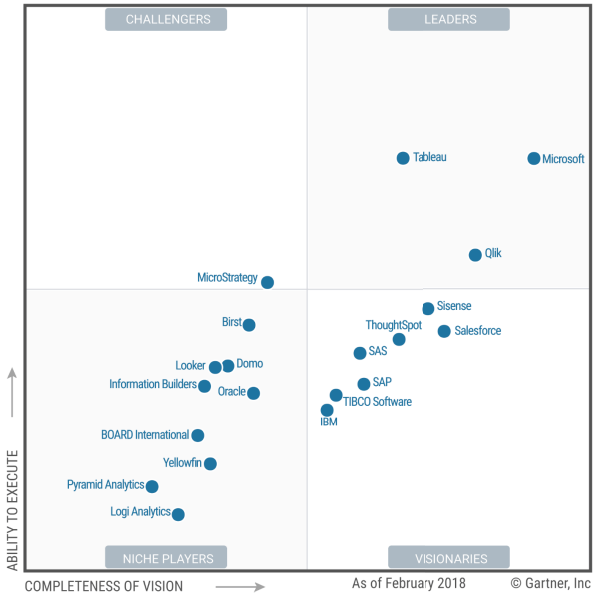


FIGURE *3.11: Positionnement des acteurs majeurs du marché des outils d'analyse descriptive et de diagnostic en 2018, vue du Gartner.*

acteurs majeurs du marché des outils d'analyse descriptive et de diagnostic. Trois acteurs se détachent particulièrement : Microsoft, Qlik et Tableau.

Certains n'existaient même pas sur ce marché, assez volatile, il y a quelques années, comme le montre le graphe 3.12 qui représente un ancien "magic quadrant" du Gartner datant de 2009

L'analyse descriptive, présente dès le début des systèmes décisionnels, reste le coeur de l'information délivrée par les systèmes décisionnels. Les nombreuses réglementations et leurs évolutions, qui affectent les organismes financiers (banques et assurances), telles que les normes de Bâle[16], rendent ces besoins d'analyse toujours d'actualité. Il n'est pas rare de trouver des systèmes décisionnels entièrement dédiés à la production des rapports pour ces réglementations. Les entrepôts de données dédiés à la gestion du risque par exemple, sont très fréquents dans les organismes financiers.

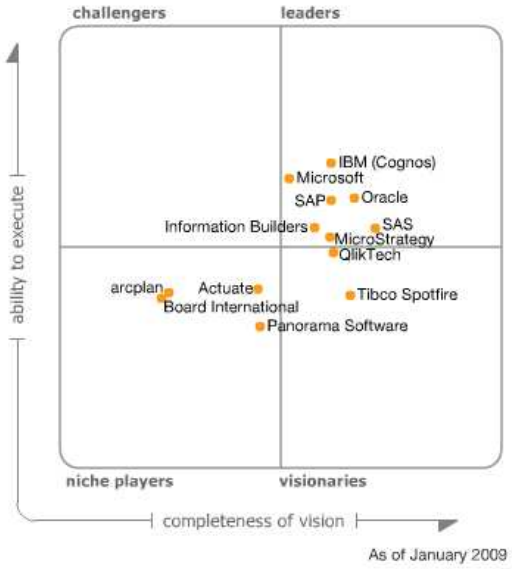


FIGURE *3.12: Positionnement des acteurs majeurs du marché des outils d'analyse descriptive et de diagnostic en 2009, vue du Gartner.*

3.4.2 L'analyse de diagnostic

Les gestionnaires utilisent l'analyse diagnostiquée pour répondre à la question *Pourquoi ?* Ou *Pourquoi est-ce arrivé ?*.

Ils se concentrent sur des résultats connus et tentent de déterminer les facteurs et les événements qui ont contribué à cet événement. Cela est basé sur une action ou un fait qui a eu lieu dans le passé. Par exemple, la société a perdu un gros client, et on cherche à savoir pourquoi cela s'est passé afin que cela ne se reproduise pas.

Du point de vue de la chronologie, l'analyse descriptive et de diagnostic, sont les premières informations qui ont été délivrées au travers les premiers systèmes décisionnels. Les outils logiciels qui permettent de délivrer ces informations sont ceux qui ont le plus de maturité, pour ceux qui ont survécu aux multiples rachats et fusions des différents sociétés créatrices de ces logiciels.

16. Entrée en vigueur en 2010, Bale III est une réforme financière qui a pour but de renforcer la sécurité et la solidité du système bancaire. Cette réforme a été mise en place après la grande crise financière de 2007 afin d'éviter que de tels événements se reproduisent. Elle a été mise au point par le comité de Bâle qui coordonne l'efficacité du contrôle prudentiel et la coopération entre les principaux régulateurs bancaires de la planète.

3.4.3 L'analyse prédictive

Ce type d'analyse répond à la question *Ce qui est susceptible de se produire ?*
L'analyse prédictive, parfois appelée analyse avancée, est un terme utilisé pour décrire une série de techniques analytiques et statistiques permettant de prédire des actions ou des comportements futurs. Dans les organisations, l'analyse prédictive permet, par exemple, d'optimiser les prises de décision, d'accroître la compétitivité sur le marché, de réduire l'incertitude et de gérer les risques, de découvrir des schémas judicieux et des moyens d'anticiper et de réagir aux tendances émergentes. Afin de pouvoir faire des prédictions sur la base d'un ensemble de données spécifiques, on utilise une ou plusieurs variables prédictives pour prédire une variable réponse. Sous sa forme la plus simple, l'analyse prédictive est un support qui permet d'établir des prévisions pour la prise de décision en entreprise. Pour des exigences plus complexes, on utilise des techniques d'analyse prédictive avancées afin d'orienter les processus stratégiques de l'entreprise.
Il existe principalement deux catégories d'analyse prédictive : l'apprentissage supervisé et l'apprentissage non supervisé.
L'apprentissage supervisé est divisé en deux grandes catégories : la régression pour les réponses quantitatives (une valeur numérique), et la classification pour les réponses qui peuvent uniquement prendre des valeurs connues, telles que "vrai" ou "faux".
L'apprentissage non supervisé permet de tirer des conclusions à partir d'ensembles de données composés de données en entrée sans réponses étiquetées. La méthode d'apprentissage non supervisé la plus courante est l'analyse de clusters, qui est utilisée pour l'analyse exploratoire des données afin de trouver des schémas cachés ou des groupes de données.

L'analyse prédictive est l'usage qui est le plus en développement à l'heure actuelle. En effet les analyses prédictives sont utilisées dans de nombreuses situations dans le monde industriel. Par exemple dans le secteur du marketing, qui utilisent désormais du "marketing predictif", pour attirer de nouveaux consommateurs, les retenir et leur proposer un service de meilleure qualité que leurs concurrents.

— **Le Scoring Predictif :** Prioriser les différents profils de clients en fonction de leur inclinaison à acheter. Cette méthode ajoute une dimension mathématique à la priorisation conventionnelle qui repose sur la spéculation et l'expérimentation. Ce cas d'usage aide les équipes de vente et de marketing à identifier les comptes productifs plus rapidement, à perdre moins de temps sur les comptes moins enclins à acheter, et à développer des campagnes à succès.

— **Les modèles d'identification :** Identifier et acquérir des clients avec des attributs similaires à ceux déjà fidélisés. Dans ce cas d'usage, les comptes qui ont présenté des comportements désirés, comme le fait d'effectuer un achat, de renouveler un contrat ou d'acheter des services supplémentaires sert

de base pour un modèle d'identification. Il permet d'aider les équipes de ventes et de marketing à trouver des prospects exploitables plus tôt dans le processus de vente, à prioriser les comptes existants en vue d'une expansion, et à mettre en exergue les comptes susceptibles d'être plus réceptifs aux messages de ventes et de marketing.

— **Segmentation automatisée :** Découper la clientèle en segments pour des messages personnalisés. Traditionnellement, les marketers B2B sont seulement capables de segmenter la clientèle par attributs génériques tels que l'industrie dans laquelle ils travaillent. Par ailleurs cette segmentation manuelle requiert des efforts applicables uniquement aux campagnes prioritaires. Désormais, les attributs utilisés pour alimenter les algorithmes prédictifs peuvent être utilisés pour une segmentation automatisée des comptes. Ceci permet aux équipes de vente et de marketing de gérer des communications avec des messages pertinents, des conversations substantielles entre les vendeurs et les prospects, et de diriger la stratégie de contenu plus intelligemment.

De même, les analyses prédictives ont un fort impact sur l'industrie de l'Internet : Google utilise les algorithmes de Machine Learning dans ses data centers pour la maintenance prédictive de ses fermes de serveurs de Cloud public. Les algorithmes utilisent les données météorologiques et d'autres variables pour ajuster le refroidissement du data center et réduire la consommation d'énergie.
Ce genre de maintenance prédictive devient courante dans les usines. Les géants de la technologie d'entreprise comme SAP offrent un service de maintenance prédictive utilisant les capteurs de données des machines de productions connectées pour prédire quelle machine risque de rencontrer des problèmes mécaniques.
La liste des applications potentielles s'étend à perte de vue. Les analyses prédictives transforment l'industrie et sont positionnées au coeur de la transformation digitale des organisations. C'est l'usage qui a le plus évolué dans les systèmes décisionnels et qui est au cœur de son évolution actuellement.

3.4.4 L'analyse prescriptive

La progression des logiciels, des infrastructures et des données, a permis aux entreprises agiles de passer de l'analyse descriptive *que s'est-il passé ?* à l'analyse diagnostique *qu'est-ce qui a causé le problème ?* et à l'analyse prédictive *qu'est-ce qui est susceptible d'arriver ?*.
L'analyse prescriptive *comment atteindre notre objectif ?* est l'étape ultime de cette progression stratégique.
L'analyse prescriptive utilise les deux concepts d'analyse descriptive et prédictive, dans un but principal : Proposer des voies d'optimisations aux utilisateurs, de l'aide à la décision, leur permettant ainsi de réagir plus rapidement, et de façon la plus appropriée, à une situation en devenir. C'est sa capacité à donner à l'utilisateur les moyens de prendre des décisions rapidement, chose qu'il n'aurait pas pu forcément faire

en raison du nombre d'indicateurs à prendre en compte et de leur complexité, qui rend le prescriptif intéressant.

La première étape dans le cadre de ce type d'analyse est d'identifier dans les données qui sont mises à disposition (grâce à l'analyse descriptive) les leviers d'actions, ou les points possibles d'optimisations qui pourraient être intéressant. S'en suit une étape de simulations et d'optimisations mettant à profit le descriptif comme le prédictif.
L'idée est on ne peut plus simple, on retire certaines variables, on en modifie d'autres et l'on rejoue nos données pour voir si ce que l'on aurait obtenu grâce à nos optimisations et par extension si nos prédictions (forecast) s'améliorent (en théorie). Cependant il faut bien prendre en considération que bien que le prescriptif soit économiquement plus intéressant que les autres analyses, il reste bien plus complexe à mettre en œuvre. Sa mise en place ne doit pas être prise à la légère car il nécessite bien plus de temps investi pour son développement qu'une simple interface descriptive.

Les usages de l'analyse prescriptive sont déjà nombreux dans le monde industriel : la tarification, la gestion des stocks, l'allocation des ressources opérationnelles, la planification de la production, l'optimisation de la chaîne logistique, la planification des transports et de la distribution, la planification financière et bien d'autres applications. Ainsi, les systèmes de tarification des billets d'avion exploitent l'analytique prescriptif pour faire le tri entre des combinaisons complexes comprenant les conditions de voyage, le niveau de demande et la période d'achat. Ils peuvent ainsi proposer des prix permettant d'optimiser les bénéfices sans pour autant dissuader les clients.

Au vu de l'engouement actuel pour l'analyse prédictive et l'essor qui lui est donné, l'analyse prescriptive, qui s'appuie sur la maturité de l'analyse descriptive et l'évolution des techniques d'analyse predictive devrait rapidement se retrouver au cœeur des usages des systèmes décisionnels.
Les études et prédictions faites par le Gartner sur le sujet, pour l'année 2017, sont résumés au travers d'un graphe appelé "hype cycle" (voir figure 3.13). Cette courbe présente les technologies selon les axes Visibilité et Maturité.
Nous étudions cette courbe pour savoir si nos observations correspondent avec les prédictions du Gartner. Un "Hype Cycle" comprend 5 phases :
"Technology Trigger" : La première phase d'un "Hype Cycle" correspond à l'arrivée sur le marché d'un nouveau produit ou d'une nouvelle technologie.
"Peak of Inflated Expectations" : Dans la phase suivante, un emballement généralisé aboutit souvent à des attentes exagérées et non réalistes (Buzz). Un certain nombre de sociétés mettront en oeuvre cette technologie avec succès, mais beaucoup d'autres termineront en échec.

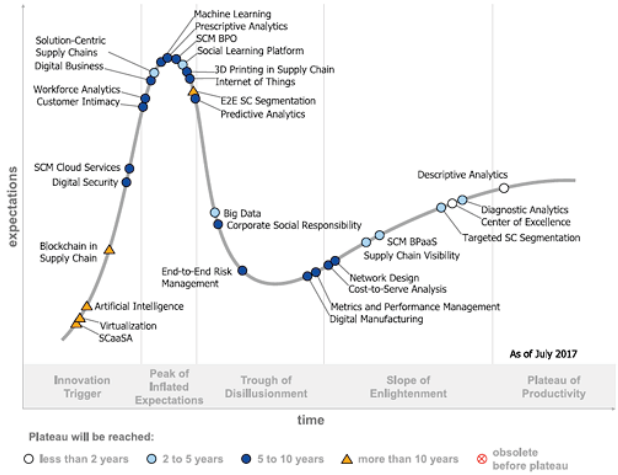


FIGURE *3.13: Hype cycle Gartner 2017- les 4 types d'analyses*

"Trough of Disillusionment" :Cette technologie entrera dans le "creux de désillusion" parce qu'elle ne parvient pas à répondre aux attentes et devient vite démodée. Par voie de conséquence, la presse se détourne généralement de ce sujet et de cette technologie.
"Slope of Enlightenment" : Bien que la presse ait peut-être cessé de couvrir cette technologie, certaines entreprises continuent à travers la « pente de l'illumination » et l'expérimentent pour comprendre ses avantages et ses pratiques d'application.
"Plateau of Productivity" : Une technologie atteint le "plateau de productivité" lorsque les bénéfices qu'elle procure deviennent largement démontrés et acceptés. La technologie devient de plus en plus stable et évolue dans une deuxième puis troisième génération. La hauteur finale du plateau est variable suivant que la technologie est largement applicable ou au contraire ne bénéficie qu'à un marché de niche. Un Hype Cycle contient de nombreuses technologies sur la même courbe. Une légende située en bas du graphique indique le délai estimé restant avant d'atteindre le plateau de productivité.
Dans le cadre de nos quatre usages des données, on voit que l'analyse descriptive et de diagnostic sont positionnées sur le "Plateau of Productivity", ce qui correspond bien à nos constations précédentes, ces analyses ont démontré le bénéfice qu'elles apportent et continuent de s'améliorer.

L'analyse prédictive se situe à la limite entre "Peak of Inflated Expectations" et "Trough of Disillusionment". Ce qui traduit bien l'engouement et l'emphase qui sont mis sur ce type d'usage, comme nous l'avons décrit précédemment. Les attentes des organisations sont très fortes sur le sujet, et un peu de recul nous montrera si l'analyse prédictive atteint le seuil de "Plateau of Productivity",

comme l'analyse descriptive et de diagnostic.

L'analyse prescriptive quant à elle se situe tout en haut du "Peak of Inflated Expectations". Ce qui correspond aussi à notre constat de l'usage de l'analyse prescriptive qui en est au tout début de sa démocratisation dans les organisations et reste encore au stade d'expérimentation pour certaines voire de planification. L'avenir nous dira si dans les prochaines années, l'analyse prescriptive se diffuse dans les processus des organisations de façon plus globale.

Ces quatre type d'analytiques, qui sont une évolution des usages des données, au travers des systèmes décisionnels, imposent à l'architecture de référence de s'adapter et d'évoluer elle aussi afin de les supporter.

C'est notamment l'architecture des données, au travers de l'aspect modélisation qui va permettre à ces différents usages de cohabiter. Nous étudions dans la section suivante l'impact de l'évolution des données et de leurs usages sur la modélisation des systèmes décisionnels.

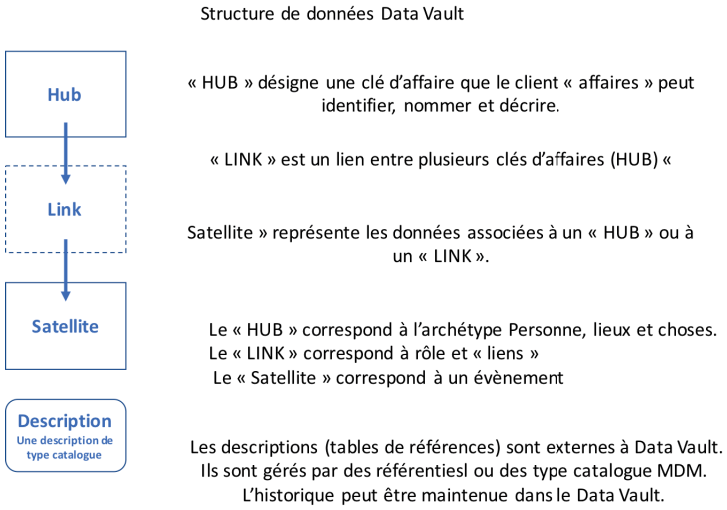


FIGURE *3.14: Structure de données du data Vault*

3.5 L'évolution de la modélisation

Depuis de nombreuses années, deux visions classiques s'affrontent en ce qui concerne la modélisation des entrepôts de données : l'approche Inmon [35] de modélisation d'entrepôt de données d'entreprise par sujet et normalisée et l'approche Kimball de modélisation en étoiles où l'intégration en un entrepôt d'entreprise est assurée par des dimensions conformes et l'usage d'une matrice de bus.

Bien que moins présente que ces deux approches classiques, deux autres approches font évoluer les modélisations des systèmes décisionnels : l'approche de modélisation *Data Vault* (par voûtes de données) et l'approche *Anchor modeling* [63], qui sont toutes les deux des modélisation ensemblistes. Ces deux modélisation agiles étendent le Datawarehouse sans le dégrader. Le "datavault" "historise" les données issues de différentes sources et "l'anchor modeling" gère les changements apportés à la structure des données et à leur contenu.

La modélisation Data Vault (DV) a été inventée par Dan Linstedt[17][13] au début des années 2000. La modélisation Anchor modeling (ou modélisation d'ancre)[18] a été inventée par Lars Ronnbark dans les années 2010 [42]. La figure 3.15 est une représentation graphique d'une modélisation de type Anchor Modeling ; la figure 3.16 celle d'une modélisation de type Data Vault.

Lors de l'étude de ces deux approches, au travers les travaux de Hultgren [30] [31] et Ronnbark [63]

17. http ://danlinstedt.com/
18. http ://www.anchormodeling.com/

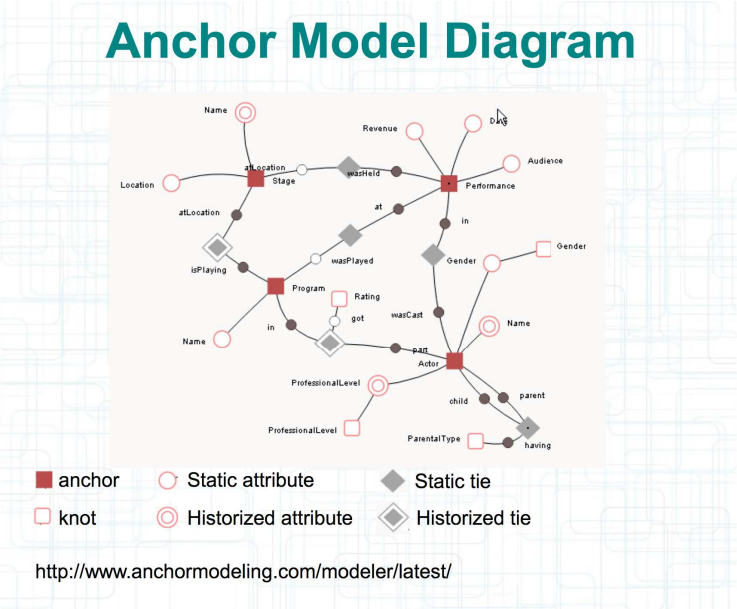


FIGURE *3.15: Modélisation de type ANCHOR MODELING*

nous avons constaté qu'elles étaient assez proches, nous avons cependant retenu l'approche Data Vault dans nos travaux, car leur modélisation est proche de la modélisation que nous utilisons dans les systèmes décisionnels (la forme 3NF). L'approche Anchor modeling n'est toutefois pas exclue de nos travaux de recherche futurs.

Plusieurs autres travaux académiques étudient en détail cet aspect de l'évolution de schémas dans les entrepôts de données, dont ceux de Favre [19] qui détaillent l'évolution du schéma de l'entrepôt guidée par les utilisateurs, ou bien ceux de Oliviera et Kaldeich [40] qui étudient l'évolution des schémas des entrepôts de données au travers de processus.

Dans le cadre de nos travaux, nous optons pour une approche au niveau des processus (celle des Data Vault), similaire à celle de Oliviera et Kaldeich. Cette approche en Data Vault a été reprise par Inmon, lors d'une collaboration avec Linstedt [81], qui a abouti à une évolution de la position de Imnon, en introduisant dans son concept d'entrepôt de données [35] une modélisation en Data Vault.

3.5.1 L'approche Data Vault

.

Jusqu'à présent, les entrepôts étaient uniquement modélisés selon une architecture de données. Data Vault introduit une notion d'architecture de processus. Les structures de données sont déterminées selon

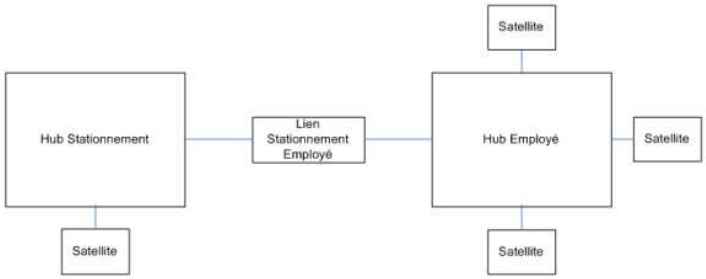


FIGURE *3.16: Modèle de Data Vault*

une modélisation relationnelle ET selon une notion de processus selon la "fonction" de la donnée.

La structure du Data Vault a été conçue en considérant l'évolution dans son contexte technique et non dans le contexte "affaires". Le changement des processus et des structures de données est ciblé plutôt que les changements et l'évolution des fonctions d'affaires. Les étapes d'historisation des données, d'intégration et normalisation pour l'entreprise et la présentation des données sont traitées de façon distincte.

Un modèle Data Vault est composé de trois types d'entités : les *hubs*, les liens *links* et les *satellites*.

Les objectifs d'une modélisation en Data Vault :

— Permettre de retracer facilement l'information aux sources de données (ex : audit de données) ;

— Être robuste aux changements du modèle d'affaires (ex : relation 1-N devenant N-N) ;

— Réduire les contraintes aux règles d'affaires en différant celles-ci (ex : magasin de données en aval) ;

— Permettre un chargement efficace des données.

Le principe de base des Data Vault est de séparer l'information structurelle (Hub+Link) des attributs descriptifs (Satellites). La figure 3.17 illustre la modélisation Data Vault ; Dans le paragraphe suivant nous détaillons ces trois notions que sont le Hub, le Lien (Link) et le Satellites ;

Un hub :
Les hubs sont des concepts d'affaires, les entités contiennent les clefs naturelles (clefs d'affaires) qui identifient le concept et qui sont par nature très stables. Elles ne contiennent aucune donnée qui décrit l'entité (celles-ci sont gardées dans les entités satellites décrites plus bas). Elles constituent souvent le point de raccordement (d'où le terme anglais «hub») entre plusieurs secteurs d'une organisation. La figure 3.16 montre un exemple de modèle Data Vault. Les entités Stationnement et Employé sont des

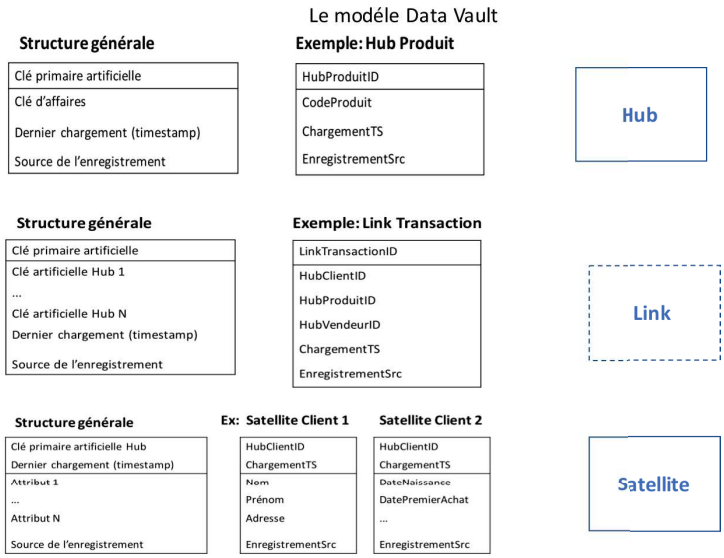


FIGURE *3.17: Synthèse des trois entités d'une modélisation Data Vault*

hubs et l'entité Stationnement Employé est un lien. Les critères d'un Hub sont les suivants :

- représente un et un seul concept ;
- ne contient aucun élément de données descriptif (exemple : le nom d'un employé) ;
- ne contient aucune relation ;
- contient idéalement une clef naturelle unique composée d'au moins un élément de données qui identifie le concept (exemple : le numéro du stationnement). Si, par exemple, dans deux sources une même valeur de clef naturelle ne correspond pas à la même instance de concept, il faut alors identifier uniquement les différentes instances en utilisant la clef naturelle combinée avec le nom de la source d'où provient la donnée ;
- contient toujours au minimum deux informations permettant la traçabilité : la source d'où provient la donnée et quand la donnée a été amenée dans le hub.
- est associé à au moins un satellite pour le décrire ;

Un lien :

Un lien est une entité dépendante représentant une interaction entre au moins deux concepts dont elle dépend. Comme dans le cas d'un modèle étoilé, la granularité d'un lien (son niveau de détail) est dictée par les hubs en relation avec le lien. Les critères à respecter pour être un "bon" lien sont les suivants :

- est associée à au moins deux concepts (hubs) parents (remarque : dans le cas d'un lien hiérarchique, on peut associer deux fois au même hub) ;

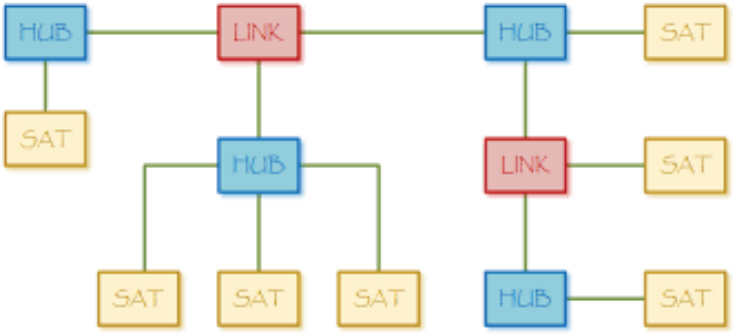


FIGURE *3.18: Exemple de Modélisation Data Vault*

— ne contient aucun élément de données descriptif (exemple : la date de début et de fin de la relation). Les satellites d'un lien contiennent la partie descriptive ;

— est le seul type d'entités contenant les relations entre concepts ;

— est utilisé peu importe la cardinalité de la relation entre deux concepts. Elle ne considère pas des cardinalités plus spécifiques qu'une relation plusieurs à plusieurs. Un changement de cardinalité n'affectera donc aucunement le modèle (remarque : une règle de cardinalité plus spécifique peut être documentée au niveau des métadonnées) ;

— contient toujours au minimum deux informations permettant la traçabilité : la source d'où provient le lien et quand l'instance du lien a été amenée dans l'entrepôt ;

— est implanté par des clefs étrangères qui pointent vers les identifiant (internes) des hubs en relation ;

— est défini avec le grain le plus fin possible, sauf pour des liens redondants définis à une granularité plus élevée pour des raisons de performance ;

— est créé si le grain change et l'ancien lien demeure ce qui permet d'éviter la ré ingénierie du modèle existant et assure la vérifiabilité.

Les satellites :
Les satellites contiennent les données qui décrivent les hubs et les liens à un moment donné et à travers le temps. Ces entités contiennent le contexte (provenant des processus d'affaires) d'un hub ou d'un lien. Comme les données descriptives changent souvent, l'idée des satellites est de conserver les changements lorsqu'ils surviennent. Le satellite est une entité dépendante (ou faible) toujours en relation avec un hub ou un lien. Inversement, un hub ou un lien doit toujours contenir au moins un satellite pour le décrire.

3.5.2 Règles de base d'une modélisation Data Vault :

- Les données sont normalisées AVANT le chargement. Les données d'un "Satellite " ne dépendent que de la clé du satellite ;
- Une clé d'affaires n'est définie qu'une seule fois dans une structure de données ;
- Les données ne sont pas filtrées, corrigées ni interprétées. Toutes les données ont une traçabilité jusqu'à la source originale ;
- Les données ne sont jamais modifiées ;
- Les clés de la voûte ne sont jamais utilisées hors de la voûte.
- L'accès à la voûte est restreint, elle n'a pas une structure répondant directement à une exploitation finale des données.

3.5.3 Lien avec la modélisation dimensionnelle :

- Les Hubs et leur Satellites correspondent aux tables de dimension ;
- Les Links et leur Satellites correspondent aux tables de faits ;
- Il faut appliquer les règles d'affaires lorsqu'on charge les Data mart (schéma en étoile) à partir du Data Vault.

3.5.4 Avantages de Data Vault

- Au chargement, il n'y a pas de dépendance entre les fichiers de données. Puisque les Hubs sont découplés (aucune clé étrangère d'un Hub à un autre), on peut les charger en parallèle. Même chose pour les Links et Satellites.
- L'intégration des données se fait sous un mode passif. Les données d'un satellite se retrouvent sous les mêmes HUB et LINK lorsqu'il a les mêmes structures de clés.
- Lorsqu'une règle d'affaires change, les structures en place ne sont pas modifiées. De nouvelles structures sont ajoutées sans impact à l'existant. La «navigation» vers les données est modifiée.

Plusieurs de ces critères font en sorte que la structure existante du modèle n'a pas besoin d'une ré-ingénierie lorsque des changements surviennent dans l'environnement d'affaires. La structure du modèle Data Vault est conçue de telle sorte que lorsque des changements surviennent, ceux-ci n'ont pas d'impacts sur les parties existantes du modèle. Des liens sont ajoutés sans réviser les structures existantes. La structure est donc flexible et les nouveautés sont ajoutées avec beaucoup moins d'efforts. Il n'y a aucun besoin de conversion de données existantes dans la nouvelle structure. La figure 3.14

résume les composants d'une modélisation en data vault.

3.5.5 Inconvénients de Data Vault

— Un seul fichier génère plusieurs tables à charger. C'est le prix de l'indépendance des chargements.

— La voûte n'est pas accessible facilement. C'est une représentation du FAIT, il est organisé selon la source de données et non la destination finale et il ne change pas.

— Data Vault n'est pas un modèle exploitable. Data Vault est une fondation pour l'historisation des données. Pour avoir une version exploitable, il faut créer la partie "navigation" avec les besoins d'affaires précis.

La modélisation Data Vault exige une modélisation de données très différente de ce qui est connu aujourd'hui (modèle relationnel et dimensionnel), même si le modèle logique est réalisé en 3NF, le modèle physique Data Vault dépend des spécifications du chargement.

3.5.6 Conclusion sur les Data Vaults

Plusieurs travaux sur la modélisation en Data Vault, dont ceux de Jovanovic et Bojicic [39], positionnent la modelisation en Data Vault non pas en opposition avec celle de Kimball ou Imnon mais en complément, en la positionnant au niveau du composant ODS (ou staging), pour historiser les données des systèmes opérationnel, du "staging "amélioré en sorte.

Cette évolution en terme de modélisation de données propose une évolution intéressante pour les systèmes décisionnels, une plus grande flexibilité dans la gestion de l'ODS, et des possibilités d'utilisations pour des utilisateurs souhaitant accéder à certaines données, plus fraîches directement dans l'ODS, au moment où elle sont disponibles sans attendre leur intégration des l'entrepôt de données, par exemple.

Ces nouvelles possibilités de modélisation dans les systèmes décisionnels nécessitent d'être mise en œuvre sur des bases de données capables d'en tirer parti, dans un contexte de données massives. Dans l'optique d'explorer les possibilités que peuvent offrir notamment la technologie Apache Hadoop, nous consacrons la prochaine section à cette technologie, son historique, sa composition, les acteurs du marché et bien sûr son influence sur l'évolution des architecture des systèmes décisionnels.

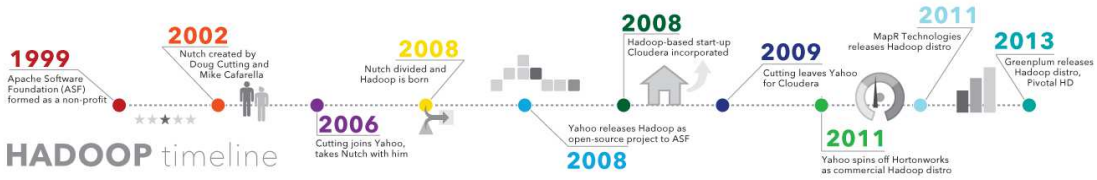


FIGURE *3.19: Historique de Apache Hadoop*

3.6 La technologie Apache Hadoop

3.6.1 Historique de Apache Hadoop

Hadoop[19] a été créé en 2004 par Doug Cutting pour les besoins du projet Apache Nutch, un moteur de recherche open source. Hadoop se base alors sur les travaux de Google au niveau du GFS (Google's distributed filesystem) et de MapReduce (algorithme de calcul à grande échelle) pour l'analyse des données d'un système GFS. Rapidement (2005) une version open source voyait le jour sous l'impulsion de Yahoo.
En 2006, Hadoop devient un sous-projet d'Apache Lucene et en 2008 un projet indépendant de la fondation Apache. La figure 3.19 reprend sous forme graphique les différentes étapes qui ont conduit à la forme du produit Apache Hadoop, telle que nous le connaissons aujourd"hui.

Pour l'anecdote, Doug Cutting, le créateur d'Hadoop, a baptisé l'infrastructure du nom de l'éléphant en peluche de son fils.

3.6.2 Les enjeux du marché Apache Hadoop

Selon une étude récemment menée par Zion Research[20], et synthétisée au travers du graphe de la figure 3.20, le marché mondial d'Hadoop valait 7, 69 milliards de dollars en 2016, et pourrait atteindre une valeur de 87 milliards de dollars en 2022, avec une croissance annuelle de plus de 50% entre 2016 et 2022. L'augmentation du volume de données structurées et non structurées au sein des grandes entreprises, et la volonté des entreprises d'utiliser ces données, sont les principaux facteurs de la croissance du marché de la plateforme de traitements distribués.
Ces prédictions de revenus liées à cette technologie sont la preuve que Apache Hadoop est bien un incontournable dans le traitement des données massives.

19. https ://www.lebigdata.fr/hadoop
20. https ://www.zionmarketresearch.com/news/hadoop-market

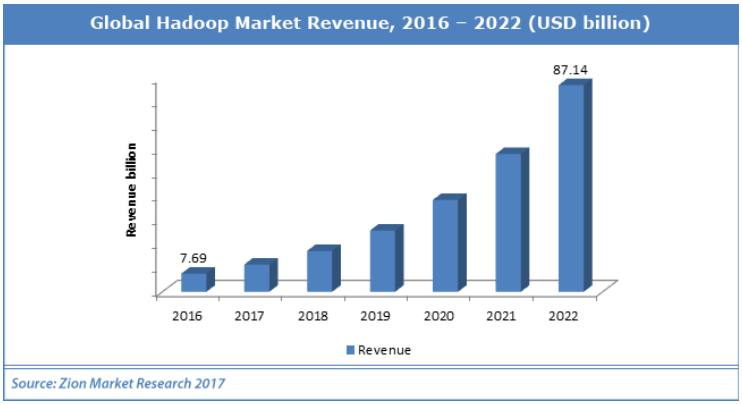


FIGURE *3.20: Prédiction du marché de la technologie Apache Hadoop en 2017*

3.6.3 Le lexique Apache Hadoop

Il est difficile de se retrouver dans la jungle de termes autour du "monde" Apache Hadoop, pour les raisons suivantes :

- Ce sont des technologies jeunes ;
- C'est le monde "open source" ;
- Beaucoup de buzz et de communication de sociétés qui veulent prendre le train des données massives en marche ;
- Beaucoup d'acteurs différents (des mastodontes, des spécialistes du web, des start-up, . . .).

Les mots comme framework, HDFS, Cluster, node, MapReduce font partie du lexique de base du monde Apache Hadoop ainsi ces composants et leur dérivés. Hadoop est un "framework" logiciel "open source" permettant de stocker des données, et de lancer des applications sur des grappes de machines standards. Cette solution offre un espace de stockage massif pour tous les types de données, une immense puissance de traitement et la possibilité de prendre en charge une quantité de tâches virtuellement illimitée. Il est basé sur le langage Java.
Son système de gestion de fichier est un système distribué qui se nomme HDFS (Hadoop Distributed File System), que nous décrivons en aval de cette section. La figure 3.21 montre le fonctionnement de ce fichier.

Ce système de fichiers distribué favorise un taux élevé de transfert de données entre les noeuds et permet un fonctionnement ininterrompu du système en cas de défaillance d'un d'entre eux. Cette approche diminue le risque de panne majeure, même lorsqu'un nombre important de noeuds deviennent inopérants. Hadoop s'inspire de Google MapReduce, un modèle logiciel qui consiste à fragmenter une application en

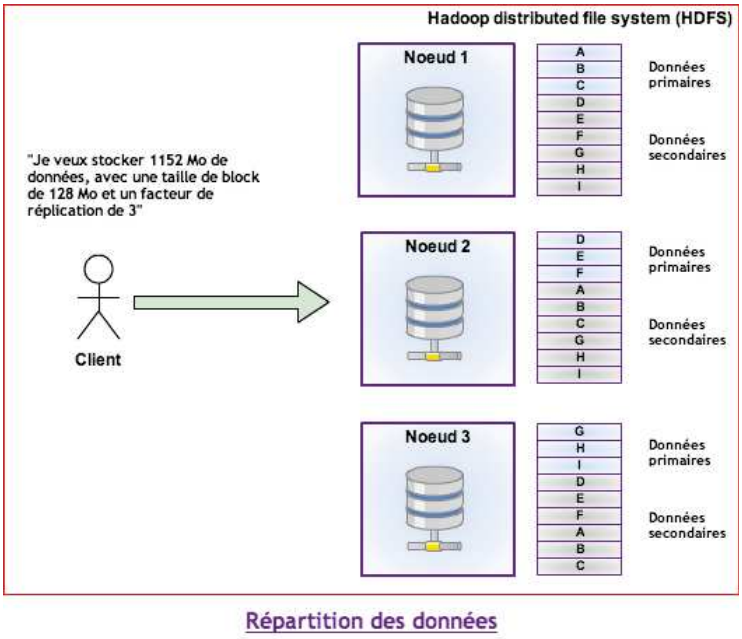


FIGURE *3.21: Apache Hadoop-Fichier HDFS*

de nombreux petits composants. Chacun de ces composants (appelé fragment ou bloc) peut s'exécuter sur n'importe quel noeud du cluster[21].
L'architecture Apache Hadoop est de type "Share nothing" : aucune donnée n'est traitée par deux noeuds différents même si les données sont réparties sur plusieurs noeuds (principe d'un noeud primaire et de noeuds secondaires).

En synthèse : Apache Hadoop se compose du noyau Hadoop, de MapReduce, du système de fichiers distribué (HDFS) Hadoop et d'un certain nombre de projets associés, notamment Apache Hive, HBase et Zookeeper.

3.6.4 Les composants d'Apache Hadoop

Dans une distribution Hadoop on va retrouver les composants suivants (ou leur équivalence) HDFS, MapReduce, ZooKeeper, HBase, Hive, HCatalog, Oozie, Pig, Sqoop, etc. Ces solutions sont des projets Apache, qui peuvent être utilisés séparément mais l'intérêt d'Apache Hadoop est d'utiliser ces composants

21. Un cluster est une grappe de serveurs sur un réseau, appelé ferme ou grille de calcul

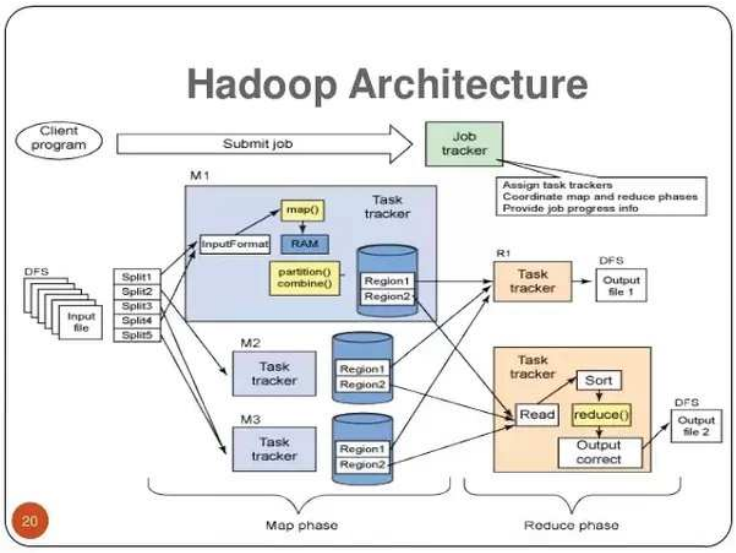


FIGURE *3.22: Architecture Apache Hadoop*

ensemble car ils sont conçus pour être compatibles et complémentaires. Nous décrivons plus en détail chacun de ces composants dans cette section, que nous utilisons dans une de nos contributions exposées au chapitre 4.

HDFS (Hadoop Distributed File System) :

HDFS est un système de fichiers Java utilisé pour stocker des données structurées ou non sur un ensemble de serveurs distribués. HDFS s'appuie sur le système de fichier natif de l'OS pour présenter un système de stockage unifié reposant sur un ensemble de disques et de systèmes de fichiers hétérogènes. La consistance des données est basée sur la redondance. Une donnée est stockée sur au moins n volumes différents.

Node :

Node (Master/slave) : Dans une architecture Hadoop chaque membre pouvant traiter des données est appelé node (Noeud). Un seul d'entre eux peut être master même s'il peut changer au cours de la vie du cluster.

Il est responsable de la localisation des données dans le cluster (il est appelé Name Node). Les autres sont des slaves appelés Data Nodes. Bien qu'il puisse y avoir plusieurs Name Nodes, la "promotion" doit se faire manuellement. Le Name Node est donc un Single Point Of Failure (SPOF) dans un cluster Hadoop. Au sein du cluster, les données sont découpées et distribuées en blocks selon les deux paramètres suivants :

— Blocksize : Taille unitaire de stockage (généralement 64 Mo ou 128 Mo). C'est-à-dire qu'un fichier de 1 Go (et une taille de block de 128 Mo) sera divisé en 8 blocks.

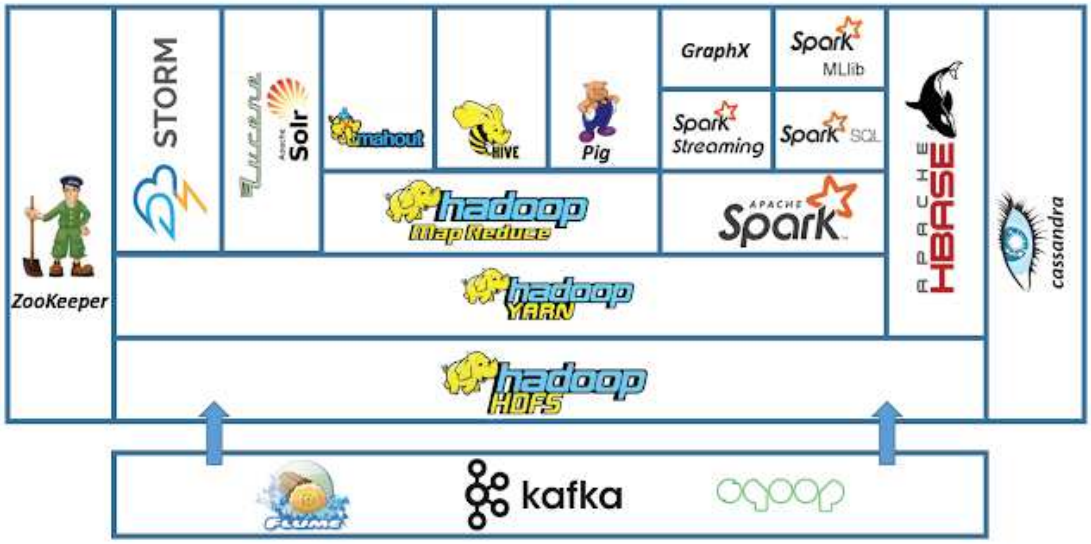


FIGURE *3.23: Un environnement Hadoop*

— Replication factor : C'est le nombre de copies d'une données devant être réparties sur les différents noeuds du cluster (souvent 3, c'est-à-dire une primaire et deux secondaires).

Enfin, un principe important d'HDFS est que les fichiers sont de type *write once* car dans des opérations analytiques on lit la donnée beaucoup plus qu'on l'écrit. C'est donc sur la lecture que les efforts ont été portés. Ce qui signifie que l'on ne modifie pas les données déjà présentes.
Cela a pour conséquence que lorsqu'un fichier HDFS est ouvert en écriture, il est verrouillé pendant toute la durée du traitement. Il est donc impossible d'accéder à des données ou à un résultat tant que le traitement n'est pas terminé et n'a pas fermé le fichier (fichier qui peut être très volumineux dans le cadre des données massives).

MapReduce :
A l'origine crée par Google pour son outil de recherche web. C'est un "framework" qui permet le décomposition d'une requête importante en un ensemble de requêtes plus petites qui vont produire chacune un sous-ensemble du résultat final : c'est la fonction Map.
L'ensemble des résultats est traité (agrégation, filtre) : c'est la fonction Reduce.

Les composants Apache Hadoop :
La figure 3.23 illustre tous les principaux composants d'un environnement (ou distribution) Apache Hadoop.

HBase :

HBase, une base de données NoSQL basée sur HDFS.

Hive :

Hive, une base de données relationnelle basée sur Hadoop, utilisable en SQL et accessible avec JDBC. Hive est à l'origine un projet Facebook qui permet de faire le lien entre le monde SQL et Hadoop. Il permet l'exécution de requêtes SQL sur un cluster Hadoop en vue d'analyser et d'agréger les données.
Le langage SQL est nommé HiveQL. C'est un langage de visualisation uniquement, c'est pourquoi seules les instructions de type "Select" sont supportées pour la manipulation des données. Dans certains cas, les développeurs doivent faire le lien (mapping) entre les structures de données et Hive.

Mahout :

Apache Mahout est un projet de la fondation Apache visant à créer des implémentations d'algorithmes d'apprentissage automatique et de "datamining". Même si les principaux algorithmes d'apprentissage se basent sur MapReduce, il n'y a pas d'obligation à utiliser Hadoop. Apache Mahout ayant été conçu pour pouvoir fonctionner sans cette dépendance.

Pig :

Pig est un outil de scripting basé sur Hadoop permettant de manipuler aisément de grandes quantités de données avec un langage proche du Python ou Bash.
Pig est à l'origine un projet Yahoo qui permet le requêtage des données Hadoop à partir d'un langage de script. Contrairement à Hive, Pig est basé sur un langage de haut niveau "PigLatin" qui permet de créer des programmes de type MapReduce. Contrairement à Hive, Pig ne dispose pas d'interface web.

Oozie :

Oozie est une interface Web de gestion des traitements (job) Hadoop pour les lancer et les planifier aisément en incluant les notions de dépendances de jobs à d'autres jobs. Oozie est une solution de workflow (au sens "scheduler" d'exploitation) utilisée pour gérer et coordonner les tâches de traitement de données à destination de Hadoop.
Oozie s'intègre parfaitement avec l'écosystème Hadoop puisqu'il supporte les types de jobs suivant :

— MapReduce (Java et Streaming) ;

— Pig ;

— Hive ;

— Sqoop.

— Autres jobs tels que programmes Java ou scripts de type Shell.

Sqoop :

Sqoop permet le transfert des données entre un cluster Hadoop et des bases de données relationnelles. C'est un produit développé par Cloudera (voir section 3.6.5).

Il permet d'importer/exporter des données depuis/vers Hadoop et Hive. Pour la manipulation des données Sqoop utilise MapReduce et des drivers JDBC.

Cassandra :

Cassandra est une base de données orientée colonnes développée sous l'impulsion de Facebook. Elle supporte l'exécution de jobs MapReduce qui peuvent y puiser les données en entrée et y stocker les résultats en retour (ou bien dans un système de fichiers).

Cassandra, comparativement à HBase, est plus performante pour les écritures alors que HBase est plus performante pour les lectures.

YARN :

YARN (Yet-Another-Resource-Negotiator) est aussi appelé MapReduce 2.0, ce n'est pas une refonte mais une évolution du framework MapReduce. Il est developpé par Hortonworks (voir section 3.6.5). YARN apporte une séparation claire entre les problématiques suivantes :

— Gestion de l'état du cluster et des ressources.

— Gestion de l'exécution des jobs.

YARN est compatible avec les anciennes versions de MapReduce (il faut simplement recompiler le code)

Apache ZooKeeper :

ZooKeeper est un service de coordination des services d'un cluster Hadoop. En particulier, le rôle de ZooKeeper est de fournir aux composants Hadoop les fonctionnalités de distribution. Pour cela il centralise les éléments de configuration du cluster Hadoop, propose des services de clusterisation et gère la synchronisation des différents éléments (événements).

ZooKeeper est un élément indispensable au bon fonctionnement de HBase.

Ambari :

Ambari est un projet d'incubation Apache initié par HortonWorks et destiné à la supervision et à l'administration de clusters Hadoop. C'est un outil web qui propose un tableau de bord. Cela permet de visualiser rapidement l'état d'un cluster.

Ambari dispose d'un tableau de bord dont le rôle est de fournir une représentation :

— De l'état des services.

— De la configuration du cluster et des services.

— Des informations issues de Ganglia et de Nagios.

— De l'exécution des jobs.

— Des métriques de chaque machine et du cluster.

De plus Ambari inclue un système de gestion de configuration permettant de déployer des services d'Hadoop ou de son écosystème sur des clusters de machines. Ambari se positionne en alternative à Chef, Puppet pour les solutions génériques ou encore à Cloudera Manager (voir section 3.6.5) pour le monde Hadoop.
Ambari ne se limite pas à Hadoop mais permet de gérer également tous les outils de l'écosystème tels que :

— Hadoop

— HDFS

— MapReduce

— Hive, HCatalog

— Oozie

— HBase

— Ganglia, Nagios

Flume :
Flume est une solution de collecte et d'agrégation de fichiers logs, destinés à être stockés et traités par Hadoop. Il a été conçu pour s'interfacer directement avec HDFS au travers d'une API native. Flume est à l'origine un projet Cloudera (voir section 3.6.5), reversé depuis à la fondation Apache. Son alternative est Apache Chukwa.

Apache Drill :
Initié par MapR, Drill est un système distribué permettant d'effectuer des requêtes sur de larges données. Il implémente les concepts exposés par le projet Google Dremel.
Drill permet d'adresser le besoin temps réel d'un projet Hadoop. MapReduce étant plutôt conçu pour traiter de larges volumes de données en "batch" sans objectif de rapidité et sans possibilité de redéfinir la requête à la volée.
Drill est donc un système distribué qui permet l'analyse interactive des données, ce n'est pas un remplacement de MapReduce mais un complément qui est plus adapté pour certains besoins.

Apache HCatalog :
HCatalog est developpé par HortonWorks, il permet l'interopérabilité d'un cluster de données Hadoop

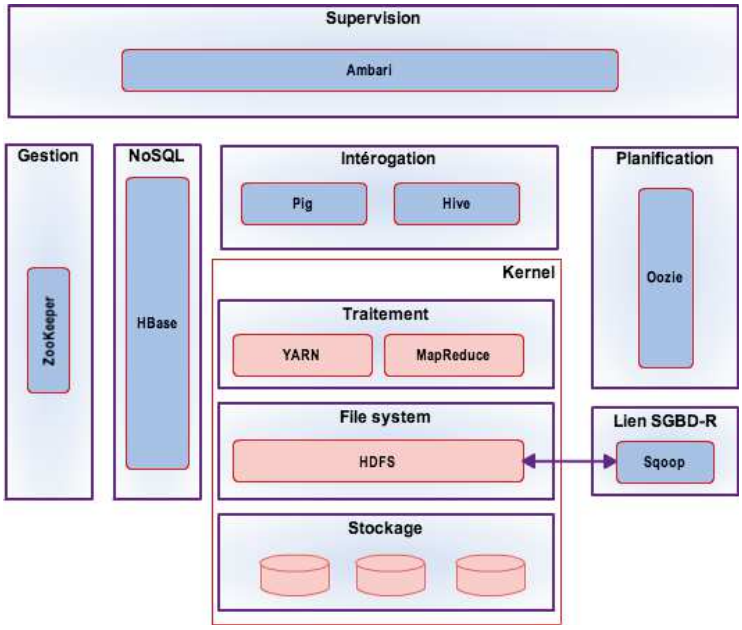


FIGURE *3.24: Vue de la plateforme Apache Hadoop*

avec des systèmes externes. C'est un service de management de tables et de schéma des données Hadoop. Il permet :

— d'attaquer les données HDFS via des schémas de type tables de données en lecture/écriture.

— d'opérer sur des données issues de MapReduce, Pig ou Hive.

Apache Tez :
Tez (Hortonworks) est un nouveau framework en incubation chez Apache. Il utilise YARN et remplace MapReduce afin de fournir des requêtes dites "temps réel". La faible latence est en effet un pré-requis à l'exploration interactive des données stockées sur un cluster Hadoop.
C'est un concurrent d'Apache Drill (MapR) ou de Cloudera Impala.

Les composants du monde Apache Hadoop sont donc très nombreux, avec des fonctionnalités qui se recoupent et il n'est pas aisé de se familiariser avec tous, sans les pratiquer. Ils présentent l'avantage pour la plupart d'être en mode libre (donc non sous licence), ce qui signifie qu'ils sont très attractifs par comparaison aux éditeurs de logiciels "classique". Cependant, certains acteurs du marché, en proposant une plateforme qui regroupe (et intègre) ces composants, de façon simple et graphique, en y ajoutant des améliorations (payantes) qui simplifient son utilisation, et son industrialisation, ont la préférence des organisations voulant utiliser cette technologie.

C'est donc bien un nouveau marché qui se crée autour de cette technologie. Le graphe de prévision de revenu de la figure 3.20 en est la parfaite illustration, avec plus de 87 milliards de revenu associé à cette technologie.
La section suivante détaille les acteurs majeurs de ce marché autour de Apache Hadoop.

3.6.5 Les acteurs industriels autour de Hadoop

Trois distributions majeures se partagent le marché : Cloudera, HortonWorks et MapR, toutes les trois se basant sur Apache Hadoop.On peut toutefois les distinguer en fonction de la distance qu'elles prennent avec cette base :

— MapR : noyau Hadoop mais "repackagé" et enrichi de solutions propriétaires.

— Cloudera : fidèle en grande partie sauf pour les outils d'administration.

— HortonWorks : fidèle à la distribution Apache et donc 100% open source.

Il existe d'autres distributions, voire des offres cloud, mais qui n'offrent pas l'ensemble des fonctionnalités d'une plate forme Hadoop ou ne sont pas open source (ou a minima gratuites) comme Intel Distribution for Hadoop ou bien Greenplum (Pivotal HD).

HortonWorks :
HortonWorks a été formé en juin 2011 par des membres de l'équipe Yahoo en charge du projet Hadoop. Leur but est de faciliter l'adoption de la plate forme Hadoop d'Apache, c'est pourquoi tous les composants sont open source et sous licence Apache. La figure 3.25 illustre les composants qui sont inclus dans cette distribution.
Le modèle économique d'HortonWorks est de ne pas vendre de licence mais uniquement du support et des formations.
Cette distribution est la plus conforme à la plate forme Hadoop d'Apache et HortonWorks est un gros contributeur Hadoop. Parmi les projets reversés il y a :

— ARN ;

— HCatalog ;

— Ambari.

Les éléments suivants composent la plate forme HortonWorks :

— Cœur Hadoop (HDFS/MapReduce) ;

— NoSQL (Apache HBase) ;

— Méta-données (Apache HCatalog) ;

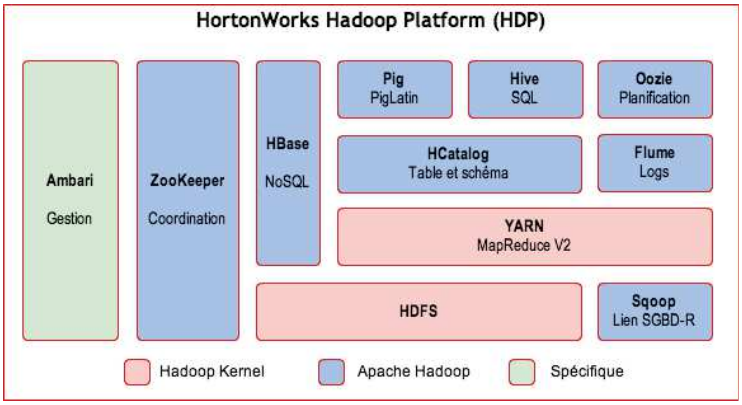


FIGURE *3.25: Distribution HortonWorks*

— Plate forme de script (Apache Pig) ;

— Requêtage (Apache Hive) ;

— Planification (Apache Oozie) ;

— Coordination (Apache Zookeeper) ;

— Gestion et supervision (Apache Ambari) ;

— Services d'intégration (HCatalog APIs, WebHDFS, Talend Open Studio for Big Data, Apache Sqoop) ;

— Gestion distribuée des logs (Apache Flume) ;

— Apprentissage (Apache Mahout).

Cloudera

Cloudera se revendique comme la compagnie commerciale sur la technologie Hadoop. En effet la compagnie a été fondée par des experts Hadoop en provenance de Facebook, Google, Oracle et Yahoo. Si leur plate-forme est en grande partie basée sur Hadoop d'Apache, elle est cependant complétée avec des composants maison essentiellement pour la gestion du cluster. La figure 3.26 illustre les composants qui sont inclus dans cette distribution.
Le modèle économique de Cloudera est la vente de licences mais aussi du support et des formations. Cloudera propose une version entièrement open source de leur plate-forme.

Les composants de la plate forme CDH (Cloudera's Distribution including Apache Hadoop) sont les suivants :

— HDFS : File System distribué ;

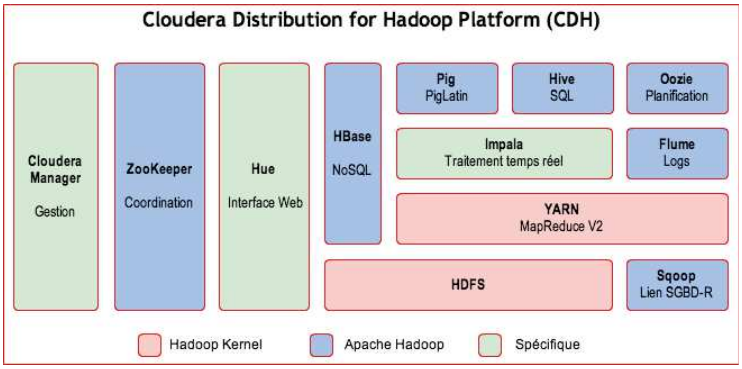


FIGURE *3.26: Distribution Cloudera*

— MapReduce : Framework de traitement parallélisé ;

— HBase : Base de données NoSQL (accès read/write aléatoires) ;

— Hive : Requêtage de type SQL ;

— Pig : Scripting et requêtage Hadoop ;

— Oozie : Workflow et planification de jobs Hadoop ;

— Sqoop : Intégration de bases SQL ;

— Flume : Exploitation de fichiers (log) dans Hadoop ;

— ZooKeeper : Service de coordination pour les applications distribuées ;

— Mahout : Framework d'apprentissage et de datamining pour Hadoop.

— Hadoop Common : Un ensemble d'utilitaires ;

— Hue : SDK permettant de développer des interfaces utilisateur pour les applications Hadoop ;

— Whirr : Librairies et scripts pour l'exécution d'Hadoop et de services liés dans le cloud.

Composants non Apache Hadoop :

— Cloudera Impala : Moteur temps réel de requêtage SQL parallélisé de données stockées dans HDFS ou HBase. Contrairement à Hive de Hadoop, Impala n'utilise pas le framework MapReduce qui exige que les résultats de recherche soient écrits sur le disque, ce qui lui permet d'exécuter les requêtes plus rapidement. La consultation des données peut être interactive.

— Cloudera Manager : Déploiement et gestion des composants Hadoop.

A noter que Cloudera Manager n'est pas entièrement Open Source mais dispose d'une version gratuite avec quelques restrictions :

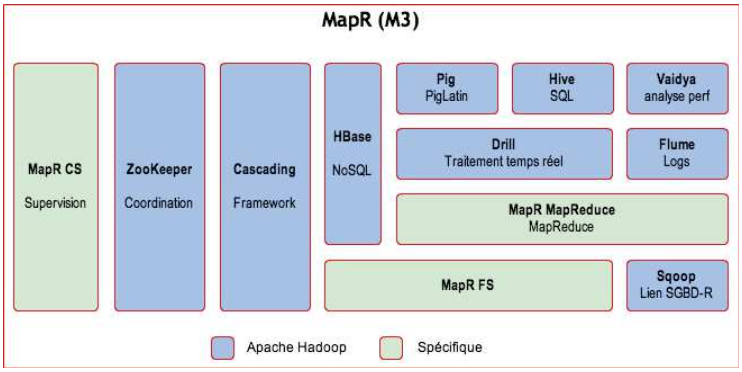


FIGURE *3.27: MapR*

— La version gratuite est limitée à 50 noeuds ;

— Certaines fonctionnalités sont uniquement disponibles sur la version commerciale (comme le monitoring, les sauvegardes et les mises à jour automatiques) ;

— Support uniquement pour la version payante.

MapR :

MapR a été fondée en 2009 par d'anciens membres de Google. Bien que son approche soit commerciale, MapR contribue à des projets Apache Hadoop comme HBase, Pig, Hive, ZooKeeper et surtout Drill. MapR se distingue surtout de la version d'Apache Hadoop par sa prise de distance avec le cœur de la plate-forme. Ils proposent ainsi leur propre système de fichier distribué ainsi que leur propre version de MapReduce : MapR FS et MapR MR.
Trois versions de leur solution sont disponibles :

— M3 : version open source ;

— M5 : Ajoute des fonctions de haute disponibilité et du support ;

— M7 : Environnement HBase optimisé.

MapR a remporté de beaux succès commerciaux depuis sa création. On peut cite le partenariat avec EMC pour une la création et le support d'une version spécifique à la plate-forme Hadoop d'EMC.

— MapR est à l'origine de la version cloud de MapReduce d'Amazon : Elastic Map Reduce (EMR) ;

— Enfin ils ont été retenus par Google pour l'offre Big Data de Google Compute Engine (GCE).

Contenu de la distribution MapR M3 :
Composants Apache :

— HBase ;

— Pig ;
— Hive ;
— Mahout ;
— Cascading ;
— Sqoop ;
— Flume.

MapR propose son propre système en remplacement de HDFS :
— Une version maison de HBase (performance et fiabilité améliorées). Avantages :
— Système plus adapté au mode read/write que HDFS.
— MapR intègre un serveur NFS (Network File System) pour l'intégration au SI de l'entreprise.
— Simplification de mise en œuvre (surcouche du File System de l'OS et non remplacement comme HDFS).
— Plus de Single Point Of Failure.

MapR FS reste compatible avec les API MapReduce/HDFS et HBase. MapR propose son propre système en remplacement de MapReduce d'Apache.
Avantages :
— MapR annonce de meilleures performances ;
— Entièrement optimisé pour HBase.

Ces trois principales distributions que sont Cloudera, HortonWorks et MapR sont les formes de solution Apache Hadoop que les organisations mettent en place pour leurs projets autour des données massives.
Au moment où nous rendons ce mémoire, HortonWorks et Cloudera viennent d'annoncer leur fusion[22]*, signal fort pour le marché lié aux données massives, le choix de distribution Hadoop devient plus restreint, signe aussi d'une évolution du marché.*

La technologie Apache Hadoop est fortement reliée à son traitement des données MapReduce. Ce dernier est parfaitement adapté au traitement des données massives, en mode "batch" et non en temps réel. Selon le cas d'usage qui est attendu, ce dernier n'est pas la solution la plus adéquate. Une autre technologie du monde Apache , Apache Spark, pouvant pallier ce manque, sans pour autant vouloir remplacer et rendre obsolète MapReduce.
Dans la section suivante nous décrivons cette technologie, la comparons avec Apache Hadoop et la positionnons dans le domaine des traitements des données massives.

22. https ://www.zdnet.fr/actualites/cloudera-et-hortonworks-fusionnent-pour-52-milliards-de-dollars-39874547.htm.

FIGURE *3.28: Description et composants de Apache Spark*

3.7 Apache Spark versus Apache Hadoop

Depuis plus de 10 ans, Apache Hadoop est considéré comme la principale technologie de traitement des données massives. Il s'agit effectivement d'une solution adaptée pour le traitement de larges ensembles de données.

Pour les calculs "one-pass", MapReduce est effectivement très efficace, mais se retrouve moins pratique pour les cas d'usage nécessitant des calculs multi-pass et des algorithmes. Pour cause, chaque étape du traitement de données est décomposée entre une phase Map et une phase Reduce (voir figure 3.22).

Entre chaque étape, les données doivent être stockées dans le système de fichier distribué (HDFS) avant que la prochaine étape ne puisse débuter. Dans la pratique, cette approche se révèle très lente. De plus, les solutions Apache Hadoop incluent généralement des "clusters" difficiles à configurer et à gérer. Plusieurs outils doivent également être intégrés pour les différents cas d'usage autour des données massives. Pour le Machine Learning, il faudra par exemple utiliser Mahout. Pour le traitement de flux de données, il sera nécessaire d'intégrer Storm [23]. Ce qui rend complexe l'utilisation de la technologie.

23. Storm est un système informatique distribué qui fonctionne en temps réel, de type FOSS (Free Open Source System), développé par Apache Software Foundation).

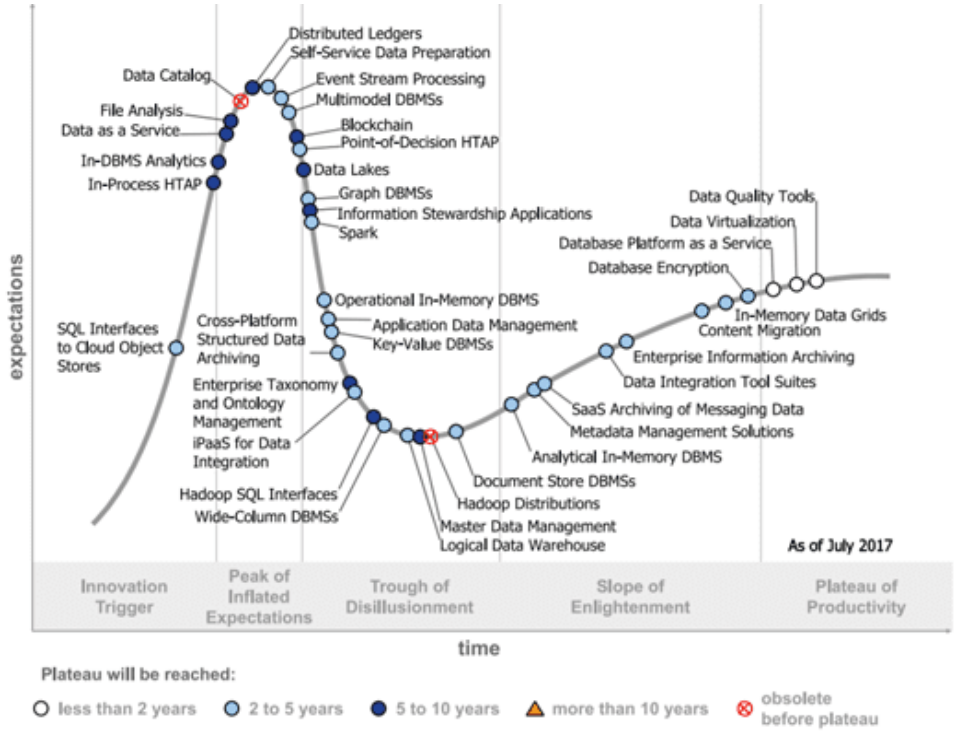


FIGURE *3.29: Courbe 'Hype" du Gartner pour les 2017*

Apache Spark[24] est une nouvelle solution (un framework aussi) qui permet d'offrir une alternative à MapReduce pour cas d'usage nécessitant un moteur de traitement parallèle de données, pour des analyses de grande envergure.

Le principal avantage de Apache Spark est sa vitesse, puisqu'il permet de lancer des programmes 100 fois plus rapidement que Hadoop MapReduce "in-memory", et 10 fois plus vite sur disque. Il est également facile à utiliser, et permet de développer des applications en Java, Scala, Python et R. Son modèle de programmation est plus simple que celui d'Apache Hadoop.

Grâce à plus de 80 opérateurs de haut niveau, le logiciel permet de développer facilement des applications parallèles.

Un autre avantage d'Apache Spark est sa généralité. Il fait à la fois office de moteur de requêtes SQL (Spark SQL), de logiciel de traitement de données en flux (Spark Streaming), et de système de traitement par graphes (GraphX) (voir figure 3.28).

Apache Spark regroupe aussi une grande quantité de bibliothèques d'algorithmes (MLib) pour le Machine Learning. Ces bibliothèques peuvent être combinées en toute simplicité au sein de la même application.

24. http ://spark.apache.org/

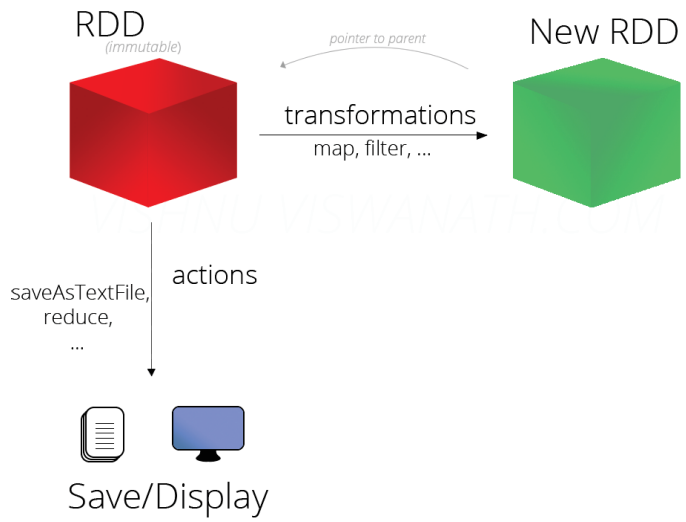


FIGURE *3.30: Mode de fonctionnement d'un Resilient Distributed Datasets -RDD avec Spark*

La figure 3.28 synthétise son architecture avec ces quatre librairies : Spark SQL, Spark ML, GraphX et Spark Streaming.

L'un des grands principes de Spark est l'utilisation des *Resilient Distributed Datasets (**RDD**)*. Un RDD représente une partition des données que l'utilisateur va utiliser pour ses calculs. Cette partition est mise en mémoire (RAM), ainsi cela évite au système de faire des lectures disque à chaque fois qu'une donnée est nécessaire à un calcul. C'est en effet les lectures sur disque à répétition qui ralentissent Hadoop. Si la partition demandée est trop grande, le système ira chercher certaines données sur le disque. Également, il est possible de créer de nouveaux RDDs en en modifiant d'autres. L'API permet également de sauvegarder des RDDs dans HDFS.

Le moteur Apache Spark[25] peut être exécuté sur des clusters Hadoop reposant sur le gestionnaire de ressources YARN, ou sur Mesos. Il est également possible de le lancer sous forme autonome ou sur le "cloud" avec le service Elastic Compute Cloud de Amazon. Il permet d'accéder à diverses sources de données comme HDFS, Cassandra, Hbase et S3. Il peut donc être totalement décorrélé du monde Apache Hadoop.

L'autre point fort de ce moteur est sa communauté massive d'utilisateurs et développeurs. Depuis 2009,

25. https ://www.lebigdata.fr/apache-spark-tout-savoir

plus de 1000 développeurs ont contribué au projet Apache Spark, qui est utilisé par un grand nombre d'entreprises pour le traitement d'ensembles de données volumineux. IBM a même annoncé en 2015 un centre de développement commun avec la communauté Apache Spark[26], preuve de son intérêt pour cette technologie. Quelques mois après IBM a abandonné son développement de distribution Apache Hadoop (IBM BigInsight)[27], pour s'associer avec HortonWorks, preuve de sa confiance en l'adoption massive de cette technologie dans le domaine des données massives. IBM intégre Apache Spark désormais dans tous ces outils autour de la science des données (Data science) et des données massives[28], au travers de son offre Watson Studio notamment.
Plutôt qu'un remplacement d'Apache Hadoop, Apache Spark peut être considéré comme une alternative à Hadoop MapReduce ou un complément. Spark n'a pas pour vocation de remplacer Hadoop, mais de fournir une solution unifiée et compréhensible pour gérer différents cas d'usage liés aux données massives. Il n'y a donc aucun intérêt à vouloir les opposer.

Il est intéressant de regarder l'état des lieux d'utilisation et de perception de Apache Hadoop et de Apache Spark. La courbe "Hype" du Gartner (dont nous avons décrit la lecture dans la section 3.6) de 2017 3.13, positionne Apache Hadoop dans la partie de "désillusion", en attente de voir comment les organisations vont industrialiser (ou pas) cette technologie. Apache Spark se situe à la fin de la zone de "grande attente" du marché. La maturité des deux technologies n'étant pas identique, de l'adoption et l'intégration de Spark par les acteurs du marché des données massives, dépendra son évolution.

Au delà des outils, composants et autres frameworks liés à la mouvance du monde Apache Hadoop, nous étudions dans la section suivante l'impact que ce "nouveau" monde des technologies des données massives a sur les systèmes décisionnels existants.

3.8 L'impact de la technologie Hadoop sur les systèmes décisionnels

Lorsque l'on découvre toutes les possibilités de la technologie de Apache Hadoop, il est tentant de se dire que cette technologie va supplanter tous les systèmes de bases de données classiques (SGBDR), notamment sous la perspective économique mais aussi la puissance de traitement des données de gros volume et non structurées, en particulier.
L'intérêt premier des organisations a donc été d'étudier l'aspect économique de ces solutions basées sur le monde Apache HAdoop versus le modèle économique des systèmes décisionnels existant. C'est d'ailleurs

26. https ://www.ibmbigdatahub.com/video/ibm-spark-technology-center-new-home-what-ifs
27. https ://blogs.gartner.com/merv-adrian/2017/06/21/ibm-ends-hadoop-distribution-hortonworks-expands-hybrid-open-source/
28. https ://www.ibm.com/analytics/data-scienc

la raison principale qui a drainé les premiers projets industriels autour de Apache Hadoop : essayer de remplacer les SGBDR des systèmes décisionnels par Apache Hadoop dans un but de réduire les coûts de licence attenant. Très vite ces projets ont été abandonnés pour plusieurs raisons :

— les compétences des personnes sur la plateforme Apache Hadoop ;

— l'investissement financier déjà fait autour des systèmes décisionnels ;

— les difficultés techniques d'intégration aux outils décisionnels existants et de passage en production.

Apache Hadoop permet d'archiver un grand nombre de données à moindre coût, de stocker et de traiter rapidement, et avec de très bonnes performances, des données structurées et non-structurées en utilisant de l'analytique avancée.
Le système décisionnel, quant à lui, est basé sur une modélisation, ce qui implique quelques contraintes. Il est optimisé pour créer des modèles de données performants, utilisés par les requêtes interactives des outils d'analytique (voir chapitre 3.4). De plus, il supporte très bien la concurrence entre les utilisateurs.

Cependant, et bien que Apache Hadoop offre des fonctionnalités évolutives et de tolérance aux pannes sur les bases de données traditionnelles, il souffre des limitations suivantes [56] :

— Copies en double des données : pour augmenter la disponibilité des données, la réplication des données dans Hadoop peut être utile. Mais cela entraîne également des inefficacités [71]. Les copies multiples de données résidant dans Hadoop entraînent également une dégradation des performances sur des périphériques bon marché (par exemple, des E / S de disque faibles).

— Prise en charge SQL limitée : il existe différents sous-projets, tels que Hive, qui offrent un support de type SQL sur Hadoop. Mais ils n'offrent pas des fonctionnalités SQL complètes telles que des sous-requêtes imbriquées, le support des transactions, etc.[29][28].

— Problèmes de performances : une grande quantité de temps de traitement est consacrée à l'initialisation, à la planification, à la coordination et à la surveillance des tâches. Les facteurs à l'origine de ces problèmes sont le manque d'optimiseur de requêtes et les E / S de disque intensives.

— Structure difficile à gérer : Comme nous l'avons déjà expliqué, Hadoop étant une infrastructure massive avec différents modules, il existe un nombre de paramètres d'optimisation des performances très important, ce qui rend la phase de déploiement du cluster Hadoop, très complexe.

Actuellement, les entreprises possédant des entrepôts de données migrent donc très rarement totalement vers Hadoop et s'orientent plus sur une solution mixte "Data Warehouse / Hadoop".

La mise en place d'une plateforme Apache Hadoop en aval de l'entrepôt de données, par exemple, permet de traiter des données non structurées, d'explorer les données avant de les insérer dans l'entrepôt

de données et ainsi, de s'affranchir des contraintes de l'entrepôt de données (éviter des problèmes organisationnels, consommateurs de temps, tout en donnant de l'agilité à la solution).
Les acteurs du marché de l'entrepôt de données traditionnels, comme Teradata, IBM ou Oracle, l'ont bien compris en créant des partenariats avec des distributions Hadoop comme Hortonworks et Cloudera. Bien que les technologies évoluent rapidement chacune de leur côté, par exemple le temps réel est implémenté dans certains entrepôts de données ou bien des requêtes interactives sont désormais possibles dans l'écosystème Hadoop, via l'arrivée de Kudu, Impala et Drill [29]), ces deux mondes sont donc pour l'instant complémentaires. Cette complémentarité, pour en tirer parti, entraîne donc une évolution au niveau de la conception de l'architecture des systèmes décisionnels qui intègre ces nouvelles possibilités technologiques.

Dans la section suivante nous exposons l'impact de ces nombreuses évolutions technologiques sur les architectures de référence des systèmes décisionnels et comment elles sont prises en compte.

29. https ://kudu.apache.org/overview.html.https ://drill.apache.org/. https ://impala.apache.org/

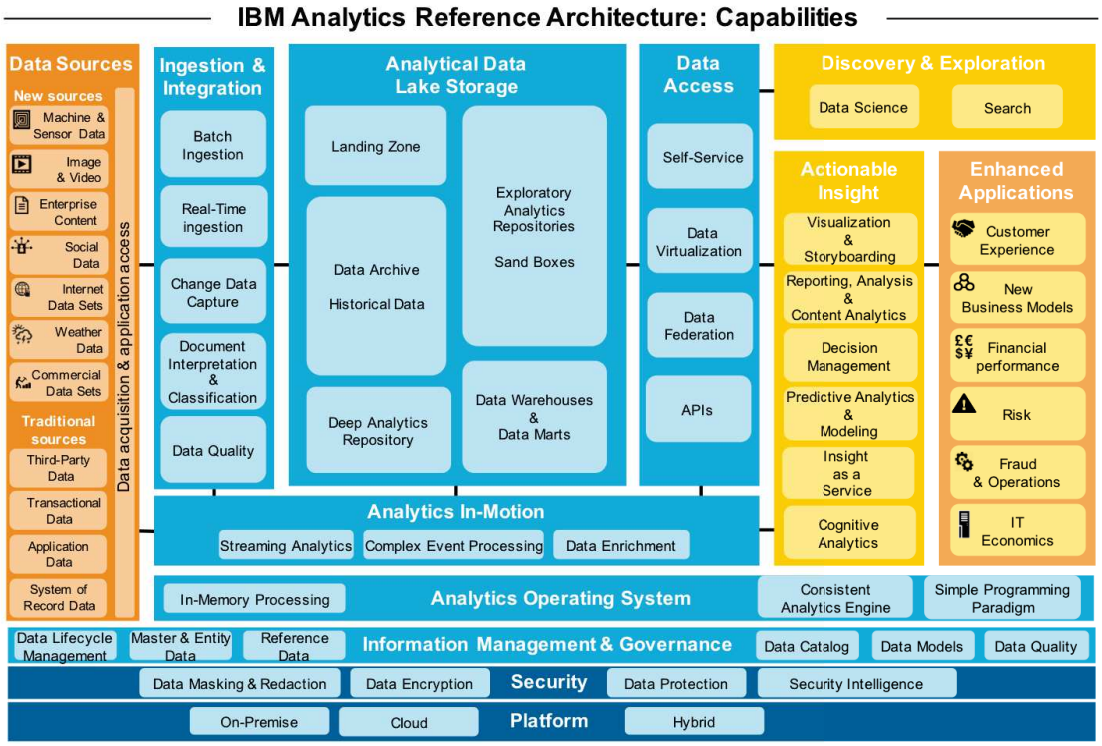


FIGURE *3.31: Architecture de référence analytique d'IBM*

3.9 L'impact de l'évolution des systèmes décisionnels sur les architectures de référence

Les architectures de référence autour des systèmes décisionnels suivent elles aussi les évolutions technologiques et les intègrent au niveau de leurs composants ou en ajoutent de nouveaux. Si on l'on reprend notre exemple de l'architecture de référence d'IBM, détaillée dans la section 2.4 et illustrée au travers la figure 2.8, on retrouve ces évolutions dans la nouvelle version de cette architecture de référence, illustrée dans la figure 3.31.

Au delà de l'aspect visuel et graphique, qui a lui aussi évolué, c'est la notion de SBB (System Building Box) qui s'est transformée en composants. Lorsque l'on détaille chaque composant, on y retrouve désormais toutes les évolutions technologiques que nous avons évoquées dans le chapitre 3.

3.9.1 Les données - data sources

Le composant "Data sources" contient les diverses ressources d'informations qui fournissent des données au système décisionnel. Il comprend les "nouvelles données" et les "traditionnelles".
Les nouvelles sources sont des contenus non structurés, provenant des capteurs (par exemple), des données sociales, des événements météorologiques, du contenu d'entreprise précédemment inexploité, des images et des vidéos.
Les données traditionnelles, sont les sources de données d'entreprise "originelles", qui restent des ressources très précieuses. Elles sont constitués de données d'application des organisations, de données transactionnelles, de systèmes d'enregistrement (données historiques, données de référence, etc.) et de données tierces, fournies par des organisations tierces, par exemple, données de recensement. Cette évolution du composant "données" intègre les évolutions que nous avons détaillées en section 3.3.

3.9.2 L'acquisition

Le composant Ingestion et intégration se concentre sur les processus et les environnements qui traitent de la capture, de la qualification, du traitement et du mouvement des données afin de les stocker et de les utiliser dans le système décisionnel. Ce composant est la passerelle par laquelle les données sont introduites dans le système décisionnel, transformées en un format adapté à la consommation par les outils d'analyse et les utilisateurs, et placées dans des zones de stockage adéquates. Ce composant comporte cinq "sous composants" possibles.

Ingestion par lots ou Batch ingestion : cette fonctionnalité peut être utilisée pour acquérir et préparer des données structurées ou non structurées en mode "batch" (ou traitement par lots), à la demande ou à intervalles réguliers. Cette fonctionnalité comprend des paradigmes standard d'extraction, de transformation et de chargement (ETL) et d'extraction, de chargement et de transformation (ELT), en plus de la préparation et du déplacement manuels des données.

Ingestion en temps réel : cette capacité peut être utilisée pour ingérer et préparer des données structurées ou non structurées provenant de systèmes source en temps réel ou quasi réel. La fonctionnalité traite des données transactionnelles transmises via un "concentrateur" de messages (message Hub) ou un bus de services d'entreprise, ainsi que des données en continu telles que des données de capteurs ou des flux vidéo.

Change Data Capture : Change Data Capture ingère les modifications dans un élément de données particulier, ou de nouvelles versions de cet élément, en temps quasi réel et charge ces modifications dans

une des zones de stockage du système décisionnel prévu à cet effet. Comme les ETL, les technologies de capture de changement de données ont traditionnellement fonctionné sur des données structurées, mais le concept de réplication incrémentielle, en temps quasi réel, d'une source de données vers un référentiel cible peut également s'appliquer aux éléments de données semi-structurés ou non structurés.

L'interprétation et la classification de documents : L'interprétation et la classification de documents consiste à ingérer des données non structurées, telles que des images ou des documents, sous forme physique ou électronique, et à charger le contenu dans une des zones de stockage du système décisionnel prévu à cet effet. Le contenu peut être stocké au format «brut» sous forme d'images ou de documents, ou le contenu peut être analysé lors de l'ingestion et utilisé pour créer des éléments de données plus structurés, éventuellement sous forme de métadonnées pour accompagner le contenu brut. Par exemple, un formulaire numérisé peut être analysé, les champs du formulaire extraits et les données résultantes stockées sous la forme d'un enregistrement structuré dans un référentiel relationnel ou en colonne. Le contenu peut également être interprété pour être classé selon des hiérarchies de documents structurés, appelées taxonomies ou ontologies.

L'analyse de la qualité des données : L'analyse de la qualité des données consiste à mesurer la qualité des données lors de l'ingestion et de la transformation, en appliquant des règles générées et gérées dans le composant "Gestion de l'information et gouvernance" de l'architecture de référence. Des mesures de qualité peuvent être appliquées lors de l'ingestion par lots ou en temps réel, et des analyses peuvent être effectuées périodiquement sur des ensembles de données analytiques. Les mesures de qualité sont ensuite stockées en tant que métadonnées avec les ensembles de données auxquels elles se rapportent et peuvent être examinées par les utilisateurs souhaitant utiliser ces ensembles de données.

3.9.3 Analytique en mouvement - Analytics in motion

Ce composant fournit des fonctionnalités d'analyse en continu. Ce type de fonctionnalité s'applique lorsque des données arrivent non seulement dans le système décisionnel en temps quasi réel, mais doivent également être analysées en temps quasi réel. Les données peuvent ou non être conservées ultérieurement dans une des zones de stockage du système décisionnel prévues à cet effet, pour archivage et/ou pour une analyse d'historique ultérieure. Ce composant intègre des évolutions liées à la fois à l'infrastructure, des logiciels mais aussi des usages, que nous avons décrits dans les sections 3.4, 3.2, 3.1.
Il y a trois "sous-composants" dans ce composant :

L'ingestion en temps réel : L'ingestion en temps réel est la capacité d'accepter des données

dans le système décisionnel, en temps réel (avec gestion et réponse aux événements déterministes) ou presque en temps réel. L'ingestion de données de cette manière est une condition préalable à "Streaming Analytics", comme décrit ci-dessous, mais, selon le cas d'utilisation, les données peuvent simplement être conservées dans une des zones de stockage sans avoir besoin d'analyses de diffusion.

L'analyse en continu - streaming analytics : L'analyse en continu est la capacité à appliquer des modèles analytiques sophistiqués, y compris des capacités cognitives, pour traiter des données, en tirer des informations et lancer des actions en temps réel ou presque. Ces traitements peuvent être des techniques simples de filtrage de données et des modèles de déviation simples, ou des analyses avancées telles que des algorithmes prédictifs hautement complexes. Les données sont généralement analysées sur une fenêtre de temps ou sur un nombre particulier d'événements entrants.

Le traitement d'événements complexes : Le traitement d'événements complexes est la capacité de corréler plusieurs événements non liés, de déterminer des modèles ou des anomalies et d'initier des actions. Cette capacité est conceptuellement similaire, mais généralement moins sophistiquée que le "Streaming Analytics", car elle n'implique généralement pas de modèles analytiques ni de capacités cognitives.

3.9.4 Système d'exploitation du système décisionnel - Analytics Operating System

Le composant Système d'exploitation du système décisionnel fournit un paradigme de programmation cohérent et les fondations dont peuvent tirer parti tous les composants du système décisionnel. Il prend en compte, notamment, l'influence de Apache Spark, que nous avons détaillé dans la section 3.7.

Traitement en mémoire : Câble du moteur de calcul exécutant des travaux de traitement de données parallèles, tous issus de la mémoire système native. Cela permet une analyse informatique à grande vitesse des données massives via des algorithmes itératifs qui partagent de manière fiable les données.

Paradigme de programmation simple : fournir un modèle de programmation général permettant aux développeurs de composer facilement un ensemble diversifié de calculs. Ce modèle de programmation général invite les développeurs à utiliser des programmes parallèles imitant les programmes séquentiels courants, permettant ainsi une expérience de développement familière.

L'utilisation d'un moteur d'analyse unique et cohérent : L'utilisation d'un moteur d'analyse

unique et cohérent pour le mouvement et l'analyse des données tout au long du cycle de vie analytique facilite un traitement cohérent et des résultats cohérents des analyses exploratoires en modélisant et en testant l'opérabilité, en réduisant les tests de régression.

En reprenant ces sous composants avec l'angle technologique Spark, on retrouve les propriétés que l'on veut exploiter dans ce composant :

— Spark fournit un moteur de traitement en mémoire hautement performant qui peut être appliqué à tous les composants de l'architecture de référence, de l'ingestion à l'intégration, en passant par l'exploration et l'analyse des données.

— Le paradigme de programmation de Spark fournit un niveau d'abstraction supérieur à celui d'autres paradigmes tels que Map Reduce, et encapsule les meilleures pratiques, réduisant ainsi la taille du code, améliorant la productivité et l'efficacité et réduisant le temps de rentabilisation.

L'un des aspects les plus importants de Spark est qu'il fournit un framework cohérent qui peut être utilisé dans divers composants du système décisionnel, que ce soit en acquisition ou usages des données.

3.9.5 Zone de stockage du système décisionnel - Analytical Data Lake Storage

Le composant zone de stockage constitue le cœur du système décisionnel. Il couvre les diverses zones dans lesquelles les données sont stockées, pour être consommées par les outils d'analyse et les utilisateurs.
En fonction des types de données stockées et des modèles de consommation pour ces données, les zones vont intégrer certaines des évolutions technologiques que nous avons évoquées précédemment dans cette section.
On va y retrouver les composants "historiques" tels que l'entrepôt de données, l'ODS ou les magasins de données (zone "Data warehouse et Data marts"). On y découvre par contre la nouvelle zone que l'on peut nommer une zone "d'atterrissage des données" (landing zone), liée aux évolutions à la fois des données (voir section 3.3) mais aussi aux technologies comme Apache Hadoop (voir section 3.6).

La zone d'atterrissage : Une zone d'atterrissage est l'endroit où les données sont généralement placées, dans l'attente de subir une adaptation de leur format pour les rendre exploitables, par exemple. En fait, la zone d'atterrissage peut être logiquement divisée en plusieurs sous-zones avec des ensembles de données intermédiaires ou «de transfert» utilisés pour stocker temporairement des données pendant leur transformation, leur intégration et leur préparation à la migration vers d'autres zones. L'utilisation des fichiers HDF de Apache Hadoop peut trouver tout son sens dans cette zone, par exemple (comme

nous l'évoquons dans la section 3.6).

Data archive : Data Archive est une catégorie de zone utilisée pour stocker les données source qui doivent être conservées en raison de la valeur métier ou des exigences réglementaires. Cette zone prend en compte une des évolutions que nous avons décrit en section 3.1.4.
Data Archive est généralement optimisée pour le stockage à faible coût de volumes de données volumineux et pour gérer des opérations d'écriture à haut débit et à grande vitesse, pour une efficacité maximale, et tire souvent parti des capacités de protection des données à des fins d'audit. Les données de la zone Data Archive ne sont généralement pas analysées directement, mais elles sont plutôt extraites et migrées vers d'autres zones pour analyse.

L'historique : L'historique est une catégorie de référentiel utilisée pour stocker des données historiques - historiques de transactions, historiques de changement d'état, etc. À l'instar des catégories déjà mentionnées, une zone d'historique est généralement optimisée pour un stockage à haut volume et à faible coût, et pour des opérations d'écriture efficaces. Les données sont ensuite extraites et migrées vers d'autres zones pour analyse.

Deep Analytics : Deep Analytics est une zone utilisée pour une analyse approfondie de volumes de données extrêmement volumineux, afin d'identifier des modèles, des tendances et des indicateurs. Les référentiels d'analyse approfondie sont également fréquemment utilisés pour former et valider des modèles prédictifs. Étant donné que les volumes de données sont souvent extrêmement volumineux et que la réponse en temps réel n'est généralement pas requise, les plates-formes de stockage à faible coût, telles que Hadoop, sont souvent utilisées pour instancier des zones de Deep Analytics.

Exploration - bac à sable : Cette zone peut être assimilée à un bac à sable, où certains utilisateurs peuvent tester des jeux de données ou des nouvelles technologies.

3.9.6 Accès aux données - data access

Le composant "Data Access" fournit les fonctionnalités permettant d'accéder aux données stockées dans les zones de stockage du système décisionnel et de les mettre à la disposition des utilisateurs via des outils analytiques.
Le composant d'accès aux données comporte quatre ensembles de fonctionnalités distincts :

— La fonction "libre-service" fait référence à la capacité d'un utilisateur averti à rechercher, détecter

et examiner des données disponibles dans les zones de stockage.

— Les capacités de virtualisation des données permettent la découverte, la consultation et l'accès aux données, indépendamment de leur emplacement physique, de leur représentation ou de la plate-forme technologique sur laquelle elles sont stockées. La virtualisation des données permet aux utilisateurs de découvrir et d'explorer des données sans avoir à comprendre des représentations de données spécifiques ni à acquérir des compétences informatiques spécifiques à une plate-forme.

— La fédération de données peut être considérée comme l'un des aspects de la virtualisation des données, dans lequel les données résidant dans plusieurs ensembles de données répartis sur plusieurs sites peuvent être visualisées et consultées sous la forme d'un ensemble de données logique unique. Un exemple en est l'utilisation de vues fédérées couvrant plusieurs bases de données ou ensembles de données.

— Les API ouvertes permettent aux applications d'accéder aux données des différentes zones de stockage, soit via des fonctionnalités et des services appartenant aux catégories mentionnées ci-dessus, soit directement via des mécanismes tels que les protocoles ODBC ou JDBC.

Ces fonctionnalités sont liées à l'intégration des évolutions des infrastructures, que nous évoquons en section 3.2.

3.9.7 Découverte et exploration-Discovery et Exploration

Ce composant tient compte des évolutions des profils des utilisateurs tels que :

— Data Engineer, les ingénieurs des données ;

— Business Analyst : le utilisateurs "classiques" des systèmes décisionnels ;

— App (Application) Développer : les développeurs qui vont être capables d'utiliser les nouvelles technologue telles que Apache Hadoop par exemple ou développer des API ;

— Data scientist : des profils capables de manipuler des algorithmes de science des données (par exemple).

3.9.8 Usages - Actionable Insight

Le composant usages (Actionable Insight) de l'architecture de référence intègre tous les nouveaux (et courant) usages des données autour du système décisionnel, évoqués dans la section 3.4. C'est-à-dire l'analyse descriptive, de diagnostic, prédictive et prescriptive, sous des nominations différentes, mais qui couvrent ces domaines. Ces fonctionnalités peuvent aller de la visualisation de données et d'informations, au reporting et au tableau de bord, à l'analyse prédictive et aux capacités cognitives.

3.9.9 Cas d'usage des données - Enhanced Application

Ce composant recense les principaux cas d'utilisation des données, du point de vue métier, pour lesquelles des applications "clés" en main peuvent être proposées et qui exploitent directement des composants mis en place dans cette architecture de référence.
Le contenu de ce composant est basé sur un rapport conjoint IBM et Gartner qui indique que la majorité des projets autour des données répondent à six impératifs commerciaux dans les organisations :

— Acquérir, développer et fidéliser les clients ;

— Optimiser les opérations, contrer la fraude et les menaces ;

— Maximiser les connaissances, assurer la confiance et améliorer l'économie informatique ;

— Gérer le risque ;

— Créer de nouveaux modèles d'entreprise ;

3.9.10 Gouvernance et gestion de l'information - Information management governance

La gouvernance de l'information fait référence aux disciplines, technologies et solutions utilisées pour gérer les informations au sein d'une entreprise. La gouvernance des données et information comprend trois volets fondamentaux : le cycle de vie, la qualité et la sécurité des données. On y inclut aussi les données de référence de l'entreprise, la gestion des métadonnées et aussi leur catalogue. Nous revenons plus en détails sur ces aspects dans le chapitre 4.

Data Lifecycle Management : Data Lifecycle Management est une discipline qui s'applique non seulement aux données analytiques, mais également aux données opérationnelles, de base et de référence au sein de l'entreprise. Cela implique la définition et la mise en œuvre de politiques sur la création, le stockage, la transmission, l'utilisation et la destruction éventuelle des données, afin de garantir qu'elles soient traitées de manière à être conformes aux exigences commerciales et aux mandats réglementaires. Dans le cadre des données massives, cet aspect est très important, de part la volumétrie des données qui est générée, mais aussi au niveau des processus d'archivage de ces données (voir section 3.1.4).

Les données de base - Master data and Entity : Les données de base et d'entité fournissent aux utilisateurs et aux applications une "source unique de vérité" pour les entités critiques. Ce sont les notions de "Master Data Management" qui sont parfois assimilées aux données de référentiels d'une organisation.

Reference Data : Les données de référence sont un concept similaire aux données de base et aux données d'entité, mais elles se rapportent à des éléments de données communs tels que les codes de localisation, les taux de change, etc., utilisés par plusieurs groupes ou secteurs d'activité au sein de l'entreprise. À l'instar des données principales et des données d'entité, les données de référence sont généralement utilisées par les systèmes opérationnels et analytiques. Elles sont donc généralement stockées en dehors du système décisionnel et accessibles lorsque nécessaire pour l'intégration ou l'analyse des données.

Catalogue de donnée : Data Catalog est un environnement contenant des métadonnées relatives aux données stockées dans les zones de stockage du système décisionnel. Le catalogue conserve la localisation, la signification et le lignage des éléments de données, les relations entre eux et les politiques et règles relatives à leur sécurité et à leur gestion. Le catalogue est essentiel pour permettre une gouvernance efficace de l'information et pour permettre l'accès en libre-service aux données à des fins d'exploration et d'analyse.
Cette notion est clé dans un projet de gouvernance des données, et nous y revenons en détail dans le chapitre 4.

Modèle de données : Les modèles de données fournissent une représentation cohérente des éléments de données et de leurs relations au sein de l'entreprise. Un modèle de données d'entreprise efficace facilite la représentation cohérente des entités et des relations, simplifiant la gestion des données et leur accès.
Dans la section 3.5 nous évoquons ces évolutions de modèle, comme le Data Vault par exemple.

Qualité des données : Les règles de qualité des données décrivent les exigences de qualité pour chaque ensemble de données du composant de stockage et fournissent des mesures de la qualité des données pouvant être utilisées par les consommateurs potentiels de données pour déterminer si un ensemble de données convient à un usage particulier. Par exemple, les ensembles de données obtenus à partir de sources de médias sociaux sont souvent rares et donc de "mauvaise qualité", mais cela n'empêche pas nécessairement l'utilisation d'un ensemble de données. Pourvu qu'un utilisateur des données connaisse sa qualité, il peut utiliser ces connaissances pour déterminer quels types d'algorithmes peuvent être appliqués à ces données. Les règles de qualité des données sont définies et gérées dans le composant Gestion de l'information et gouvernance. Ils peuvent être appliqués à un ensemble de données lors de leur ingestion dans le composant Ingestion et Integration, ou à tout moment après leur ingestion.

3.9.11 La sécurité

La sécurité est une considération cruciale pour les systèmes décisionnels. Les données sont un actif d'entreprise qui doit être correctement sécurisé avec les informations qui en découlent. Chaque entreprise doit protéger ses ressources informatiques afin de s'assurer qu'elles ne sont utilisées de manière appropriée que par les utilisateurs autorisés et qu'elles ne sont pas exposées à des concurrents ou à des criminels. Pour certains types d'informations, il existe également des exigences réglementaires en matière de confidentialité et de sécurité, ainsi que des sanctions significatives en cas de non-conformité.

Le composant de sécurité de l'architecture de référence fournit des fonctionnalités permettant de prendre en charge la sécurité des ressources d'informations dans le système décisionnel. Il y a quatre ensembles de capacités dans ce composant :

Masquage de données : Le masquage et la rédaction des données permettent de modifier les attributs de données sensibles de manière à ne pas exposer leur valeur réelle (masquage) ni supprimer ou masquer les valeurs (rédaction). Cela garantit que les attributs sensibles tels que les numéros d'identification personnels, les numéros de compte, etc., ne sont pas exposés à des utilisateurs non autorisés.
Le cryptage : Le cryptage de données est la capacité de crypter des données soit en stockage (au repos), soit en cours de transmission sur un réseau (en mouvement). Le chiffrement garantit que même si les données sont accédées ou copiées, elles ne peuvent être lues que par un utilisateur ou une application autorisée.
Protection des données : La protection des données (Data Protection) est la capacité à utiliser ces capacités et d'autres pour identifier et protéger les actifs d'informations sensibles, surveiller leur accès et leur utilisation, identifier les anomalies et prévoir les menaces avant qu'elles n'entraînent des violations de sécurité potentiellement coûteuses ou catastrophiques.
Sécurité Intelligente : La sécurité Intelligente (Security Intelligence) est la mise en œuvre de toutes ces fonctionnalités sur une plate-forme commune qui favorise la définition et la mise en œuvre cohérente des stratégies de sécurité dans toute l'entreprise.

Avec la venue de la réglementation autour des données personnelles (RGPD), nous proposon de rajouter un composant dédie à ce domaine CONFIDENTIALITE des DONNEES (Data Privacy).

3.9.12 Les plate-formes et infrastructures

Le composant "Platform" de l'architecture de référence se concentre sur les différents modèles de déploiement (cloud, sur site et hybride) qui prennent en charge le déploiement des autres composants.

Il est important de noter que l'architecture de référence ne présume aucun modèle de déploiement particulier. Ce n'est pas une "architecture cloud" ou une "architecture sur site". Les différents composants et fonctionnalités, ainsi que les différentes offres qui les prennent en charge, peuvent être déployés de différentes manières sur différentes plates-formes pour répondre aux exigences fonctionnelles et non fonctionnelles d'un client.
Comme nous l'avons évoqué dans la section 3.2, l'hybridation au niveau technologique est au cœur de l'évolution des systèmes décisionnels. Ce composant, dans l'architecture prend une importance clé, si l'on compare à l'architecture de référence (voir figure 2.8) d'il y a quelques années. Nous pensons que c'est dans son exploitation que se trouve l'évolution et l'adaptation des systèmes décisionnels d'aujourd'hui.

Un certain nombre de considérations doivent être prises en compte pour décider comment et où déployer différents composants architecturaux, ainsi que plusieurs options pour chacun. Ceux-ci inclut :

— Déploiements sur site, où tous les composants de la solution résident dans le centre de données du client (ou dans plusieurs centres de données dans le cas des grandes entreprises).

— Déploiements dans le cloud, où tous les composants de la solution résident dans un seul centre de données tiers (cloud).

— Des solutions cloud hybrides, dans lesquelles certains composants de la solution peuvent résider dans le centre de données d'un client et d'autres résident dans le cloud, et / ou les composants de la solution résident dans plusieurs centres de données cloud d'un ou de plusieurs fournisseurs de services cloud.

Il convient aussi d'intégrer les possibilités offertes par les gestions complètes d'offre à la fois de service, de plateformes ou de logiciels que l'on peut regrouper sous les possibilités suivantes :

— Services gérés, où un fournisseur de services tiers (à l'organisation) prend en charge l'exploitation et la gestion de la solution, dans un centre de données sur site ou dans un centre de données en nuage appartenant au fournisseur et géré par celui-ci.

— Services hébergés, où le matériel physique, l'infrastructure de stockage et de réseau, et éventuellement les logiciels de plate-forme tels que systèmes d'exploitation, serveurs Web et systèmes de bases de données, sont détenus et gérés par un fournisseur de services dans leur centre de données. le logiciel est la responsabilité du client.

— La plate-forme en tant que service (PaaS) est une forme de service hébergé dans laquelle l'infrastructure est détenue et gérée par le fournisseur de services, mais où le logiciel d'application appartient et est géré (et même éventuellement développé) par le client. Infrastructure en tant que service (IaaS) est un sous-ensemble du PaaS dans lequel seules les infrastructures physiques telles que le matériel, le stockage et les serveurs sont fournies par le fournisseur de services.

— Logiciel en tant que service (SaaS) est un modèle de livraison de logiciel où les logiciels d'application et leur infrastructure sous-jacente sont détenus et gérés par un fournisseur de services, généralement sous licence pour les clients sur la base d'un abonnement. Solution en tant que service (souvent aussi, en confusion, abrégé en SaaS) est un sur-ensemble grâce auquel une solution composée de plusieurs composants est gérée par le fournisseur de services et concédée sous licence aux clients.

Ces possibilités ne sont pas exclusives et toutes peuvent être combinées entre elles selon les exigences techniques et fonctionnelles d'une organisation. Ce qui fait la richesse des possibilités de solutions offertes pour la mise en œuvre et le déploiement d'une architecture décisionnelle.

Au travers cet exemple détaillé d'architecture de référence on peut saisir les évolutions majeures qui s'opèrent dans les systèmes d'information, sous le prisme de l'architecture. Cela démontre l'adaptation phénoménale qui s'est opérée et qui s'opère encore autour des systèmes d'information, sous l'influence des données massives.

Cependant malgré ces adaptations impressionnantes, le système décisionnel reste confronté à certaines limites dans un contexte de données massives et ne peut combler, à lui seul, l'attente des organisations autour de la valorisation de leur capital de données.
Dans la section suivante nous émettons des hypothèses sur ces limites.

3.10 Les limites des systèmes décisionnels

Au travers la section 3 nous avons pu constater que les systèmes décisionnels ne cessent d'évoluer que ce soit au niveau des données qu'ils intègrent, des usages, de l'infrastructure, de la modélisation, des outils logiciels et de leur architecture.
Les systèmes décisionnels ont su et savent s'adapter :

— aux nouveaux besoins d'intégrer des données non structurées et structurées ;

— tirer parti des avancées technologiques pour améliorer leur performance, êtres plus dynamiques et réactifs ;

— intégrer de nouvelles technologies de type Apache Hadoop ;

— adapter leur modélisation pour être réactifs et évolutifs ;

— étendre leur champ d'application, du descriptif au prescriptif.

Pourtant leur principale limite réside en ce qui fait leur force : délivrer l'information attendue. En effet pour que les systèmes décisionnels rendent la fonction pour laquelle ils sont conçus cela suppose une information attendue et définie à la "sortie" du système.
Leur conception est en effet dirigée par ce besoin en information à délivrer. C'est pour cela que l'on peut les qualifier de système ***Information driven***.
Or l'enjeu des organisations autour des données massives et surtout de la valorisation de leur capital de données, c'est de trouver des pistes de nouvelles informations, auxquelles les utilisateurs métiers n'ont pas encore pensé. Ce qui signifie que sans besoin d'information exprimé le système décisionnel est limité pour tirer parti de ces données massives. Sans ces besoins exprimés, les données ne sont pas intégrées.

Les organisations sont donc en recherche d'un système qui lui n'attend pas de savoir quelle information est attendue mais bien que cette information soit trouvée à partir des données existantes.
C'est donc un système non plus dirigé par l'information mais par les données qui est recherché, on parle alors d'un système ***data driven***[30].
Plusieurs travaux académiques académiques, dont ceux de Power [58][59], positionnent un nouveau composant du système d'information en parlant de *data driven Decision Support System*, dont nous émettons l'hypothèse qu'il correspond au système attendu par les organisations pour valoriser leur patrimoine de données.

Dans la littérature scientifique, tout comme dans le monde industriel, ce sont les lacs de donnée (ou data lake) qui ambitionnent de se positionner comme ce nouveau composant du système d'information, dont l'objectif est d'apporter une réponse aux besoins de capitalisation et valorisation des données.

Après avoir établi la provenance de ce système "data driven", nous nous concentrons sur la compréhension de ce nouveau composant du système d'information, son positionnement vis à vis des systèmes décisionnels et les composants fondamentaux de son architecture.

30. https ://www.gartner.com/smarterwithgartner/the-key-to-establishing-a-data-driven-culture/

3.11 Synthèse du chapitre 3

Dans ce chapitre nous avons recensé, sous l'influence des données massives, les éléments des systèmes décisionnels sujets à évolutions :

— les logiciels ;

— l'infrastructure ;

— les données ;

— les usages ;

— la modélisation.

Pour chacun de ces éléments nous avons étudié l'impact sur l'architecture des systèmes décisionnels et comment ces derniers s'adaptent et intègrent leur évolution pour évoluer eux aussi. Nous nous sommes particulièrement intéressés à la technologie Apache Hadoop et aux principaux outils qui la composent afin de mieux appréhender leur valeur ajoutée et ce qu'ils amènent comme changement dans le domaine des système décisionnels.

Au travers cette étude nous avons émis comme principale hypothèse que la limite des systèmes décisionnels sous l'influence des données massives, pour répondre aux besoin de valoriser les données d'une organisation, réside dans le concept même qui les définit : être des systèmes qui délivrent une information connue et définie au préalable, et qui est le guide de leur conception, car ce sont des systèmes dit "information driven".

Nous avons donc constaté que si les systèmes décisionnels apportaient un élément de réponse pour les organisations dans la valorisation d'une partie de leurs données, ils avaient besoin d'être complétés par un autre système dit "data driven", c'est-à-dire dirigé par les données et non plus par l'information.

Les lac de données (ou data lake), nouveau concept très récent semble correspondre à cette attente des organisations.

L'objectif de notre prochain chapitre est de mieux appréhender ce concept, faire un état des lieux des connaissances, donner notre définition des lacs de données, les positionner dans le système d'information, vis-à-vis du système décisionnel.

Chapitre 4

Vers un nouveau modèle d'architecture du système d'information intégrant le concept de lac de données

---

En accord avec plusieurs travaux académiques, dont ceux de Power [58][59], qui positionnent un nouveau composant du système d'information en parlant de *data driven Decision Support System*, nous émettons l'hypothèse que les systèmes attendus par les organisations pour valoriser leur patrimoine de données doivent s'enrichir d'un composant non plus dirigé par l'information mais par les données ***data driven***[1]. Ce composant selon notre vision pourrait correspondre à un lac de données.

Après une rapide synthèse de l'existant sur ce concept, nous donnerons notre définition et développerons notre proposition de modèle de l'architecture globale dans laquelle le lac de données constitue un composant complémentaire.

1. https ://www.gartner.com/smarterwithgartner/the-key-to-establishing-a-data-driven-culture/

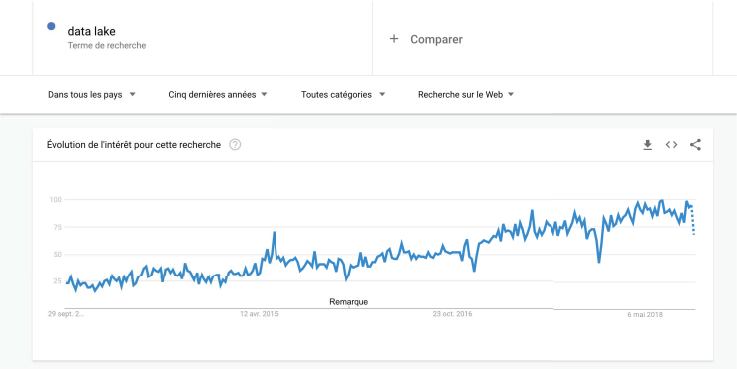


FIGURE *4.1: Recherche d'information sur les lacs de données dans le moteur de recherche Google*

4.1 Introduction au lac de données

Le sujet des lacs de données est un sujet très récent, dont l'intérêt ne cesse de croître comme le montre le graphe de la figure 4.1 qui représente le nombre de fois où le mot "data lake" a été tapé dans le moteur de recherche google au cours de ces cinq dernières années.
Ce côté récent se traduit par une production en littérature scientifique encore limitée. Les premiers travaux académiques, ceux de Fang [17], sur le sujet datent de 2015, et ses références bibliographiques sont au nombre de cinq [2], presque toutes provenant du monde industriel. En effet c'est via le monde industriel que les premières définitions du terme lac de données ont été données.

James Dixon, CTO [3] societé Penthao [15] est celui à qui est attribué la "paternité" du terme et sa première définition. C'est en 2010, dans un blog que Dixon [15] emploie pour la première fois le mot "lac de données". Il y donne en guise de définition l'analogie suivante :

« Si vous considérez un Data mart comme un magasin d'eau en bouteille - nettoyé et emballé et structuré pour une consommation facile - le lac de données est une grande masse d'eau dans un état plus naturel. Le contenu du lac de données s'écoule d'une source pour remplir le lac, et divers utilisateurs du lac peuvent venir examiner, plonger ou prélever des échantillons ».

2. Putting The Data Lake To Work", CITO Research, April 2014 [2] John Monroe, "Predicts 2015 - Managing Data Lakes of Unprecedented Enormity", Garnter, December 2014 [3] Nick Heudecker, "The Data Lake Fallacy : All Water and Little Substance", Garnter, July 2014 [4] Noel Yuhanna, "Market Overview – Big Data Integration", December 2014 [5] Edd Dumbill, "The Data Lake Dream", January 2014
3. Chief Technical Officer

James Dixon voulait que le lac de données devienne un large ensemble de données brutes, structurées ou non, où différents utilisateurs viendraient examiner, scruter les données ou en extraire des échantillons, afin de réaliser des analyses ou dégager des tendances.
En 2014, le Gartner [22] ne voit dans le concept de lac de données que celui d'une nouvelle façon de stocker des données à moindre coût. Pourtant quelques années après, sa position a évolué [4], au regard de l'adoption massive de ce concept dans les entreprises [50] (voir section 4.4). En effet désormais, Gartner positionnent les lacs de données comme le "graal" de la gestion de l'information et le positionne comme le stimulant clé pour créer de l'innovation dans les organisations au travers de la valorisation de leur patrimoine de données.
Au-delà de ces deux positions autour des lacs de données, nous explorons à la fois la littérature scientifique et industrielle pour mieux comprendre les lacs de données, sans prétendre faire un "état de l'art" du sujet, nos exposons une synthèse de ces recherches dans la section suivante.

4.2 État de connaissance des lacs de données - Discussions

Nos premiers travaux [49] sur l'état de connaissance sur les lacs de données nous ont permis de répertorier les principales contributions académiques autour de ce sujet, elles restent encore limitées mais sont en augmentation. Pour preuve les travaux de Ansari [5] autour du profilage sémantique dans les lacs de données (Septembre 2018), sur lesquels nous revenons en aval de ce chapitre.
Les travaux de Fang [17] sont une première prise de position dans la littérature sur les lacs de données, pour lui, un lac de données :

— prend en charge le stockage de données, sous sa forme native, pour un coût faible. Ce coût est faible sur deux aspects, le stockage se fait sur des serveurs dits de "commodité" (technologie X86) et surtout parce qu'il n'est effectué aucun formatage ni nettoyage ni préparation des données (étape très coûteuse généralement),

— stocke une grande variété de types de données, des blobs aux SGBD traditionnels, des données multi-structurées aux données multimédias,

— ne met en forme les données qu'au moment où elles sont exploitées. Cela signifie que les efforts coûteux de modélisation et d'intégration des données sont réduits grâce à cette approche. Cette approche est connue sous le nom de schéma sur lecture ou *schema on read*,

— effectue des analyses basées sur un seul domaine. Comme la valeur est initialement floue, les utilisateurs doivent donc développer des analyses particulières pour utiliser les données,

— les stratégies de gouvernance sont configurées pour identifier, réutiliser et éliminer les données,

4. https ://www.gartner.com/webinar/3745620

— indique la provenance des données, dont les indications sur leur source, leur origine, qui les a modifiées, quelle est la version de leur changement, etc.

Pour Fang [17], il n'y a pas d'architecture de base particulière d'un lac de données et il associe fortement la création d'un lac de données avec la mise en place d'un environnement Apache Hadoop (voir section3.6). De plus, il voit le déclin du système décisionnel au profit du lac de données, qu'il voit même à terme dans les "nuages".
Dans ses travaux, le lac de données est vu comme une méthodologie des gestion de données où toutes les données d'une organisation sont rassemblées, physiquement, sur une plate-forme à base de Apache Hadoop.
Il alerte sur le risque de non gouvernance des données, comme tout projet de gestion de données classique. Nous voyons quatre limites [49] à la vision de Fang sur les lacs de données :

— elle est centrée sur la technologie Apache Hadoop exclusivement,

— elle ne prend pas en compte le fait que certains critères pourraient "bloquer" le mouvement des données, tels la gravité des données (voir le chapitre 5),

— l'axe gouvernance est découplé d'un lac de données,

— le lac de données est vu comme le "tueur" des entrepôts de données.

Dans [17], il n'y a pas de proposition autour de l'architecture d'un lac de données. Dans le cadre de nos travaux, nous regardons vers le monde industriel pour examiner les architectures étudiées ou mises en place, et essayons de trouver leurs points de convergence et divergence pour établir notre point de vue.
Les travaux autour de l'architecture dans le monde industriel sont pilotées par des vendeurs de logiciels, liés à la technologie Hadoop, tels que Cloudera, HortonWorks ou MapR (voir section 3.6)et leur vision d'architecture fortement influencée par leur plate-forme.

— elle est centrée sur la technologie Apache Hadoop exclusivement ;

— les données sont rassemblées au sein de la même plate-forme ;

— l'axe gouvernance est peu développé avec une proposition d'outils très limités ;

— le lac de données est vu comme le remplaçant des entrepôts de données.

L'éditeur IBM [10] est le seul à proposer, au travers ses travaux une vision d'architecture élaborée et mature, certes orientée par les produits logiciels qu'il promeut, mais qui aborde un certain nombre des problématiques qui nous questionnent. Les premiers travaux des lacs de données d'IBM n'utilisent pas le mot "data lake" mais "data reservoir" (réservoir de données [77]), dénomination qui sous la pression et l'engouement marketing du mot "lac de données" sera abandonné. Pourtant cette dénomination de "réservoir de données" est un indicateur des premiers travaux industriels qui commencent à mettre en doute,

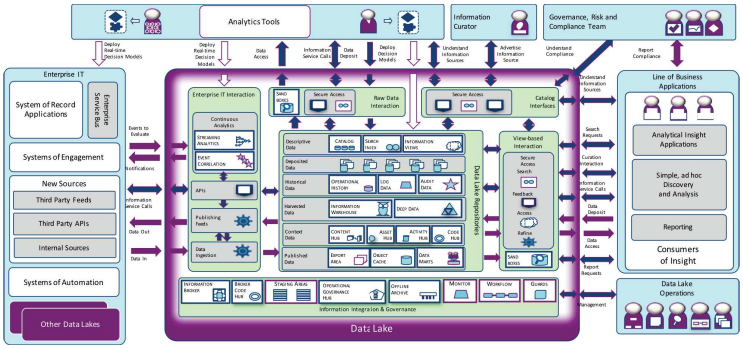


FIGURE 4.2: *Architecture de Référence d'un lac de données proposée par IBM [10]*

rapidement, la technologie Apache Hadoop comme seule réponse à la mise en œuvre des lacs de données.

En 2014, Bill Inmon [34] publie un livre sur l'architecture des lacs de données, afin selon lui d'éviter de stocker des données inutiles et impossibles à utiliser. Il prêche pour une évolution des architectures des lacs de données vers le modèle des systèmes décisionnels (qu'il connaît parfaitement mais qu'il ne nomme pas explicitement), pour éviter de stocker des données seulement brutes mais des données déjà "raffinées" avec des processus de type E.T.L. Il met en exergue l'utilisation des métadonnées, évoque le profil des utilisateurs des lacs de données (les data scientists) et propose une organisation des types de données ("analog data", "application data", "textual data") à l'intérieur du lac de données. Il focalise ses propositions sur cette catégorisation des données à intégrer dans le lac de données. Aucune remise en compte ou questionnement sur le mode de stockage de ces données, qui n'est pas son point de réflexion dans cet ouvrage.

C'est Russom [66] [67] qui le premier amorce les limites de la technologie Hadoop, comme seule réponse aux lacs de données et propose une hybridation de technologies, pour répondre à certaines limites techniques mais aussi de conception que rencontrent les lacs de données basés exclusivement sur la technologie Apache Hadoop.
Selon Russom [66] [67], il existe deux grands types de lacs de données : les lacs de données basés sur Apache Hadoop et les lacs de données basés sur les bases relationnelles. Même si aujourd'hui, Apache Hadoop est beaucoup plus commun que les bases de données relationnelles en tant que plate-forme pour les lacs de données, un quart des organisations que Russom a interrogées dans ses travaux disent que leur lac de données tire parti des deux.
Les lacs de données, commencent donc à évoluer, deux ans après les premiers travaux de Fang [17] certains sont déjà multi-plate-formes et hybrides, tout comme les entrepôts de données actuels (voir

section 3.2).

IBM [10] fait évoluer la connaissance des lacs de données, au niveau de la gouvernance des données et surtout dans la prévention des risques pour éviter que le lac de données ne se transforme en marécage (cf. 4.8.5). Pour cela, IBM met en exergue l'importance capitale d'un composant du lac de données : son catalogue de métadonnées (cf. 4.8.2). L'emphase est tellement forte, et le composant si crucial, que les travaux actuels autour des lacs de données, dans la littérature scientifique, s'y concentrent.
Les métadonnées (et leur catalogue) (cf. 4.8.2 et 4.8.5), sont un des sujets les plus étudiés dans les travaux en cours ou achevés sur le sujet des lacs de données.

L'aspect modélisation, et optimisation du modèle de métadonnées est au centre des travaux de Nogueira, Romdhane et Darmont[55], avec une proposition de mise en oeuvre du Data vault (cf. section 3.5) pour stocker les métadonnées du lac de données.

Dans leurs travaux Terizzano [75] définissent le lac de données comme un référentiel central contenant des quantités énormes de données brutes, issues de sources de données multiples, que les utilisateurs autorisés peuvent facilement utiliser pour plusieurs activités analytiques indépendantes. Ils identifient l'importance de la description des données comme l'un des principaux défis dans les lacs de données (donc les métadonnées). Ils proposent d'aller au-delà de la description de la structure des données ingérées et de couvrir également la signification sémantique des données en utilisant des ontologies et des vocabulaires de domaine.

Dans leurs travaux Hai et al. [26] proposent une solution de gestion des métadonnées du lac de données appelée Constance. Constance se concentre sur la découverte, l'extraction, la synthèse de métadonnées structurelles et l'annotation de données et de métadonnées avec des informations sémantiques. Le composant de Constance qui gère les métadonnées sémantiques est le jumelage sémantique des métadonnées. Il comporte une modélisation ontologique, un couplage de données et une annotation sémantique.

Dans ses travaux, Ansari propose une approche de profilage sémantique pour les lacs de données [5], afin d'éviter que le lac ne se transforme en "marécage" de données. Il démontre comment le "Web sémantique" peut améliorer la ré-utilisabilité et la détection des données ingérées dans le lac de données. Ces différents travaux montrent l'importance de la prise en compte de la gestion des métadonnées dans la conception des lacs de données.

Dans nos travaux nous abordons, dans les sections 4.8.2 et 4.8.5, la gestion des métadonnées mais n'en faisons pas le sujet principal de nos travaux.
Nos travaux se positionnent donc en complément de ceux existants ou en cours sur le sujet de la gestion des métadonnées des lacs de données.

Alrehamy et al. [3] présentent leurs travaux sur un "Personal Data Lake", un lac de données central pour analyser et interroger des données personnelles. Ce travail se concentre principalement sur les problèmes de confidentialité des données individuelles. La position prise par Alrehamy et al. [3] de concentrer les données en un seul endroit pour optimiser la gestion et la sécurité est intéressante, le lac de données étant positionné comme le support pour le stockage de ces données. Ces travaux soulèvent le problème de la confidentialité des données, qui sous l'aspect de la réglementation RGPD[5] devient un aspect crucial des lacs de données. En effet, certaines organisations, créent un lac de données pour rassembler toutes les données autour de la connaissance de leurs clients (par exemple), ce qui entraîne une contrainte supplémentaire sur le lac de données ; celui de la protection et de la sécurité des données. Même si les travaux de Alrehamy et al. [3] n'étudient pas le problème de la confidentialité des données d'un lac de données dans son ensemble, ils soulèvent un point d'attention autour de la sécurité des données personnelles lors de sa conception.

Les travaux de Plale et Suriarachchi [73] sont les premiers à discuter de l'aspect architecture et infrastructure des lacs de données dans la littérature scientifique. Ils s'intéressent à l'intégration des données dans un lac de données, notamment leur lignage et la traçabilité des données originelles jusqu'à leur transformations dans le lac. Ils proposent une architecture de référence pour la gestion de cette traçabilité, dans un contexte de données massives. Ils évaluent leur architecture à travers un prototype construit à l'aide d'outils provenant du monde Apache Hadoop notamment Hadoop (HDFS), Spark et Storm (voir section 3.6). Leur proposition d'architecture, qui est illustrée dans la figure 4.3, reprend quelques éléments de celle d'IBM [10] qui est illustrée en figure 4.2. Cette approche complète, au moyen d'une proposition d'architecture, les précédents travaux cités, sur la gestion des métadonnées. Elle y ajoute la notion de traçabilité des données et de leurs transformations.

Les travaux les plus aboutis sur l'aspect architecture, composants et positionnement des lacs de données sont ceux d'IBM [10]. Ils sont certes dictés par les produits logiciels que l'éditeur veut mettre en évidence, avec un accent fort sur l'aspect gouvernance des données, en particulier l'aspect " catalogue" des métadonnées.

5. Le règlement européen sur la protection des données ou RGPD qui est une directive européenne qui oblige toutes les entreprises et les administrations, à respecter certaines règles concernant le traitement des données à caractère personnel.

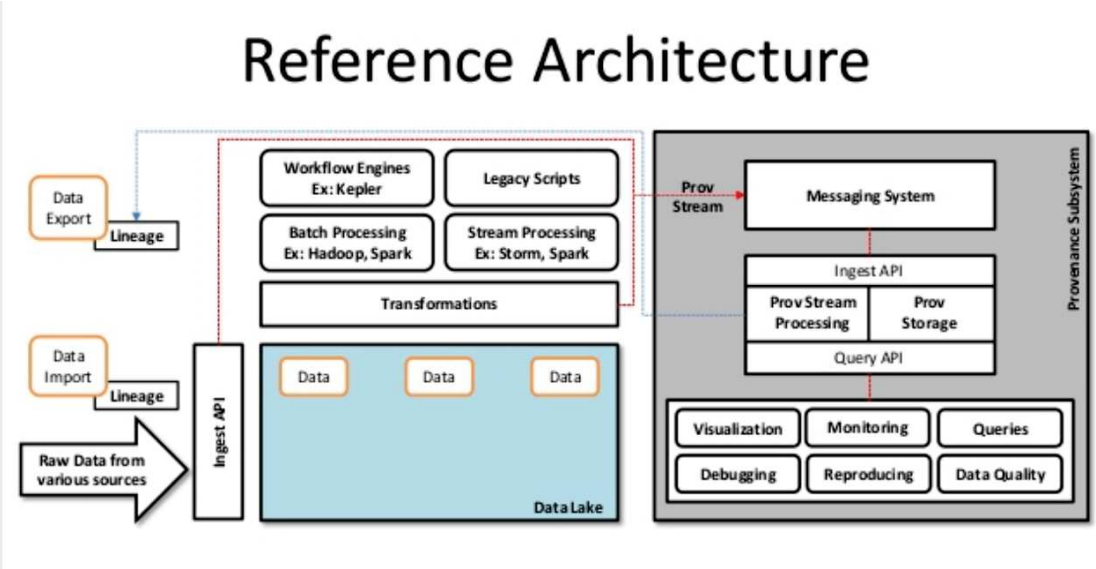


FIGURE *4.3: Architecture de Référence d'un lac de données proposée par [73]*

Les positions d'IBM [10] sur l'aspect stockage mono-technologique (Apache Hadoop), commencent à évoluer, vers une ouverture au mode hybride à la fois pluriel sur les modes de stockage (relationnel, NoSQL, Apache Hadoop, etc.) mais aussi au niveau des plate-formes (différents serveurs, sur site ou dans les nuages). Cette évolution est celle que retranscrit Russom [66] dans ses constatations auprès des organisations qui mettent en place des lacs de données en mélangeant différentes technologies.
Dans leurs travaux Miloslavskaya et Tolstoy [53], reprennent les grands principes du phénomène des données massives, les fameux "3V" cités par le Gartner [21], y rajoutent le quatrième V proposé par IBM (voir figure 3.9) pour la véracité, mais aussi deux autres V : variabilité, valeur et visibilité.
Ils amorcent un positionnement des lacs de données, comme devant être inclus dans le reste des systèmes "IT", et s'attachent à étudier trois modes d'acquisitions des données que sont les modes "batch" pseudo temps réel, le temps réel (streaming) et l'hybride des deux, pour en conclure que toutes les données volumineuses ne sont pas rapides et que toutes les données rapides ne sont pas volumineuses.
L'aspect architecture n'est pas abordé mais cité en perspective au vu de l'influence de ces modes d'acquisitions différents.

Leur vision des lacs de données est déjà une évolution de la vision de Fang [17], leur travaux datant de 2016 et ceux de Fang, de 2015. Ils imaginent le lac de données comme un grand "pool" de données rassemblant l'ensemble des données historiques, données accumulées et nouvelles données produites en temps quasi réel, dans un seul et même endroit, dans lequel le schéma des données n'est pas défini

tant que les données ne sont pas interrogées. Les lacs de données doivent être gérés et protégés et posséder des architectures proposant des évolutions d'échelle à haute disponibilité. Les données doivent être gérées au sein d'un catalogue centralisés où le lignage de données est indiqué.
Le cœur de ces travaux [53] n'est pas le sujet des métadonnées, même s'ils citent l'importance de la gestion des métadonnées.
Dans leur travaux, la vision du lac de données reste encore centrée sur un seul endroit "physique/technique" qui regroupe toutes les données acquises et n'intègre pas encore l'aspect hybridation technologique comme le suggère Russom [66].

Dans cet état des lieux des connaissances sur les lacs de données, lié à notre analyse des différents travaux sur le sujet, nous avons listé les mots clés sur lesquels il y a ou peut avoir convergence :

Données massives, stockage, données structurées et non structurées, sémantique, gouvernance, métadonnées, données brutes, accès, exploration, "schema "on read" et sécurité.

Le positionnement vis-à-vis des systèmes décisionnels reste un point de divergence dans certains travaux. Nous prenons position sur cet aspect dans la section 4.6.

L'aspect duplication physique de toutes les données dans un seul environnement physique, mono-technologique reste lui aussi encore un point de divergence, mais qui évolue. Nous prenons aussi position sur cet aspect en étudiant dans le chapitre 5.1 un facteur qui nous semble remettre en question cet l'angle mono-technologique, celui de la gravité des données.

Ces différentes analyses sur les différents travaux et nos propres expériences nous amènent à proposer pour les lacs de données, la définition que nous donnons dans la section suivante.

4.3 Notre définition des lacs de données

Basé sur cet état des connaissance et notre point de vue, nous amorçons notre définition d'un lac de données[49] :
Le lac de données est une collection de données/ ou un ensemble de donnée qui sont :

- **Indépendantes d'un schéma d'information pré établi,**
- **De formats non contraints (tous formats acceptés),**
- **Non transformées,**

- **Conceptuellement rassemblées en un endroit unique mais potentiellement non matérialisées,**
- **Destinées à un ou des utilisateurs experts en science des données,**
- **Munies d'un catalogue de méta-données,**
- **Munies d'un ensemble de règles et méthodes de gouvernance de données.**

L'objectif premier du lac de données est de permettre l'exploration, sans a priori, de l'ensemble de données qui le composent, en vue de découvrir des nouvelles pistes d'information à exploiter dans le contexte d'une valorisation des données d'une organisation. Le lac de données est un système dirigé par les données qui vient compléter le système décisionnel en place. Il devient un composant incontournable du système d'information, sous l'influence des données massives. Nous résumons notre point de vue dans la figure 4.15.

Mieux appréhender le concept des lacs de données, son positionnement dans le système d'information, passe par la compréhension de leurs enjeux que ce soit au niveau de l'organisation elle-même mais aussi dans le milieu industriel qui le met en place.

Dans la section suivante nous présentons la synthèse de nos travaux de recherche sur les enjeux des lacs de données.

4.4 Les enjeux des lacs de données

Les entreprises mettent au cœur de leur transformation digitale la capitalisation de leurs données pour les valoriser ultérieurement. Les lacs de données se positionnent comme une des solutions qui permet de valoriser le capital données des organisations. Ils sont conçus pour pouvoir cataloguer, capturer, stocker, exploiter, manipuler et gérer une masse de données disponibles. D'abord perçus seulement comme des environnements de stockage à bas coût [22], le potentiel de valorisation des données qu'ils stockent s'est transformé en enjeu stratégique pour les organisations [51].

Le tableau ci-dessus illustre le marché potentiel que représente ce sujet pour des organisations au niveau mondial et l'importance de ce sujet dans le monde industriel.
On constate que le marché des lacs de données est estimé autour de 8,81 milliards de dollars d'ici 2021, qu'il est en progression (32,1 % à l'horizon de 2021). L'enjeu des lacs de données est bien réel et les attentes de retour sur investissements importantes.
L'adoption du concept de lac de données s'est accélérée avec la convergence du besoin de plate-formes fédératrices dans les entreprises, pour faciliter l'exploitation des données, et les possibilités technologiques

Data Lakes Market	2014	2015	2016-e	2021-p	CAGR (2016-2021)
Market Size	1.82	2.12	2.53	8.81	28.3%
Y-o-Y		16.3%	19.5%	32.1%	

e – Estimated; p – Projected

Note: Y-o-Y for the year 2021has been calculated from 2020-2021

Source: Press Releases, Investor Presentations, Expert Interviews, and MarketsandMarkets Analysis

FIGURE *4.4: Évolution du marché des solutions des lacs de données*

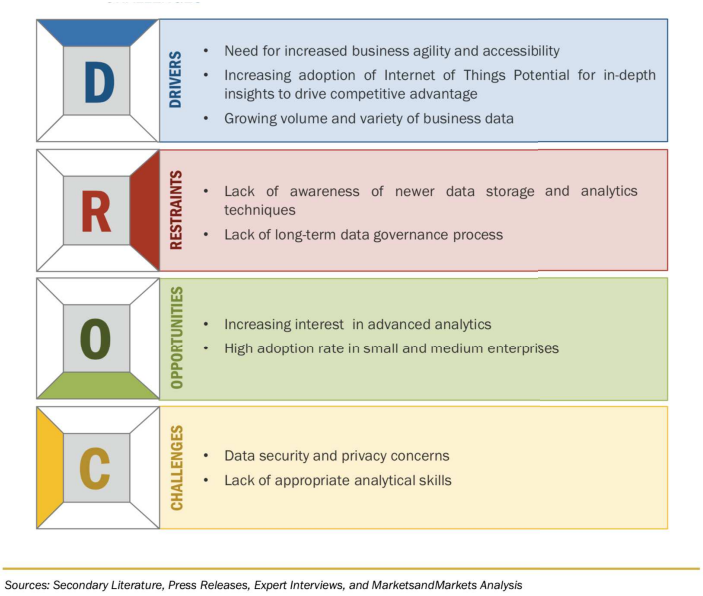


FIGURE *4.5: Marché des lacs de données : Moteurs, freins, opportunités et défis*

autour de certaines technologies associées aux « masses de données ».

Le rapport [51] d'analyse du marché des lacs de données indique que le principal frein à la croissance du marché des lacs de données est le manque de sensibilisation aux nouvelles techniques de stockage, d'analyse des données et le manque de processus de gouvernance à long terme des données.

Ce rapport identifie aussi la sécurité des données, leur confidentialité et le manque de compétences analytiques appropriées comme les principaux défis de la croissance du marché des Data Lakes. Le tableau 4.5 résume les moteurs, freins, opportunités et défis du marché des lacs de données.

Le besoin des organisations d'être agiles, l'accentuation des données émises associée au besoin de ces

mêmes organisations de tirer parti de ces nouvelles données à tous les niveaux, sont les principaux moteurs de création des lacs de données et de leur expansion.
Le manque de stratégie de gouvernance appliqués à ces lacs, mais aussi le manque de maîtrise des nouvelles techniques d'analytiques disponibles (telles que l'analyse prédictive ou prescriptive) sont les principaux freins à l'adoption massive des lacs de données.
La disponibilité de ces nouvelles techniques analytiques représentent cependant la plus belle des opportunités pour mettre en valeur les lacs de données. Ce taux d'adoption et de mise en pratique de ces nouvelles techniques est plus grand dans les organisations de taille moyenne que dans les grandes, qui éprouvent plus de difficulté à se transformer.
Enfin les défis des lacs de données résident sur les points de sécurité, autour de la protection des données (sensibilité de certaines) et bien sûr la confidentialité. Le manque de compétence autour du sujet des lacs est aussi un vrai défi aujourd'hui pour les organisations qui veulent mettre en place les lacs de données.

A la lumière de ce rapport, nous constatons que les lacs de données sont un véritable enjeu pour les organisations, quelle que soit leur taille, loin du phénomène marketing que l'on pouvait peut-être penser qu'il était. Désormais c'est un incontournable dans les organisations et nous intéressons à son positionnement vis-à-vis des différents systèmes, selon la définition de Le Moigne (voir figure 2.3).
L'objectif de la prochaine section est de faire une proposition de positionnement des lacs de données, basée sur la définition des systèmes de Le Moigne [38].

4.5 Proposition d'un modèle pour les systèmes d'information avec des lacs de données

Sous la mouvance du phénomène des données massives, les projets de lac de données sont associés à des projets de type "analytique", comme les projets décisionnels dans les organisations. Or les différents travaux [5][17][10][26][66][55] et écrits sur ce domaine mettent en exergue les points suivants :

— l'importance d'avoir une gestion des métadonnées,

— l'importance de gérer la sécurité et la confidentialité des données,

— l'importance de mettre en place le processus de gestion du cycle de vie des données,

— l'importance de la traçabilité des données et des traitements.

Tous ces points relèvent de l'aspect gouvernance des données, plus que de l'analytique. De notre point de vue les lacs de données sont donc plus à associer à un projet de gouvernance de données qu'un projet de type décisionnel.

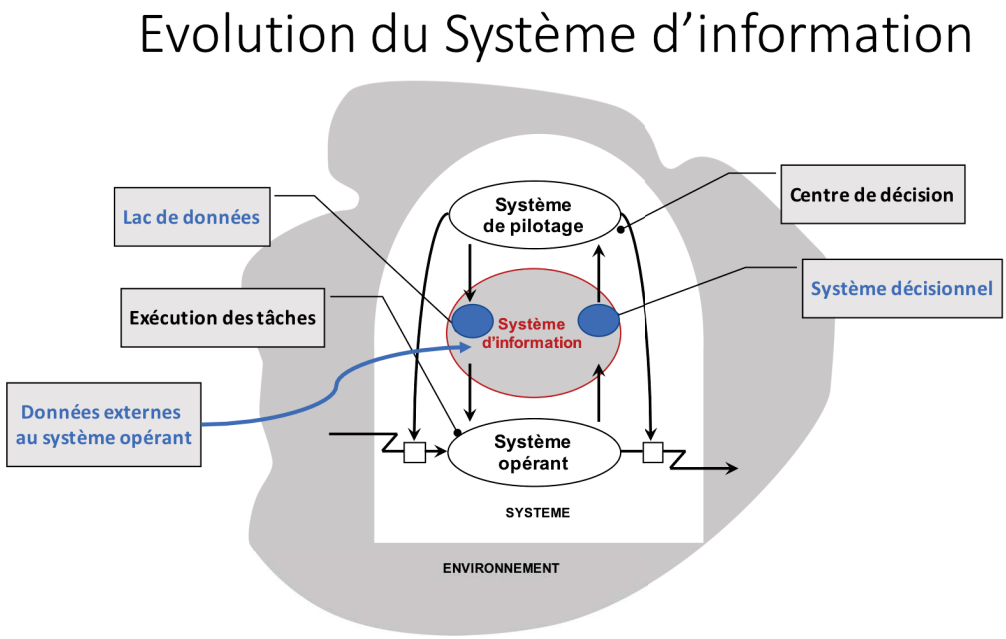


FIGURE *4.6: Proposition de positionnement des Lacs de données dans le Système d'information*

La fonction première d'un lac de données est d'être un système dirigé par les données, *Data driven*, il a donc une fonction différente du système décisionnel, qui lui est dirigé par l'information, *Information driven*.
Il a de plus comme objectif d'aider au système de pilotage d'une organisation en capitalisant une partie des données provenant du système opérant.

Sa place est donc **dans le système d'information, au côté des systèmes décisionnels.**

Nous schématisons notre proposition dans la figure 4.7, en se basant sur la description de Le Moigne, et en faisant une évolution des composants du système d'information. Dans cette illustration 4.6 nous ajoutons aussi la notion de données "externes" provenant de l'extérieur, c'est à dire en dehors du système opérant. Cet ajout est lié à la prise en compte des données massives.

De notre point de vue le lac de données est donc un nouveau composant du système d'information.

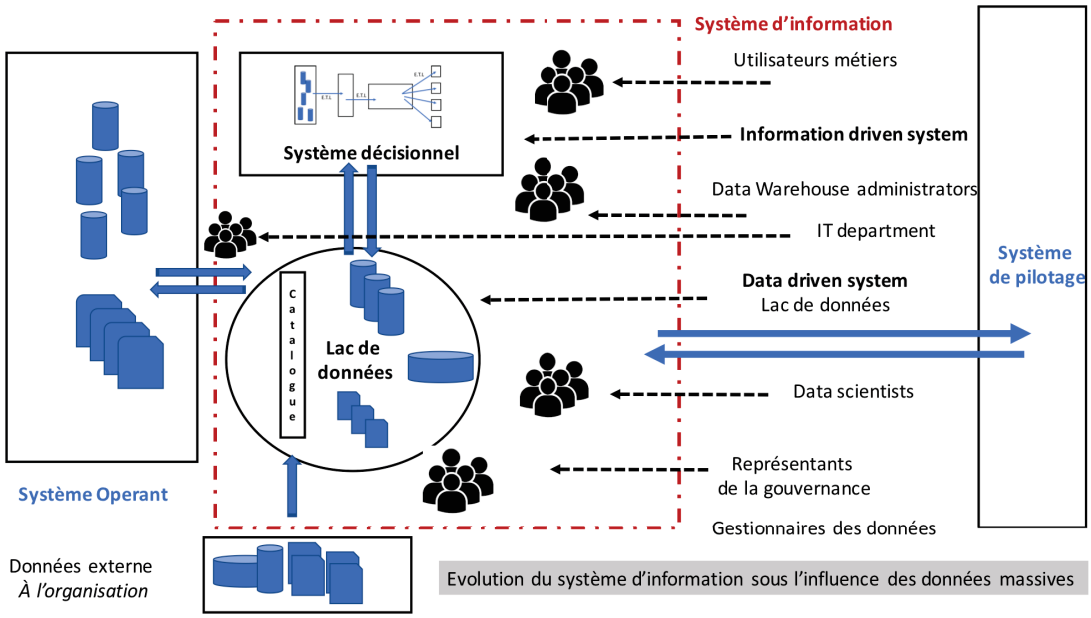


FIGURE *4.7: Interaction des lacs de données dans les systèmes d'une organisation*

Dans la figure 4.7 nous illustrons de façon plus fine les interactions du lac de données avec le système décisionnel (nous approfondissons ce point dans la section 4.6), mais aussi avec le système opérant et de pilotage, ainsi qu'avec les différents acteurs/utilisateurs du système d'information.

Nous détaillons dans la section suivante la position du lac de données vis-à-vis du système décisionnel, au sein du système d'information.

4.6 Les lacs de données vis-à-vis des systèmes décisionnels

Dans la section précédente nous avons pris position d'insérer les lacs de données comme un nouveau composant du système d'information, et l'avons positionné au côté du système décisionnel. Il nous semble important de bien préciser, au travers cette section quelles sont les différences entre ces deux composants qui justifient notre différentiation.

4.6.1 L'acquisition des données

Dans certains travaux [17] Les lacs de données sont souvent comparés aux systèmes décisionnels, notamment les entrepôts de données, car ils permettent de stocker des masses de données dans le but

de les transformer en information. On attend pourtant des lacs de données de proposer une plus grande flexibilité que les systèmes décisionnels en n'imposant pas un schéma strict aux flux de données entrants. En effet comme le lac de données n'est pas conçu dans le but de délivrer une information définie au préalable, il n'impose pas de schéma de flux d'entrée aux données. Le lac de données permet de cette façon d'insérer toutes les données, quelles que soient leur nature et leur origine. Car au-delà du stockage, l'un des enjeux des lacs de données est de pouvoir très facilement traiter et transformer les données en nouvelles pistes informations afin d'accélérer les cycles d'innovation, et ainsi être un support aux différentes initiatives autour des données [11].

Le lac de données a vocation à ingérer des flux de données bruts et à les rendre utilisables en les transformant pour satisfaire différents besoins d'exploration. Ceci est très similaire au mode d'alimentation des entrepôts de données.
Cette nouvelle approche est cependant différente en ce sens qu'elle permet de charger les données et de les transformer ensuite pour les rendre exploitables. On parle de **schéma "on read"** versus un **schéma "on write"**.

En effet le fait de savoir quelle information est attendue, induit un schéma extraction, transformation, chargement défini au préalable. On part de l'information attendue, pour trouver les données adéquates dans le système opérant, puis on les extrait, on les transforme en information, puis on les charge. On "écrit" donc ce schéma au départ de la conception du système décisionnel.
Avec les lac de données, on ne se pose pas de question puisque on ne sait pas à l'avance l'information que l'on va dériver des données. Ce n'est que lorsque l'utilisateur du lac de données a une idée et va utiliser les données, qu'il va appliquer son schéma d'extraction, de transformation et de chargement. Il le fait à la "lecture" des données.
Les initiatives autour des données sont très souvent limitées par les difficultés inhérentes aux phases de collecte et d'ingestion dans les systèmes. C'est souvent sur cette phase que nombre de projets peuvent échouer. Sur ce point, le fait de pouvoir charger les données dans un état quasiment brut, et d'itérer rapidement pour les utiliser est un avantage indéniable. On parle souvent d'ailleurs d'une démarche d'ELT (Extract-Load-Transform) plutôt que d'ETL (Extract-Transform-Load) à laquelle nous sommes habitués.
Là où un entrepôt de données pousse les données de leur origine vers leurs consommateurs selon un chemin relativement fixe où chaque magasin de données est sensé satisfaire un besoin, on a ici une bien plus grande flexibilité. C'est en effet à chaque utilisateur de matérialiser son besoin et d'extraire les différentes données sources puis de les combiner pour en faire du sens.

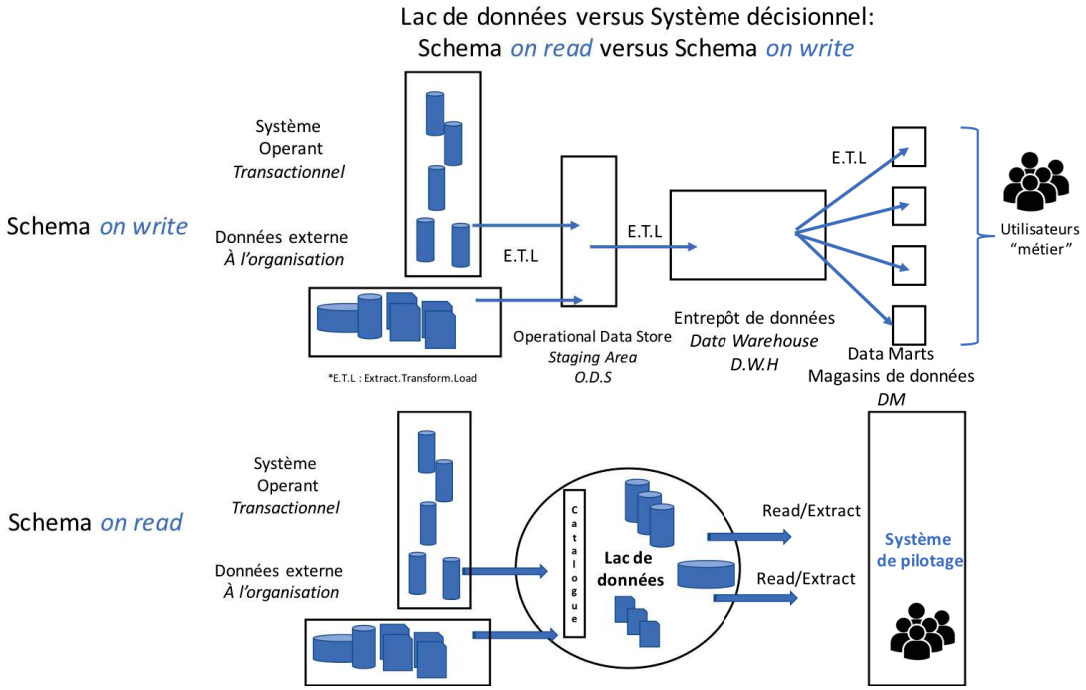


FIGURE *4.8: Lac de données versus Système décisionnel : Schema on read versus schema on write*

Ce **schéma "on read"** versus un **schéma "on write"** est une des différences majeures entre le lac de données et le système décisionnel. Nous l'illustrons au travers la figure 4.8.

4.6.2 Les données brutes

La structuration des données dans un entrepôt de données impose aux analystes d'utiliser les données au travers un formalisme rigide conçu au préalable. La transformation au chargement des données, si elle est structurante, est aussi destructrice des détails, du fait des agrégations nécessaires. L'approche de "Schema On Read" qui n'applique une structure aux données que lorsqu'elles sont utilisées permet de garder tout le potentiel des données d'origine intact. Cependant ces techniques de "Schema on read" nécessitent des compétences et des outils assez "pointus" techniquement et donc ne sont pas accessibles au plus grand nombre .

4.6.3 Le temps réel

Un autre facteur différenciant le lac de données vis-à-vis de l'entrepôt de données réside dans le côté opérationnel qui peut lui être associé. La capacité d'ingestion de flux en temps réel et de réaction aux données autorise des applications à interagir directement dessus. Alors que l'entrepôt de données synchronise la mise à jour de ces sources de données qui peuvent avoir des dépendances entre elles lors de traitements d'alimentation (ETL), le fait de ne plus avoir ce schéma d'écriture imposé rend le lac de données plus flexible et propose une mise à disposition en temps réel de la donnée, si cela est possible en s'affranchissant d'une quelconque dépendance avec une autre.

4.6.4 La véracité

Le fait de ne pas imposer de schéma strict aux données lors de leur ingestion présente un évident risque de qualité et de fiabilité. En effet le but principal des processus d'alimentation n'est pas que de transformer la donnée mais surtout de la nettoyer et la contrôler en vue d'apporter une totale confiance en l'information qui va être délivrée (la véracité de l'information). Dans les faits, on constate que les données restent finalement non structurées assez peu longtemps, puisqu'elles passent rapidement dans un sas qui va permettre de normaliser les sources et de les cataloguer pour obtenir des meta-données. La gouvernance apparaît alors comme l'un des enjeux majeurs du bon fonctionnement d'un lac de données, le catalogue de meta-données prend toute son importance, celui de jouer le rôle de garant de la traçabilité et de la véracité des données, notamment.

4.6.5 Les utilisateurs

Le lac de données est souvent basé sur des technologies qui permettent le traitement in-situ des données. Le fait de disposer, au même endroit, de puissance de calcul directement associée au stockage permet d'avoir un champ d'exploration des données très important et surtout très flexible car il n'a pas besoin d'être pensé en amont. La richesse des outillages intégrés permet à des analystes, des "data-scientists", ou des développeurs de tirer parti des données. On y associe aussi très souvent des processus d'analyse prédictive qui ont vocation à exploiter toutes les données pour établir des modèles prédictifs. La capacité de ceux-ci à être appliqués aux flux entrants apporte une dimension très pro-active à ce type de plate-forme vis-à-vis de la donnée tout en demandant aux utilisateurs de savoir manipuler ces outils plus pointus techniquement. Les utilisateurs d'un lac de données sont des utilisateurs très avertis, au fait des technologies informatiques et qui savent au moyen d'un catalogue de métadonnées

trouver les données qu'ils veulent explorer.

Ces observations nous permettent de lister les critères principaux qui différencient les lacs de données et les entrepôts de données. Dans le tableau 4.9, nous en présentons une première synthèse.

Nous avons donc positionné les lacs de données dans le système d'information et vis-à-vis du système décisionnel. En tant que composant du système d'information le lac de données peut donc être étudié selon son architecture via une démarche d'urbanisme. C'est ce que nous décrivons dans la section suivante.

	Lac de données	Entrepôt de données
Système de stockage	Hadoop, NoSQL, Base de données Relationnelle	Base de Données Relationnelle
Qualification de la donnée	Non	Oui
Valeur de la donnée	Haute	Haute
Granularité de la donnée	Brute	Agrégée
Connaissance de la donnée	non	oui
Préparation de la donnée	A la volée	Avant intégration
Intégration	Aucun traitement Donnée brute	Contrôle de la qualité, filtrage, agrégation
Transformation	Aucune puis ELT	ETL
Schéma	On read	On write
Architecture d'information	Horizontale	verticale
Modelisation	A la volée	Etoile ou flocon
Metadonnées	Oui	optionel
Modele de conception	Data driven	Information driven
Méthode d'analyse de données	unique	répétitive
Utilisateurs	Informaticien, data scientiste, analyste, développeurs	Ligne métier
Fréquence de mise à jour	Temps réel et mode batch	Mode batch
Gouvernance		
Architecture	Centralisée ou fédérée ou hybride	Centralisée

FIGURE *4.9: Comparaison lac de données entrepôt de données*

4.7 Démarche d'urbanisation appliquée aux lacs de données

Dans leur volonté de mettre en œuvre les lacs de données, les organisations veulent gagner en agilité et apporter de la flexibilité à la valorisation de leurs données. De ce fait appliquer une démarche d'urbanisation à la conception des lacs de données prend tout son sens.
L'urbanisation permet, selon Servigne [70] de partager une vision commune entre les utilisateurs et le service information mais aussi de :

- faire évoluer le SI pour qu'il soutienne et accompagne de manière efficace les missions de l'organisation et leurs transformations,
- réduire les dépenses,
- faciliter les évolutions,
- faciliter les changements de stratégies.

L'urbanisation est une discipline d'ingénierie informatique consistant à faire évoluer son système d'information pour qu'il soutienne et accompagne les missions de l'organisation et leurs transformations. Les avantages attendus de l'urbanisation sont de :

- rendre réactif par rapport aux projets métiers,
- aligner le SI avec les objectifs stratégiques,
- gérer la cohérence globale des données et des processus,
- Intégrer facilement des innovations technologiques,
- Capitaliser des données et des connaissances sur le système d'information,
- diminuer le coût de la maintenance et d'exploitation,
- augmentation de la qualité fonctionnelle,
- améliorer la capacité à améliorer le service rendu,
- fiabiliser des données,
- rendre le SI flexible.

Le terme "urbanisation" est utilisé par analogie avec les travaux d'architecture et d'urbanisme dans une ville en comparant une entreprise à une ville et à ses différents quartiers, zones et blocs.
Une telle démarche commence par le recensement et la capitalisation de l'ensemble des composants sur le système d'information de l'entreprise (bases de données, applications, services, etc.), en relation avec leur fonction, afin de les rationaliser et de permettre de valoriser le capital informationnel de l'entreprise. L'objectif d'une démarche d'urbanisation est donc d'aboutir à une structuration du système d'information permettant d'en améliorer ses performances et son évolutivité. Elle permet ainsi de donner les moyens à

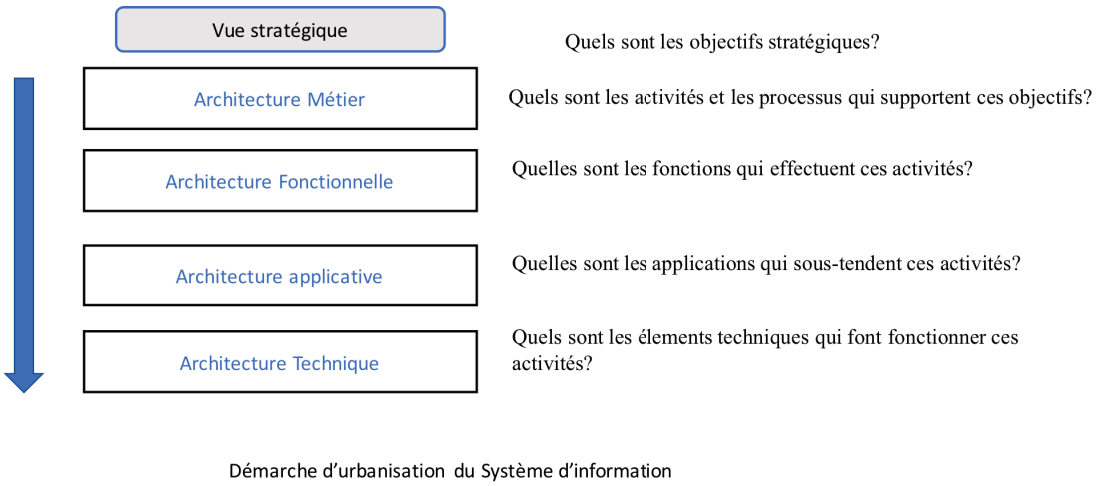


FIGURE *4.10: Démarche d'urbanisation du système d'information*

l'entreprise de faire évoluer son système d'information en connaissance de cause. La démarche d'urbanisation recentre le pilotage de l'évolution du système d'information sur la stratégie et les besoins des métiers de l'entreprise ou de l'organisation concernée. Elle est basée sur un modèle en quatre couches successives (voir figure 4.10) :

— l'architecture métier,

— l'architecture fonctionnelle,

— l'architecture applicative,

— l'architecture technique.

Ces différentes architectures s'opèrent de façon séquentielle, dans l'ordre préalablement cité, d'une manière descendante : on commence par l'architecture métier, puis on décline sur cette dernière l'architecture fonctionnelle, qui va servir de référence pour l'architecture applicative. L'architecture technique, quant à elle, vient supporter l'architecture applicative définie. Si la démarche d'urbanisation du SI s'opère de façon descendante, certains allers-retours sont nécessaires notamment entre l'architecture applicative et technique, même si des allers-retours avec les architectures métiers et fonctionnelles ne sont pas exclus. Certaines exigences dites non fonctionnelles décrivent les propriétés que le système doit avoir comme la haute disponibilité, la sécurité, l'évolutivité, etc. Elles peuvent nécessiter, de ce fait, des allers retours entre l'architecture technique et l'architecture applicative.
Par exemple, un besoin de protection des données qui nécessite une isolation ou un cryptage de la donnée, et donc un environnement technique spécifique, peut ne pas être compatible avec un choix de logiciel. L'architecture technique va alors influer sur l'architecture applicative et l'obliger à trouver une

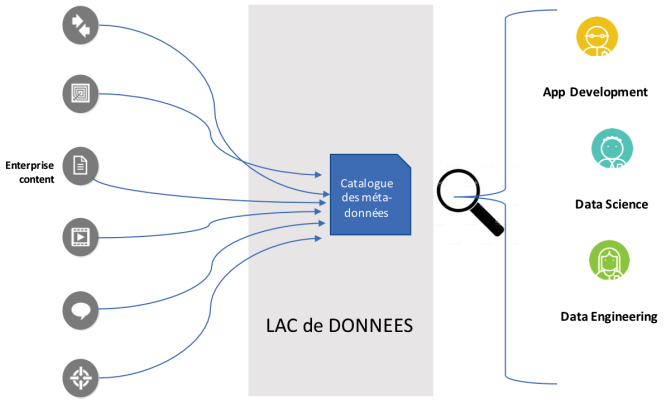


FIGURE *4.11: Architecture fonctionnelle d'un lac de données*

autre solution logicielle.
La démarche d'urbanisation du SI consiste dans un premier temps à étudier les différents secteurs fonctionnels d'une entreprise (production, administration, ventes, etc.), afin d'être en mesure d'en réaliser une cartographie, puis d'étudier de la même manière son système d'information.
Le lac de données étant un des composants du système d'information [49], sa conception obéit donc aux mêmes démarches d'architecture que les autres composants du SI notamment celle dite d'urbanisation des systèmes d'information [70].

Le paragraphe suivant détaille les quatre types architectures dans une démarche d'urbanisation.

4.7.1 L'architecture métier

L'architecture métier des lacs de données relève de la problématique d'une organisation autour de la capitalisation et valorisation de ses données pour supporter par exemple sa transformation numérique.

4.7.2 L'architecture fonctionnelle

Le lac de données a pour vocation de centraliser les données (en réponse à la capitalisation) sur un seul "réceptacle" d'entreprise (au sens logique) et de permettre aux outils logiciels de les explorer (en réponse à la valorisation). Le schéma fonctionnel d'un lac de données peut donc être représenté comme sur la figure 4.11.
Sa conception va prendre en compte les besoins fonctionnels auxquels le lac de données doit répondre :

— Accessibilité à toutes les sources de données

— Centralisation des sources de données

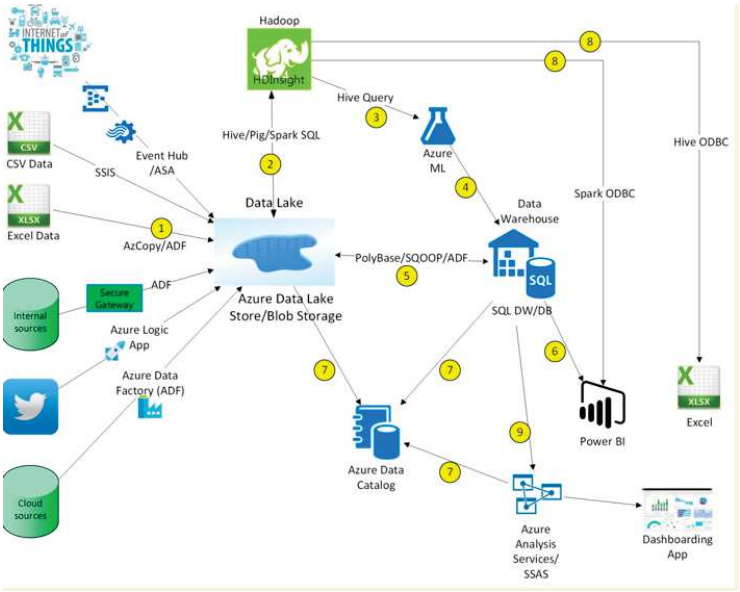


FIGURE *4.12: Architecture applicative d'un lac de données*

— Mise à disposition d'un catalogue des données disponibles

4.7.3 L'architecture applicative

L'architecture applicative est une vue informatique des lacs de données décrits dans la couche fonctionnelle. L'objectif est la distribution et la réutilisation des fonctions applicatives [69]. Les structures de données sont également décrites au niveau de l'architecture applicative. Les accès à ces données ainsi que la gestion de leur persistance (sauvegarde, sécurité) sont également décrits à ce niveau. La figure 8 donne un exemple d'architecture applicative de lac de données :
Les composants logiciels qui peuvent être déployés dans une architecture de lacs de données sont par exemple :

— les bases de données relationnelles, noSQL ou système de fichier de type HDFS, pour la partie stockage

— Les patrons d'architecture (framework) pour manipuler les données tels que Map-reduce, Apache Spark.

— Les logiciels de gestion de métadonnées tels que Informatica, IBM Metadata Catalogue

— Les suites intégrées pour les lacs de données, à base de HDFS tels que Cloudera, HortonWorks.

— Les logiciels de machine learning tels que Apache Spark, IBM Machine Learning...etc

Nous donnons un exemple de conception d'architecture applicative avec la figure 4.12[6]. Les couches logicielles y sont mentionnées, les flux de données établis, les sources de données référencées, les structures de données nommées et les interactions entre sources de données décrites.

Au sein de l'architecture applicative, Les couches logicielles sont décrites, les flux de données établis, les sources de données référencées, les structures de données nommées et les interactions entre sources de données décrites. Il y a donc un lien entre les données et les traitements qui est décrit dans l'architecture applicative. Ce lien entre données et traitements a retenu particulièrement notre attention car il soulève des questionnements autour des transferts de données, que nous allons approfondir dans le chapitre 5.1.

4.7.4 L'architecture technique

Il convient de rappeler que l'architecture technique décrit l'ensemble des composants matériels supportant l'architecture applicative. Ces composants peuvent être :

— des serveurs,

— des postes de travail,

— des équipements de stockage (baie de stockage, SAN, filers...),

— des équipements de sauvegarde,

— des équipements réseaux ("routeurs", "firewalls", "switches", "load-balancers", "accélérateurs SSL").

Le choix de ces composants techniques va être orienté par des exigences, dites non fonctionnelles des lacs de données telles que :

— la sécurité,

— la disponibilité,

— le passage à l'échelle (scalability),

— l'auditabilité,

— la performance,

— le volume,

— l'intégrité,

6. why use a data-ake. Retrieved from http ://www.jamesserra.com/archive/2015/12/why-use-a-data-lake/ : http ://www.jamesserra.com/archive/2015/12/why-use-a-data-lake/

— la robustesse,

— la maintenabilité,

— la fiabilité.

Selon l'importance de ces exigences (ou contraintes) non fonctionnelles le choix de l'architecture applicative peut être remis en cause par leur application dans l'architecture technique. Par exemple, dans le cas où le logiciel choisi ne permet pas la montée en capacité attendue ou bien ne tire pas parti des performances de l'architecture technique. Les explorations dans les lacs de données peuvent être très gourmandes en puissance de calcul et nécessiter l'utilisation des traitements massivement parallèles (architecture de type MPP (massively parallel processing), une architecture technique qui permet l'exploitation de cette technologie mais dont le logiciel de manipulation ne sait pas en tirer parti remet en cause sa sélection dans l'architecture applicative. Dans cet exemple, on se trouve face à des contraintes (non liées au métier de l'organisation) qui réduisent le champ des solutions applicatives possibles.

L'influence de ces exigences (ou contraintes non fonctionnelles) reste peu étudiée, que ce soit au niveau de la littérature scientifique ou dans le monde industriel. Cependant la maturité des projets sur les lacs de données reste encore faible, il est à prévoir l'importance croissante de l'étude d'impact de ces contraintes dans la définition des architectures des lacs de données.
Dans le chapitre 5.1, nous explorons l'influence de certaines contraintes techniques sur la remise en cause d'une architecture applicative.

Dans cette section nous avons proposé d'appliquer une démarche d'urbanisation pour la conception de l'architecture des lacs de données. Cette démarche comprend quatre phases de définition d'architecture : métier, fonctionnelle, applicative, technique. Au vu des facteurs pouvant influencer chacune d'entre elles, notamment le passage entre l'architecture applicative et technique, cette démarche peut être un moyen de limiter les possibilités de transformation en marécage des lacs de données ou leur sous-utilisation. Nous pensons que la mise en œuvre de cette démarche d'urbanisation pour les lacs de données peut accélérer la réussite de leur mise en œuvre.

Après avoir appliqué une démarche d'urbanisation à l'architecture des lacs de données, nous détaillons les principaux composants ou fonctions que nous avons identifié comme clés dans leur conception, dans la section suivante.

4.8 Les fonctionnalités des lacs de données

Nous décrivons ainsi notre proposition de fonctionnalités à inclure dans la conception d'un lac de données :

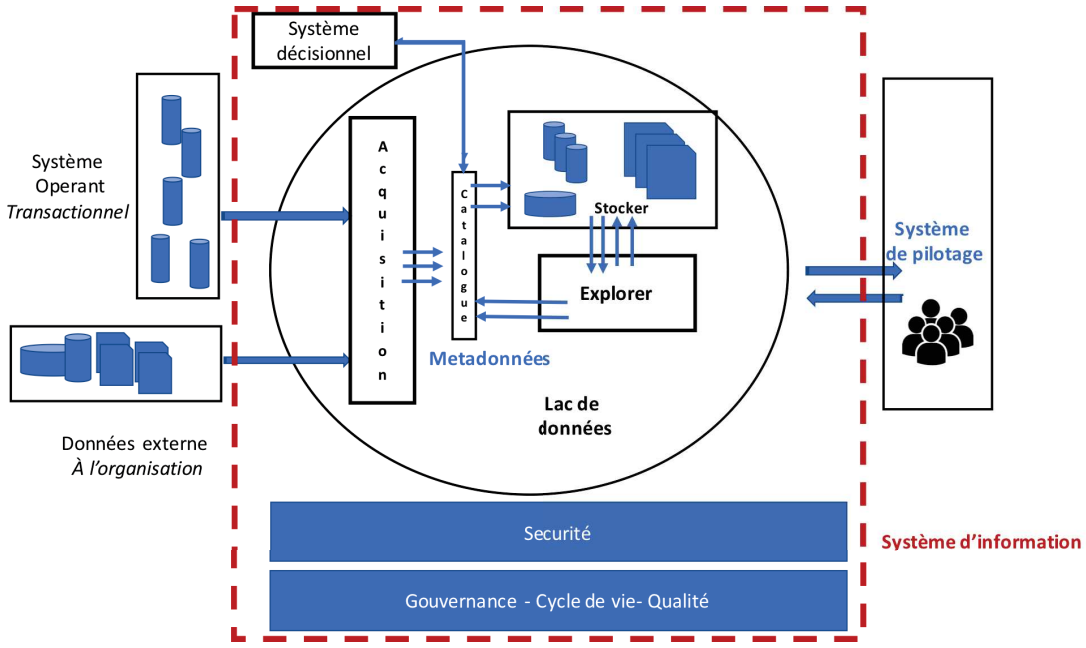


FIGURE *4.13: Macro composant ou fonctions d'un lac de données*

— L'aquisition,

— Le catalogage,

— Le stockage,

— L'exploitation,

— La gouvernance,

4.8.1 L'acquisition

Cette fonctionnalité est celle par laquelle les sources de données, internes, externes, structurées, non structurées vont être "ingérées" par le lac de données. On peut faire l'analogie avec le système décisionnel (d'entreprise), et son composant ODS (Operation Data Store). Tout comme la description que nous avons faite dans la section sur les architectures de référence (section 2.4.6) la fonctionnalité "acquisition" du lac de données peut intégrer tous types de flux (batch, streaming, etc). On peut ainsi créer des sous-fonctionnalités par type de flux, comme pour le système décisionnel.
On peut de même envisager une mutualisation de la fonctionnalité d'acquisition avec celle du système décisionnel. En effet pour certaines organisation c'est d'ailleurs le premier embryon de création du lac de données, où le lac de données avec cette fonctionnalité plus celle du catalogage (métadonnées) se place

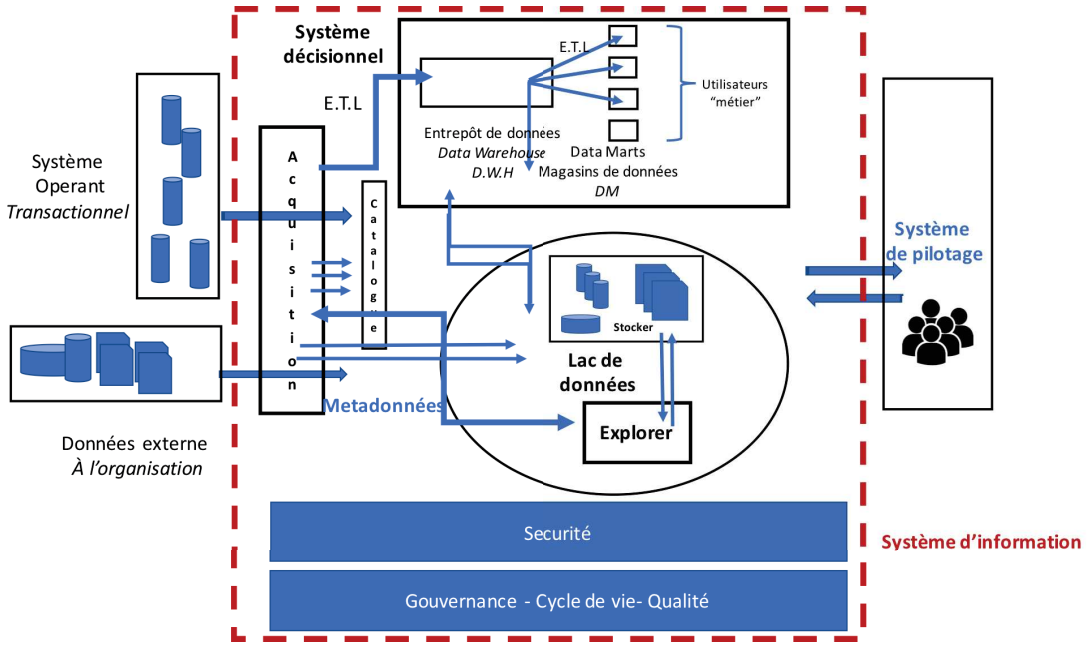


FIGURE *4.14: Mutualisation du composant acquisition entre le système décisionnel et le lac de données*

en amont de l'entrepôt de données, comme une extension de l'ODS. Nous schématisons ce cas de figure, dans la figure 4.14.

4.8.2 Le catalogage

Tous les travaux sur les lacs de données [17][26][5][55][72][10] mettent en exergue l'importance de cette fonctionnalité : la constitution d'un catalogue de méta-données. Sans cette fonctionnalité le lac de données n'est pas gouverné et donc risque de se transformer en "marécage".
Il n'y a rien de nouveau à mettre en lumière la gestion des métadonnées pour des données qui doivent servir à des fins d'analyse, c'est le cas aussi dans les systèmes décisionnels. Cependant un entrepôt de données peut fonctionner sans cela, ce n'est pas le cas du lac de données. En effet c'est au travers de ce catalogue que les utilisateurs du lac de données vont pouvoir savoir ce qu'il contient mais aussi s'assurer, notamment, de la traçabilité, la qualité, la sécurité, la propriété et la responsabilité de la donnée unitaire. Sans cela l'utilisateur est "aveugle" dans son exploration et donc potentiellement inefficace.

La gestion des métadonnées peut être définie comme l'administration des métadonnées. Concrètement, une bonne gestion des métadonnées passe par la mise en place de règles et de processus permettant d'assurer la possibilité d'accéder, de partager, de lier, d'intégrer, de maintenir et d'analyser

les métadonnées dans toute l'entreprise.

Les métadonnées sont les pièces maîtresses des lacs de données pour garantir leur gouvernance, en termes de sécurité, gestion de leur cycle de vie et qualité. Il est très difficile à l'heure actuelle d'obtenir un consensus dans la littérature scientifique autour des formats que doivent avoir les métadonnées et le nombre de leurs catégories, les avis divergeant aussi dans le monde industriel.

Outre les informations de base, les métadonnées doivent permettre à l'entreprise de disposer d'une cartographie complète pour comprendre les données et les modalités de traitement associées. Les questions que l'on peut se poser et où la réponse doit se trouver dans les métadonnées sont, par exemple :

— Qui a créé cette donnée ? Qui l'utilise ? À qui appartient-elle ? Qui en assure le traitement et la maintenance ?
— Quelle en est la définition métier ? Quelles sont les règles métier ? Quel est son degré de sécurité ? Quelles en sont les dénominations standards au sein des bases de données ?
— Où est stockée la donnée ? D'où vient-elle ? Où est-elle utilisée, partagée ? À quelle norme réglementaire ou juridique répond-elle ?
— Pourquoi stocke-t-on cette donnée ? Quel est son usage et sa finalité ? Quel est le levier métier pour l'utiliser ?
— Quand cette donnée a-t-elle été créée, actualisée ? Quand doit-elle être effacée ?
— Comment cette donnée est-elle formatée ?
— Dans combien de base de données ou sources est-elle présente ?

Les réponses à ces exemples de questions apportent une plus-value considérable aux données. Elles deviennent précises, compréhensibles par tous les métiers de l'organisation, accessibles et faciles à partager.

Les attentes autour de la métadonnée sont donc très fortes et la mise en place de la gestion des métadonnées se doit d'atteindre les objectifs suivants :

— L'**accès à l'information** par des utilisateurs sans culture technique. La donnée est créatrice de valeur quand elle est exploitée par le métier à des fins business. Avec une mise en contexte détaillée, il est aisé pour chacun d'accéder à l'information demandée en utilisant un simple moteur de recherche indexant la structure, le contenu, la qualité et la nature de chaque donnée.
— La **qualité des données**. L'évaluation de la qualité de la donnée est plus facile. Une fois qualifiées, les données n'ont plus besoin d'être passées au tamis pour savoir si elles sont utiles, d'actualité et pertinentes. C'est le rôle des métadonnées

— **Gain de temps**. En donnant un profil complet et détaillé à chaque donnée, l'utilisateur consacre son temps à l'exploitation, et non pas à l'évaluation de l'information.

— La métadonnée permet **la protection des données sensibles**. Le règlement RGPD impose un nouveau cadre pour l'utilisation des données personnelles. La cartographie des données personnelles et des données sensibles, facilite le travail de protection (chiffrement, gestion des accès...) de ces données et la mise à jour du registre des traitements, demandé par le régulateur.

— Une **exploitation et une collaboration** facilitées. La **traçabilité** procurée par les métadonnées apporte une transparence sur les traitements réalisés. Cette connaissance confère un surcroît de confiance à l'utilisateur au moment d'exploiter ces données en étant par exemple assuré de la maîtrise des impacts de ces actions sur les données.

— **La mise à jour de données cachées.** L'entreprise dispose souvent d'un réservoir de données cachées, donc gâchées. Elles peuvent être issues des systèmes opérants, d'applications complexes ou autre, ce qui rend leur analyse et leur exploitation quasi impossible. Une fois cartographiés avec les métadonnées, ces jeux d'informations retrouvent la lumière et peuvent être exploités par les métiers dans l'entreprise.

Les littératures scientifiques françaises ou anglo-saxonnes sont très abondantes sur les métadonnées. Dans un rapport l'INRIA, datant de 1999 [65], une catégorisation des métadonnées, par types de données et par types de traitements est suggérée. De même après analyse de certains travaux sur le sujet, les métadonnées ne sont jamais étudiées de façon générique, mais dans un contexte, lié à une problématique. Par exemple [55] étudient la gestion des métadonnées dans le cadre des lacs de données, et envisage une modélisation en data vault pour en faciliter leur gestion. Le contexte d'étude est le lac de données. De même pour [5], qui étudie le profilage sémantique pour la gestion des métadonnées ou [26] qui se positionne aussi dans le contexte des lacs de données et propose d'enrichir les métadonnées du lac avec des informations sémantique.

Si l'angle de vue de nos travaux n'est pas la gestion et constitution des métadonnées dans le lac de données, l'aspect gouvernance doit cependant être abordé.

Dans le cadre de nos travaux, nous avons volontairement restreint le champ à trois catégories de métadonnées, qui nous semblent essentielles pour permettre la gouvernance des lacs de données. Nous avons retenu les catégories de métadonnées suivantes :

— Techniques

— Opérationnelles

— Métiers

Les métadonnées techniques fournissent des détails sur les systèmes sources et cibles, leurs structures de zone et de table, les attributs, les dérivations et les dépendances. Les métadonnées techniques incluent

également des détails sur le profilage, la qualité, les processus d'intégration, les projets et les utilisateurs. C'est le comment et le pourquoi. Avec les métadonnées techniques nous pouvons répondre aux questions :

— d'où vient la donnée ?

— Comment a-t-elle été collectée ?

— Par quel moyen technique ?

— Quels traitements ont été effectués ? Par qui ? Pourquoi ? Quand ?

— Quelles sont les raisons qui ont motivé sa création, sa collecte, sa réception ?

— Quelle confiance pouvons-nous avoir dans cette source ?

Les métadonnées opérationnelles concernent les informations sur les applications qui utilisent les données, leur fréquence de mise à jour, le nombre d'enregistrements, les flux de données, les statistiques d'utilisation, d'accès et autre.

Les métadonnées métiers fournissent un contexte métier pour l'actif informationnel et ajoutent une signification métier aux artefacts créés et gérés par d'autres applications informatiques. Les métadonnées métier comprennent les dictionnaires contrôlés, les taxinomies, la gérance, les exemples et les définitions métier. On va retrouver des informations telles que :

— La signification et les descriptions métier des données,

— La hiérarchie qui regroupe les données en fonction des besoins métier,

— La liste des termes approuvés,

— La gérance des données et des processus,

Dans les lacs de données la centralisation de la collecte et gestion au sein d'un référentiel unique (un catalogue) des métadonnées va être le garant de la gouvernance et éviter au lac de se transformer en «marécage». Ce qui va permettre aux utilisateurs du lac de données d'avoir une vision globale des données disponibles, de leur accès et leur valeur.

L'avantage de fournir un catalogue des métadonnées va permettre par exemple de rapidement évaluer l'impact de toute transformation que ce soit au niveau des données ou des traitements. Par l'affichage des dépendances entre les objets, l'analyse d'impact permet de gérer les effets des modifications apportées aux données. Ce type d'analyse s'étend sur plusieurs outils et aide à évaluer le coût des modifications. Par exemple, un développeur peut anticiper les effets d'une modification apportée à une définition de table ou à une logique métier.

La possibilité au travers du catalogue des métadonnées de procéder à des recherches intégrées pour faciliter la localisation et l'extraction d'objets du référentiel, au moyen de fonctions de recherche rapides ou de recherche avancée, en fonction d'un nom ou d'une description, entrés intégralement ou partiellement tels que :

— Type,

— Données de création,

— Dernier modifié,

— Cas d'utilisation,

— Dépend de.

De notre point de vue, au-delà de la gestion des métadonnées, c'est la constitution du catalogue de métadonnées, et sa gestion, qui sont obligatoires, dans toute conception de lacs de données. Ce catalogue devient un élément pré requis dans la création de ce nouveau composant du SI que sont les lacs de données.

4.8.3 Le stockage

Ce composant est celui où les données acquises et cataloguées vont pouvoir être stockées avant, pendant et après (potentiellement) la phase d'exploitation (ou d'exploration) de ces données. C'est à ce composant qu'est souvent réduit un lac de données [22][17] associé à la technologie Apache Hadoop. Russom [66] emploie d'ailleurs le mot "Hadoop Data lake".
Dans nos travaux précédents (section 4.5 nous avons montré que le lac de données ne doit pas être réduit à cette fonction (ou composant) mais bien être considéré comme un concept d'architecture.
Dans ce composant, nous avons aussi montré que d'autres technologies alternatives ou complémentaires à Apache Hadoop peuvent être envisagées. Russom [66] d'ailleurs cite la complémentarité des bases de données relationnelles, il emploie le mot "relationnal data lake", mais aussi les bases de données de type NoSQL, que nous avons décrites dans le chapitre 3. Le stockage est donc potentiellement pluriel et hybride, et ce composant central et complexe dans la conception des lacs de données.
L'architecture technique et applicative vont fortement influer sur ce composant, comme nous le montrons dans le chapitre 5.1.

4.8.4 L'exploitation ou l'exploration

Ce composant est l'objectif des lacs de données : permettre l'exploitation et l'exploration des données de l'organisation. Il est de notre point de vue à destination seulement de quelques utilisateurs (par

comparaison au système décisionnel) et son outillage technologique n'est pas accessible à des utilisateurs non avertis.
C'est un point de divergence avec la position de certains travaux [10][17] qui ouvrent le lac de données à tous les utilisateurs. Cela suppose pourtant une information déjà clé en main, ou des données préparées, prêtes à être exploitées, si les utilisateurs ne sont pas familiers de certaines techniques. Par exemple, on ne peut envisager de laisser un utilisateur métier, ni mathématicien ni statisticien faire de l'analyse prédictive au travers d'algorithme de type apprentissage machine ("machine learning"). Ce qui est envisageable c'est de trouver les bons ensembles de données, sur lesquels appliquer les bons algorithmes, et alors de mettre à disposition le résultat via le système décisionnel.
De notre point de vue, ce composant n'est pas ouvert aux utilisateurs non avertis. Dans certaines études, et certaines positions du monde industriel, les lacs de données sont à destination des spécialistes de la science des données : les "data scientists'. Même si nous nous ne satisfaisons pas de cette appellation, elle peut être un consensus accordé pour désigner les utilisateurs des lacs de données. Ceci afin de les différencier des utilisateurs métiers.

4.8.5 La gouvernance

La gouvernance du lac de données a en son cœur la gestion du catalogue des métadonnées (section 4.8.2) mais ce n'est pas le seul élément nécessaire pour gouverner un lac de données. La gestion de la sécurité, du cycle de vie et de sa qualité en sont les fondamentaux.

4.8.5.1 La sécurité

La sécurité revêt deux aspects principaux : l'aspect protection des données, notamment les données sensibles, et l'aspect confidentialité, les données personnelles par exemple. Ces deux aspects sont mis en exergue au travers de l'actualité : l'une par les vols de données qui ne cessent d'être rendus publics (exemple avec la société Uber Technologie, qui en 2016 s'est fait voler des données et qui va devoir payer 148 millions de dollars aux états américains[7]) l'autre par la réglementation Européenne sur les données personnelles (RGPD).
Le lac de données est donc très exposé et doit prendre en compte cet aspect, dès sa conception (ce qui est rendu obligatoire dans la RGPD d'ailleurs). Cela se fait au travers des règles, des processus et un outillage technique adéquat. Cela peut être des outils de cryptage ou d'anonymisation de données par exemple, des règles de conservation de données, des processus de suppression et d'archivage.
Tout cela doit être abordé et traité lors de la conception d'architecture.

7. https ://www.capital.fr/entreprises-marches/usa-uber-paie-148-millions-de-dollars-pour-regler-la-violation-des-donnees-de-2016-1308667

4.8.5.2 Le cycle de vie

Le cycle de vie des données concerne l'accompagnement des données tout au long de leur durée de vie sur la base de règles qui peuvent être automatisées. Si une donnée entre dans une base de données, soit elle sera utilisée pour différentes raisons, soit elle sera stockée sans utilité, jusqu'à ce qu'elle devienne obsolète. Dans un cas comme dans l'autre, les données peuvent se voir appliquer à tout moment des opérations et des validations. Cependant, tôt ou tard, chacune arrivera au terme de sa durée de vie utile et sera archivée et/ou purgée. C'est ce concept de définition et d'organisation des données que l'on appelle cycle de vie des données ou sous le terme anglo-saxon : Data Life Cycle Management.
Le cycle de vie des données peut être vu comme un catalyseur qui fait passer une donnée d'une phase à l'autre, depuis sa création jusqu'à sa suppression.

Dans un lac de données, où le volume de données acquises mais aussi générées (via les explorations et exploitations diverses) peut être très important, prendre en compte ce cycle de vie des données, dès que l'acquisition des données se fait doit être un impératif sous peine d'être vite submergé par des données "inutiles".
Pour cela il faut identifier et caractériser les données. Il s'agit de les indexer en spécifiant notamment leur type, leur date de création, ainsi que l'utilisateur (ou le service) qui en est à l'origine. Éventuellement, on peut leur associer des mots clés. Puis il faut créer des règles destinées à préciser l'évolution de leur valeur pour définir les choix de support adaptés. Ainsi, si une donnée doit être accessible rapidement, elle sera stockée sur un environnement de haute performance. Si sa valeur est moins critique, on pourra se contenter de la migrer sur un environnement moins rapide (et moins coûteux), voire sur un support de type bande. Les règles doivent également déterminer la fréquence des sauvegardes et la vitesse de restauration. Dans ce cas là encore, il s'agit de déterminer le support adapté.
Tout au long de son cycle de vie, chaque donnée pourra ainsi être déplacée ou dupliquée de nombreuses fois sur différents supports. Logiquement, celles qui auront perdu toute valeur seront automatiquement détruites. Autre contrainte : il faut potentiellement être capable d'accéder à un ensemble de données associées entre elles. Il s'agit donc de les lier logiquement aussi, ce qui peut complexifier la gestion de ce cycle de vie.
La gestion du cycle de vie des données va influencer les choix de l'architecture technique du lac de données, notamment au niveau du stockage et des différents supports et technologie disponibles, selon son statut dans son cycle de vie. Comme nous l'avons vu dans le chapitre 3, l'architecture peut tirer parti des technologies de type flash par exemple, si les données doivent être accessibles rapidement ou bien d'une technologie moins coûteuse lorsqu'elles seront archivées..

Le cycle de vie des données dans le lac de données s'appuie donc fortement sur une bonne gestion

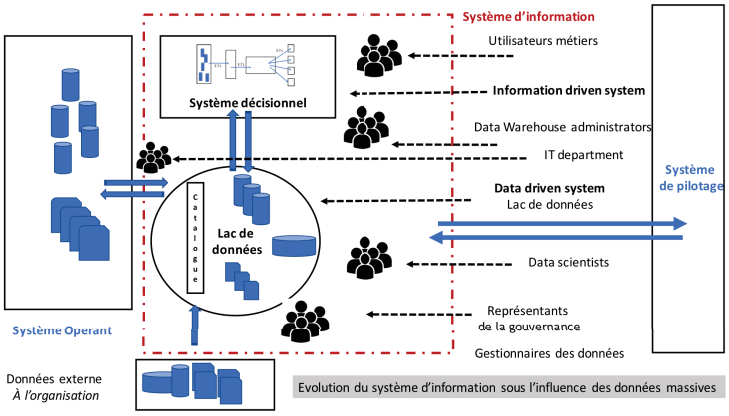


FIGURE *4.15: Interaction des lac de données dans les systèmes d'une organisation*

des métadonnées, qui doivent inclure les informations et règles nécessaires à leur suivi et mise en oeuvre. Lors de la conception d'architecture, ces informations, règles et processus doivent être définis et appliqués pour chaque donnée qui va être acquise.

4.8.5.3 La qualité

La qualité des données est considérée comme un défi dans les lacs de données où par nature tous types de données sont acceptés que ce soit des données primaires (mais qui peuvent avoir subi une transformation et nettoyage préalable obligatoire, comme des images provenant de satellite), mais aussi des données plus "brutes" qui ne sont ni "nettoyées" ni contrôlées, mais qui sont intégrées telles qu'elles sont émises. Là encore la gestion des métadonnées va jouer le rôle d'informateur pour l'utilisateur, qui grâce à la traçabilité des données peut connaître l'état de la qualité des données qu'il va utiliser. Selon le cas, il appliquera alors le même principe que dans un système décisionnel et "nettoiera" ces données au moment où il les utilise. Cette tâche de "préparation" est donc toujours existante pour garantir une qualité des données et une fiabilité de l'information qui va en dériver. Le lac de données ne s'en affranchit pas, il déporte juste le moment où cela sera fait, c'est-à-dire au moment d'utiliser les données.

4.9 Notre point de vue sur les lac de données

Comme nous l'avons énoncé au début de ce chapitre, nous positionnons le lac de données comme un nouveau composant du système d'information, qui dispose d'une architecture métier, fonctionnelle, applicative et technique, comme n'importe quel composant d'un SI [70]. Il est constitué d'un ensemble

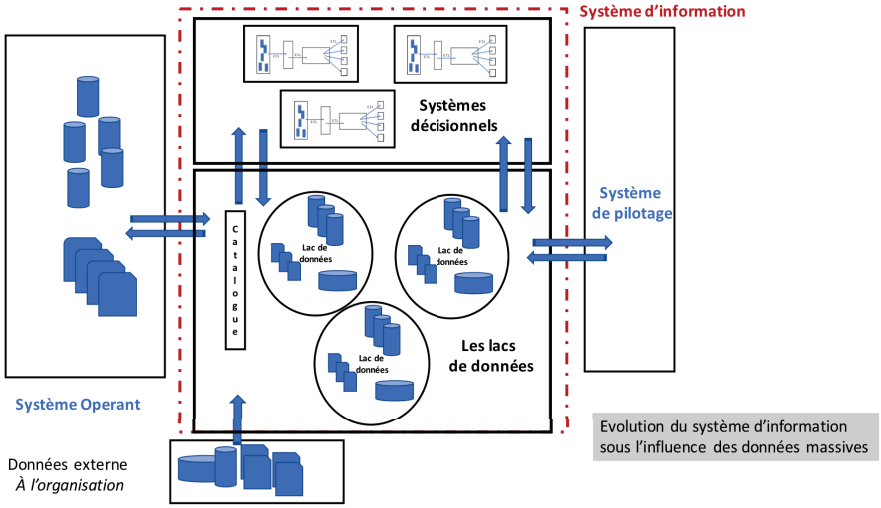


FIGURE *4.16: Vision d'une architecture globale du système d'information, sous l'influence des données massives*

de données dans leur format d'origine, accessible par des utilisateurs "avertis" et doté de technologies innovantes permettant de manipuler ces données en vue d'identifier éventuellement des informations pertinentes. Il est par définition agile et flexible.
Pour ne pas tomber dans le piège du "marécage" de données ni dans la création d'un environnement de type infocentre non contrôlé, le lac de données, tout comme le système d'information doit être gouverné. Pour cela, toutes les recommandations de gouvernance des données doivent s'appliquer :

— gestion de la qualité des données

— gestion de la sécurité des données

— gestion du cycle de vie des données

— gestion des métadonnées

Un composant vital du lac doit être constitué, il s'agit d'un catalogue des sources de données présentes et disponibles dans le lac de données et plus précisément d'un catalogue renseignant les méta-données des sources de données présentes ou accessibles. La présence de ce catalogue de méta-données, pièce essentielle de cette architecture est le garant de la gouvernance (sécurité, cycle de vie et qualité) et de la cohérence des sources de données présentes.

De notre point de vue, les projets de lacs de donnée sont donc plus à associer à un projet de gouvernance des données que d'un projet de type décisionnel. La complémentarité des deux systèmes au

sein du système d'information prend tout son sens.

Nous avons volontairement mis au singulier le système décisionnel et le lac de données dans nos différentes figures, mais comme nous le verrons dans le chapitre suivant 5.1, dans certaines grandes organisations industrielles, il peut y avoir plusieurs systèmes décisionnels mais aussi plusieurs lacs de données, comme nous le montrons dans la figure 4.16 . Pour plus de lisibilité dans nos schémas nous avons opté pour une vision plus généraliste, car la pluralité dans le système d'information des systèmes décisionnels et lacs de données n'est pas celle qui est encore la plus répandue dans le monde industriel, même si elle tend à s'étendre.

4.10 Synthèse du chapitre 4

Dans ce chapitre, nous avons introduit le concept des lacs de données, fait un état des lieux des travaux de recherche mais aussi industriels sur le sujet, donné notre définition des lacs de données et précisé les enjeux qu'ils représentaient.
Nous avons établi notre point de vue, basé sur des analyses fines et poussées des divers documents et travaux sur le sujet, mais aussi étayé à partir de constatations faites avec des collaborations industrielles, sur ce concept.
Nous avons proposé de considérer les lacs de données comme un nouveau composant du système d'information, qui comprend déjà le système décisionnel. Nous avons par ailleurs positionné les lacs de données vis-à-vis des systèmes décisionnels pour justifier ce positionnement.
Partant de cette hypothèse, nous avons appliqué aux lacs de données, une démarche d'urbanisation pour démontrer que les lacs de données sont bien un concept d'architecture, et non une implémentation spécifique (Apache Hadoop).
Nous avons établi quelles sont, selon notre point de vue, ses fonctionnalités, en mettant l'accent sur la constitution et gestion du catalogue des méta-données et sur l'aspect gouvernance des données.

Après avoir énoncé que le lac des données **EST** un concept d'architecture, notre sujet de questionnement se pose sur les éléments qui peuvent influencer sa conception et peut-être remettre en cause certains postulats d'architecture.
Notre intérêt s'est porté sur l'étude de la relation donnée-traitement, et un facteur potentiellement influant, appelé, gravité des données. Après avoir défini ce que nous entendons par gravité des données, nous étudions dans le chapitre suivant, au travers d'un cas d'usage industriel, l'influence de ce facteur sur l'architecture des lacs de données.

Chapitre 5

Influence de la gravité des données dans l'architecture des lacs de données

Dans le contexte industriel, les architectes d'information ont en charge la gouvernance, la conception et l'outillage technologique de ces lacs de données. L'architecture d'information qu'ils doivent mettre en place doit prendre en compte conjointement des contraintes fonctionnelles mais aussi non fonctionnelles. De par l'important volume de données à traiter, la diversité des formats de ces données mais aussi leur coût considéré comme peu élevé, la technologie de type Apache Hadoop s'est imposée comme référente, quasi unique, occultant les discussions sur l'impact des contraintes non fonctionnelles sur ce choix. Cette technologie comme solution unique pour les lacs de données est désormais remise en cause [66][49] par le monde industriel et des architectures hybrides, avec introduction de technologies complémentaires à Apache Hadoop, sont désormais envisagées.

Notre intérêt académique se porte sur les facteurs pouvant influencer cette hybridation technologique mais aussi applicative dans les lacs de données, sous l'influence des données massives.

La relation donnée-traitement à l'intérieur du lac de données a retenu notre attention car aucune contrainte non fonctionnelle liée à cette relation ne semble affecter l'architecture des lacs de données et aucune littérature scientifique ou industrielle n'y porte une grande attention, encore.

En vue de mieux appréhender cette relation donnée-traitement, nous nous sommes intéressés aux travaux l'étudiant dans un contexte non lié aux lacs de données. Nous avons découvert les travaux de MacCrory [46] qui introduisent la notion de gravité des données et étudient sous cet angle la relation donnée-traitement.

Nous émettons l'hypothèse que cette notion de gravité des données [45] peut être considérée comme une contrainte non fonctionnelle si elle est étudiée dans un cadre de conception d'architecture.

Partant de cette idée, un des objectifs de nos recherches en cours consiste à compléter la définition de [45] afin d'étudier son impact sur les architectures des lacs de données et d'en évaluer son intérêt. Nous vérifions nos hypothèses au travers la réalisation d'une étude d'architecture d'un cas réel de lac de

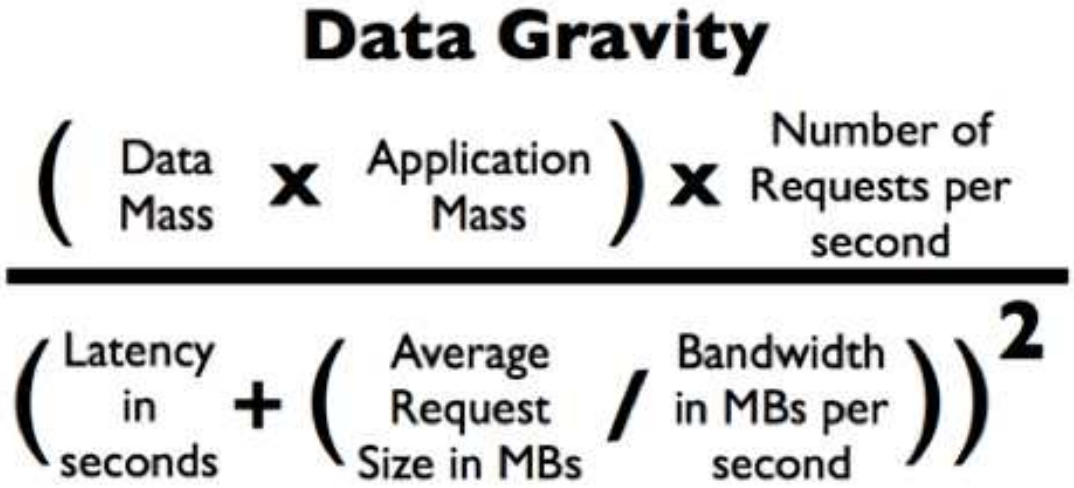


FIGURE *5.1: Formule de MacCrory sur la gravité des données*

données en milieu industriel.

5.1 La gravité des données

Les travaux de McCrory [45] s'attachent à établir une analogie entre la gravité des données et la gravité au sens physique. Pour cela il étudie la relation donnée-traitement, et, par un parallèle de raisonnement avec l'attraction physique, définit une force d'attraction entre les données et les traitements. La translation de la formule physique, dans le cadre des données est indiquée dans la figure 5.1.

La loi de la gravité stipule que l'attraction entre les objets est directement proportionnelle à leur masse. Pour établir son analogie, il émet l'hypothèse que la masse des données est liée à leur volumétrie. Il "invente" alors le terme de **gravité des données** pour décrire le phénomène dans lequel le nombre (ou la quantité) et la vitesse à laquelle les services, les applications, et même les clients sont attirés par les données, augmentent à mesure que la masse des données augmente.

À mesure que les données s'accumulent (leur volume augmentant on construit une masse), il est supposé que des services et des applications supplémentaires peuvent être attirés par ces données.

Le phénomène de gravitation peut alors être appliqué : les données voient leur gravité augmenter et devenir de plus en plus importantes, elles vont alors attirer les traitements à elles.

La relation donnée-traitement peut alors être étudiée sous un autre angle, celui de la gravité des données

en définissant quels sont les facteurs à y inclure. Pour MacCrory[45] c'est le facteur "volume" qui est le plus influant dans la gravité des données. Il ouvre cependant la porte à d'autres paramètres pouvant influencer cette gravité tels que la sensibilité, le trafic du réseau, le coût, etc.
Dans un premier temps c'est le facteur "volume" qui a retenu notre attention car nous étudions l'impact des données massives sur le système d'information et en particulier les lacs de données.
En étudiant les travaux de Walker-Alrehamy [3], qui s'appuient eux aussi sur ceux de (McCrory, 2010), un autre facteur semble lui aussi influencer la relation donnée-traitement, le facteur "sensibilité". Le cadre de leurs travaux est un lac de données fonctionnellement dédié aux données personnelles.
L'évaluation de la gravité des données, au travers de la masse des données (peu importante dans ce cas d'étude) n'est pas le paramètre le plus influent, mais celui de la sensibilité des données. Ce facteur va « peser » plus dans l'évaluation de la gravité des données, que celui du volume. Leur lac de données dédié aux données personnelles va avoir une sensibilité si forte, qu'il va attirer à lui les traitements devant manipuler les données qu'il contient. Les données ne seront donc pas déplacées vers les traitements. Pour Walker-Alrehamy [3] la sensibilité est un des facteurs rentrant en compte dans la gravité des données. Ces différents travaux [3] [46] montrent que le volume et la sensibilité des données, que l'on inclut dans la gravité des données, peuvent influencer la relation donnée-traitement.

Les architectures des lacs de données existantes sont basées sur l'acquisition de données, sur lesquelles les traitements d'exploration et d'analyse (par exemple) vont s'appliquer. Il n'est pas envisagé que les traitements des lacs de données se déplacent vers les données produites. Notre hypothèse est que si on prend en compte la gravité des données, le traitement des données du lac peut, selon le cas, potentiellement se déplacer où résident les données et non pas le contraire.
En architecture d'information, les paramètres qui composent cette gravité définie par [46], tels que le volume et la sensibilité, sont considérés comme des éléments de contraintes non fonctionnelles. Cela nous amène à considérer la gravité des données comme étant une contrainte non fonctionnelle, à évaluer lors de la conception des lacs de données.

C'est là, notre première proposition de ce chapitre : considérer la gravité des données comme une contrainte non fonctionnelle.

5.2 La gravité des données dans les lacs de données

Par tradition les systèmes opérationnels (ou transactionnels) qui sont une des sources de données principales pour l'alimentation des systèmes d'information (voir figure 4.7), sont séparés au sens physique-

technique, ils sont par tradition hébergés sur des plate-formes informatiques ou serveurs physiques différents. En effet depuis des décennies les systèmes opérationnels exigent des qualités techniques différentes de celles des systèmes d'information. Ce sont principalement les aspects non fonctionnels de chacun qui sont traités de façon distincte.
Dans le cadre des systèmes opérationnels-transactionnels, les contraintes non fonctionnelles essentielles à traiter sont les suivantes :

— Haute disponibilité ;

— Fiabilité ;

— Temps réel ;

— Sécurité- sensibilité ;

— Volumétrie ;

— Sauvegarde ;

— Évolutive-montée en charge ;

— Performance ;

— Cryptage.

Les systèmes opérationnels nécessitent de focaliser leur puissance de calcul sur les opérations transactionnelles. Ces dernières gourmandes en disponibilité et rapidité d'exécution (temps réel) ne tolèrent pas la présence de données organisées de façon particulière, redondantes et surtout nécessitant des traitements de transformations très coûteux (les processus E.T.L, par exemple), pouvant pénaliser les performances des systèmes opérationnels.
Les systèmes d'information, lors de leur mise en place, étaient peu assujettis à des contraintes non fonctionnelles aussi fortes, ils ne nécessitaient donc pas les mêmes propriétés techniques que les systèmes opérants en terme de plate-forme technique.
C'est donc une séparation physique (et souvent une rupture technologique aussi) qui a été imposée aux systèmes d'information et les a "éloigné"» des systèmes opérationnels.
Les architectes d'information ont alors proposé non pas une cohabitation mais un éloignement de l'endroit où la donnée était produite pour construire les systèmes d'information.
Les notions d'ingestion, réplication, propagation, intégration sont alors apparues pour caractériser le flux entre l'endroit où la donnée était produite et celui où elle allait être utilisée.

La donnée s'est donc déplacée vers le traitement.

Cependant la démocratisation d'accès à l'information avec de plus en plus d'utilisateurs des systèmes d'information et des données émises de plus en plus importantes et de formats très variés, imposent une évolution aux systèmes d'information actuels où désormais certaines des contraintes non fonctionnelles des systèmes opérationnels deviennent les leurs (voir chapitre 3). Notamment autour des paramètres suivants :

— L'augmentation de la "masse" de ces données disponibles, devient de plus en plus importante, et une solution basique comme augmenter simplement la capacité de stockage ne suffit plus à répondre à cette problématique.

— La notion de coût, les problématiques de réplication, d'ingestion, de gouvernance (copies multiples de données qui entraînent une baisse de la qualité des données, par exemple) mais aussi de limitation de capacité de stockage sont des paramètres qui sont désormais étudiés.

A ces nouvelles contraintes non fonctionnelles s'ajoute la notion de données sensibles, ne pouvant être aisément accessibles voire même déplaçables ou copiables.
Ces contraintes ont donc des impacts très forts sur la conception des systèmes d'information et peut limiter les choix de solutions possibles.

Dans le cadre des lacs de données, il semble que, unanimement du côté industriel et de la littérature scientifique, la séparation technique des systèmes émetteurs de données (dont le système opérant) et des lacs de données ait été adoptée comme postulat, et pratique usuelle de conception d'architecture.
Une duplication, systématique, de toutes les données que l'on veut analyser et explorer est la stratégie d'organisation technique mise en œuvre dans les lacs de données. Le lac de données étant associé à la notion de réceptacle physique unique.
Le postulat est poussé très loin car ce réceptacle unique est toujours considéré par certains [17][22][75] comme étant une plate-forme Apache Hadoop. Ce qui simplifie ainsi la solution d'architecture ou de plate-forme technique pour les lacs de données.
En effet il y a une totale abstraction des paramètres d'architecture technique dans le choix des architectures applicatives des lacs de données qui sont tournés essentiellement autour des solutions incluant la technologie Apache Hadoop.
Russom [66] est le premier à envisager une hybridation de plusieurs technologies, Apache Hadoop restant tout de même la référence, ainsi que la duplication des données vers le lac de données.

Or nous pensons que la notion de gravité de la donnée n'est pas envisagée, voire même non prise en compte, dans la définition des architectures applicatives des lac de donnée et que l'approche de duplication et copie systématique de toutes les sources de donnée ne doit pas être l'approche de

référence.
Nous proposons d'étudier l'influence que peut apporter le gravité des données, si elle est prise en compte dans la conception des architectures des lacs de données.

Sur cette voie, nous nous sommes donc attachés à définir quels paramètres non fonctionnels sont pertinents dans les lacs de données et peuvent influencer la relation donnée-traitement.
Trois ont retenu notre attention : le volume (masse), le coût et la sensibilité :

— La « masse » de ces données disponibles devient de plus en plus importante, et une solution basique comme augmenter simplement la capacité de stockage ne suffit plus à répondre à cette problématique ;

— Le coût, lié aux problématiques de réplication, d'acquisition, de sécurité mais aussi d'extension de capacité de stockage doit être désormais évalué lors de la conception des architectures des lacs de données ;

— La sensibilité des données qui entraîne une gestion spécifique. S'il n'existe pas de définition légale pour les données dites sensibles, les nouvelles réglementations sur la donnée personnelle RGPD par exemple ou bien la cybersecurité, ou la Loi de Programmation Militaire étendent celle déjà délivrée par la CNIL. Chaque organisation peut, en plus de ces obligations de réglementation avoir sa propre classification de données dites sensibles.

Afin d'englober toutes ces notions autour de la sensibilité, nous définirons qu'une donnée est dite sensible au regard du degré de sécurité criticité/précaution que nécessitent son utilisation et son traitement. Cela va donc dépendre de l'évaluation par le propriétaire de ces données. Chacun est libre d'établir sa propre gradation. Cependant, quelques données sont naturellement plus sensibles que d'autres : les données clients, les données comptables et financières de l'entreprise, les données issues de la R&D etc.
Le niveau de sensibilité de ces données peut conduire à des accès limités, voire interdits rendant ainsi la donnée non déplaçable ou non duplicable. Les nouvelles régulations et diverses lois de conformité rendent en effet ces déplacements de données plus "lourds" car s'y ajoute aussi en plus de la sécurité, parfois l'anonymisation, l'encryptage mais aussi la traçabilité des déplacements et accès. Il est essentiel que chaque entreprise établisse la cartographie de ces données et en évalue le degré de criticité, pour en évaluer la réelle contrainte sur leurs déplacements.

Ces trois paramètres (volume, coûts, sensibilité) constituent des contraintes qui peuvent avoir des impacts très forts sur la conception des composants des systèmes d'information et peuvent réduire les choix d'architecture possibles voire remettre en cause un choix existant. Ces contraintes sont liées à

la donnée elle-même et nous proposons d'enrichir la vision des travaux de recherche précédents [3] et d'étendre le concept de gravité de la donnée en le fondant sur les paramètres masse-volume, sensibilité et coût.
Dans la section suivante nous étudions l'influence de la gravité des données sur les architectures des lacs de données.

5.3 Impact de la gravité de la donnée sur les architectures des lacs de données

Dans le cadre des lacs de données, le postulat de déplacement de la donnée vers son traitement a été adopté comme pratique d'architecture, ce qui se traduit par une duplication, systématique, de toutes les données que l'on veut analyser et explorer dans le lac de données. Dans le cadre des lacs de données cette duplication de données est associée à une technologie (Apache Hadoop), qui est considérée comme l'architecture applicative de référence devant recevoir les données collectées pour le lacs de données.
Or nous pensons que, dans ce postulat, la notion de gravité des données n'est pas envisagée, voire même non prise en compte, lors de la définition des architectures applicatives des lacs de données.
Notre proposition, si elle nous la validons, en prenant en compte la gravité des données, est de ne pas adopter une approche de duplication et copie systématique de toutes les sources de données dans le lac de données.
Dans cette section nous souhaitons étayer ce point et démontrer que la gravité des données peut jouer un rôle important dans les scénarios d'implémentation des lacs de données et de ce fait doit être prise en compte dans la phase de conception.
Pour chacun de ces facteurs (masse-coût-sensibilité) nous étudions les impacts potentiels pour une architecture de lac de données.

5.3.1 L'impact du volume sur les lacs de données

Les travaux de MacCrory[45] se focalisent sur l'augmentation de la masse/volume des données, dans la relation donnée-traitement, ils font l'analogie entre cette relation et celle existant entre deux corps soumis à la force de gravitation et positionnent ainsi la gravité des données.
L'augmentation du volume des données produites, et par suite de la masse des données, est un des paramètres de la gravité de la donnée. Si cette masse devient trop importante, d'après MacCrory[45], la gravité de la donnée va être telle que le traitement des données va être attiré vers elles, et donc va devoir être déplacé.
Dans le cadre des lacs de données, le volume des données peut très vite devenir important, au fil de

l'intégration de flux volumineux ou nombreux.
Dans le cas où le flux de données est peu volumineux, mais sa fréquence élevée (exemple les flux de 'streaming') le volume généré va être important. Dans ce cas-là si on considère la gravité des données, et la relation donnée-traitement, pour la partie ingestion du flux, il n'y aura pas d'influence et aucune raison de ne pas capturer le flux. On pourra le conserver dans le lac de données.
Le point de vigilance est à apporter sur le cycle de vie des données de ce flux et la définition d'une période de conservation du flux, avant de soit l'archiver, le résumer ou le supprimer, par exemple.
Dans le cas où la source de données est "importante" (volumineuse), par exemple une base opérationnelle, un historique de transaction ou un fichier de capture de comportement web pour une application mobile, leur duplication peut entraîner un déséquilibre dans la relation donnée-traitement, plus en faveur des données. Dans ce cas de figure, la gravité des données peut imposer au traitement d'être "attiré" par les données, il n'y aura alors pas duplication des données dans le lac de données.

Ce cas de figure peut fortement impacter la conception d'architecture des lac de données. Si les données ne sont pas dupliquées, mais sont nécessaires au lac de données, alors elles doivent être référencées dans le catalogue de métadonnées. C'est à l'architecture technique des lacs de données de permettre que le traitement, initié du lac de données puisse aller vers les données. On parle d'accès aux données en mode fédération.
Les techniques de traitements appliquées doivent alors pouvoir s'exécuter sur le système source, seul le résultat du traitement sera rapatrié dans le lac de données.
Cela signifie que l'accès en mode fédération des sources de données peut être un des moyens de constituer un lac de données, et que la duplication des sources de données n'est pas systématique.
Le système décisionnel, qui lui aussi peut être une des sources de données du lac de données, peut aussi envisager son accès en mode fédération et non être dupliqué.
Dans la figure 5.2 nous illustrons un lac de données où certaines données sont dupliquées physiquement à l'intérieur du lac de données "physique" et d'autres sont "virtuellement" dans le lac de données, via le catalogue des métadonnées, mais accédées en mode "fédération", le traitement se déportant vers les données (sous l'effet de la gravité).

Ce scénario d'architecture est une remise en cause du postulat de la totale duplication des données pour constituer un lac de données, via la prise en compte de la gravité des données.

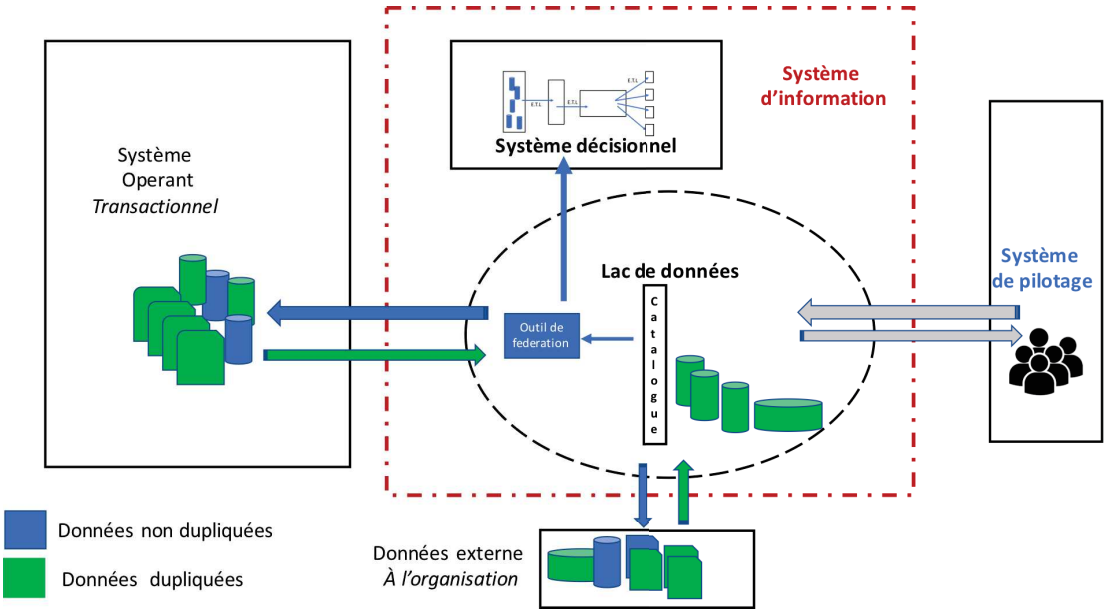


FIGURE *5.2: Architecture d'un lac de données mixte - en mode fédération et duplication*

5.3.2 L'impact de la sensibilité sur les lacs de données

Après le volume, nous intéressons à la sensibilité, que nous avons inclue dans la gravité des données (en nous référant aux travaux de Alrehamy et al. [3]), et sur comment elle pourrait influencer la relation donnée-traitement dans les lacs de données.

Dans leurs travaux Alrehamy et al. [3] constituent un lac de données dédié aux données personnelles. L'aspect sensibilité des données personnelle est tel qu'ils ont opté pour une architecture de type lac de données, où les traitements vont tous s'exécuter dans ce lac.

C'est l'étude de la relation donnée-traitement qui leur a fait choisir ce scénario, au regard d'une gravité des données si forte que c'est au traitement d'aller vers la donnée.

Au delà des données personnelles, dont la sensibilité s'est accrue notamment avec la réglementation européenne RGPD, la sensibilité des données peut concerner d'autres types de données dans une organisation, selon leur secteur d'activité.

Pour s'affranchir de cette sensibilité certaines techniques peuvent, et sont mises en œuvre, telle que l'anonymisation ou la pseudonymisation des données, qui permettent de manipuler et déplacer ces données en respectant leur sensibilité.

Cependant ces techniques peuvent faire perdre la valeur même des données qui ne sont alors plus exploitables.

L'encryptage est aussi une technique pour permettre le déplacement des données sensibles. Il permet de sécuriser le déplacement par exemple mais la donnée devra être déplacée vers un système offrant une continuité de cet encryptage.
Le niveau de sensibilité va lui aussi nécessiter un niveau de protection de la plate-forme qui héberge ces données, de très haut niveau, impliquant potentiellement un certain coût.
Déplacer la donnée, pour lui faire subir un traitement sur un autre environnement, tel que les lacs de données, peut impliquer un risque élevé, au regard de sa sensibilité et donc bloquer le déplacement de la donnée.
Cette problématique est présente dans le monde industriel soumis à d'importantes normes de conformité, directives et réglementations où la protection de la donnée est exigée.
L'exemple de la réglementation, pour les citoyens européens, RGPD qui en mai 2018 donnera aux citoyens européens des droits supplémentaires sur leurs données personnelles va impacter fortement la classification de ces données qui sont désormais considérées comme des données sensibles. Les lacs de données qui manipulent ces données sont eux aussi impactés et doivent intégrer cette contrainte.
L'exemple de la loi de programmation militaire, elle aussi, impose des niveaux de sécurisation, protection et traçabilité de certaines données qui impactent les systèmes voulant les utiliser.

Cette sensibilité est donc un élément crucial dans la relation donnée-traitement. Il doit, au même titre que le volume être intégré par conception et par défaut dans les architectures des lacs de données et peut conduire à mettre en place le scénario d'architecture hybride (duplication-accès en fédération) que nous évoquons dans le paragraphe 5.3.

Dans ce cas de figure, la gravité des données, via la sensibilité des données peut, influencer la relation donnée-traitement, et entraîner une non duplication des données vers le lac de données.

5.3.3 L'impact du coût sur les lacs de données

La duplication des sources de données pose la problématique non seulement au niveau de leur gouvernance, mais aussi sur le coût de l'extraction multiple d'une même donnée et son impact sur le système où elle est émise.
Au niveau de la gouvernance des données, multiplier les copies d'une même donnée peut entraîner une dégradation de sa valeur, engendrer des versions différentes, rendre complexe sa traçabilité et donc impacter sa qualité.
L'extraction multiple d'une même donnée engendre aussi une augmentation du coût de mise à disposition. En effet la mise à disposition de données ou sources de données a un coût sur le système émetteur ou

hébergeur de cette donnée : au niveau de son extraction, du stockage, même s'il peut être temporaire mais aussi au niveau des capacités physiques (mémoire, processeurs, etc.) mobilisées pour produire cette duplication.
La multiplication de ces sollicitations trop importantes peut donc représenter un frein à la mise à disposition de copie de données. Le volume de données à dupliquer et à extraire, ainsi que la fréquence peuvent aussi accentuer cet impact.
Un autre effet de la duplication de la donnée est le coût associé à sa traçabilité. En effet pour répondre à certaines réglementations (précédemment citées comme la RGPD), les données doivent être tracées, leurs accès et actions subies conservées, en vue d'audit par exemple. Ce traçage fait grossir les volumes des "logs" sur les différents serveurs, augmentant ainsi les coûts de traitement et de stockage de ces dernières.
Pour certaines organisations, comme les organismes financiers, cela peut représenter un important volume de données générées et à conserver au quotidien, comme nous le montrons dans la section 5.4 sur le cas d'usage industriel.

La duplication de données vers les lacs de données, génère donc un coût, qu'il convient d'évaluer dans la conception des lacs de données pour potentiellement envisager, comme pour le volume et la sensibilité un accès différent pour des données trop coûteuses à produire et à dupliquer.

Nous avons donc établi que trois facteurs tels que la masse-volume, la sensibilité et le coût du déplacement des données, dont nous avons formé l'hypothèse d'être inclus dans la gravité des données, peuvent remettre en cause la relation donnée-traitement au sein des lacs de donnés.
Nous proposons dans le cas où le lien donnée-traitement est affecté, de ne pas envisager la duplication des données concernées, de les référencer cependant dans le catalogue des métadonnées et d'y permettre l'accès en mode fédération, où seul le résultat du traitement pourra alors être transféré sur l'environnement physique du lac des données.

Si nous appliquons l'analogie de la gravité des données avec la vision physique ce sont les traitements qui utilisent ces données qui seront attirés à elles et pourront donc être déportés à l'endroit où elles se situent et non pas le contraire. C'est donc le traitement qui va aller vers la donnée et non plus la donnée que l'on va déplacer vers le traitement.

Notre proposition de prendre en compte la gravité des données pour la conception des lacs de données impacte donc l'architecture applicative des lacs de données, qui doit prévoir des logiciels permettant d'accéder des données en mode fédération, mais aussi leur architecture technique, qui doit

pourvoir supporter ces accès. Il faut donc explorer les possibilités techniques d'amener les traitements où résident les données désormais et envisager des solutions alternatives à la structure physique unique comme réceptacle des lacs de données.
Dans le cadre de nos travaux de recherche, nous collaborons avec certains industriels sur la conception de leur lac de données (voir chapitre 6), nous avons donc proposé à un de ces industriels de reconsidérer sa conception d'architecture des lacs de données (très récente) au regard de la gravité des données. Nous avons réalisé cette étude durant l'année 2017 et avons pu avoir un retour, que nous exposons dans la prochaine section.

La prochaine section étudie l'impact de la gravité des données sur les architectures applicative et technique d'un lac de données au travers un cas réel.

5.4 Étude de cas : prise en compte de la gravité des données sur un lac de données métrologie

5.4.1 L'approche et la méthodologie

Afin de vérifier notre hypothèse sur l'influence de la gravité des données dans les architectures des lacs de données, nous étudions un cas réel d'architecture de lac de données, auprès d'un industriel, et évaluons l'influence de la prise en compte de la gravité des données au travers des trois facteurs que sont le volume-masse, la sensibilité et le coût de déplacement des données, sur l'architecture applicative et technique de leur lac de données.
Au moyen d'interviews nous avons collecté les évaluations de ces trois paramètres chez l'industriel qui ont été faites initialement lors du choix des architectures applicative et technique. Nous avons ensuite procédé à notre propre estimation de ces paramètres et validé cette estimation par des mesures sur le lac de données de l'industriel et sur le système opérant, d'où proviennent une majorité des données sources. Au regard de cette nouvelle évaluation de ces trois facteurs nous avons étudié l'impact sur les architectures applicative et technique de leur lac de données.

5.4.2 Description de l'étude de cas industriel

5.4.2.1 Le contexte

Le cas d'usage que nous avons étudié est celui d'un organisme international du monde bancaire. Le besoin métier de cette organisation est de mieux valoriser leur capital de données afin de trouver des

nouvelles pistes d'information utiles. Dans ce but cette organisation a décidé de mettre en place des lacs de données. Trois lacs de données ont été définis :

— Un lac de données pour la métrologie ;

— Un lac de données dédiées aux données de leurs clients ;

— Un lac de données pour les besoins de reporting réglementaires et autres.

Le lac de données métrologie a été sélectionné comme étant le pilote en terme de technologie pour les deux autres. Son objectif de mettre en commun les données de métrologie provenant de tous les composant techniques informatique de l'organisation tels que :

— Serveurs ;

— Réseaux ;

— Baies de stockage.

Les données provenant de ces éléments du parc informatique doivent servir à :

— Améliorer la connaissance et le pilotage de ce parc ;

— Accélérer la détection et solution des pannes ;

— Prévenir les pannes ;

— Analyser les pics d'utilisation ;

— Aider à la planification de capacité nécessaires ;

— Rationaliser et optimiser le parc ;

— Réduire ses coûts.

Les choix d'architectures pour le lac de données métrologie sont décrits ci après.

5.4.2.2 L'architecture fonctionnelle

Le lac de données est un environnement de stockage unique, alimenté par la collecte de toutes les données de métrologie des éléments du parc informatique concerné.

5.4.2.3 L'architecture applicative

L'architecture applicative est pilotée par le choix de la technologie HortonWorks - HDP (décrite dans le chapitre 3.6), une plate forme basée sur Apache Hadoop et une suite d'outils d'aide à la manipulation, l'exploration et l'administration des données. Les données émises par les serveurs et autres sont poussées en temps réel dans le lac de donnée HDP et explorées par les utilisateurs du lac de données. Ces utilisateurs sont des utilisateurs "avertis". La figure 5.3 résume l'architecture en place au niveau applicatif, en

FIGURE *5.3: Plateforme HortonWorks*

reprenant les principaux composants de la plateforme HDP. On y retrouve les logiciels que nous avons décrits dans le chapitre 3.6, tels que Pig, Hive ou Spark.

5.4.2.4 L'architecture technique

L'architecture technique est basée sur des serveurs x86 (Lenovo, 2016). Huit serveurs sont dédiés à cette architecture et 10 exabytes d'espace de stockage y sont alloués. Cette architecture technique s'est basée sur l'étude des données émises par les sources de données que le lac de données veut intégrer. Une cartographie des données sources a été établie :

— nombre de serveurs de type x86 : 18000 ;

— nombre de serveurs de type Unix : 30 ;

— nombre de serveurs de type Mainframe : 6 ;

— nombre de baies de stockage : 50 ;

— type de Réseaux : LAN, MAN, WAN.

5.4.3 Évaluation initiale du volume, du coût et de la sensibilité

Lors de l'étude des architectures du lac de données métrologie, les paramètres volume-masse, coût et sensibilité ont été évalués par l'industriel, à l'aide d'un « poids », déterminé sur une échelle comprise entre 1 et 10 (où 10 est le poids le plus important). Ces paramètres ont été étudiés au travers de l'évaluation des contraintes non fonctionnelles, lors des architectures applicative et technique. Le volume a été considéré comme le facteur le plus important dans l'évaluation des contraintes non fonctionnelles.

Il a donc fait l'objet d'une estimation de la part de l'industriel, en vue d'évaluer la capacité de stockage nécessaire dans l'architecture technique. L'estimation communiquée par l'industriel est de 2 exabytes.
Le calcul a été effectué de la façon suivante :

Estimations	Serveur X86	Serveurs Unix	Serveurs Mainframe	Baies de stockage	Réseaux
Ev	12 GB	20 GB	1 GB	20GB	900 GB
Ej	311	0, 86	0, 0086	14	3, 89

Dans ce tableau :
Ev représente l'estimation de volume de données par minutes et par serveur.
Ej représente l'estimation journalière de volume de données par type serveur.
Ej se calcule de la façon suivante :

Ej = Nb de serveurs * 24*60*Ev (en Peta Bytes)

Ce qui donne un volume estimé de 330 Peta Bytes environ par jour pour le lac de données métrologie.
Sachant que peta = 10^{15} et Exa = 10^{18}
Sachant que la conservation historique des données est de 1 semaine

Ej= 2, 3 Exa byte de données

Soit le volume du lac de données estimé à : 2, 3 Exa Bytes de données

Dans le projet initial, le coût de déplacement des données n'a pas fait l'objet d'une estimation lors des choix d'architecture.
La sensibilité a été évaluée, pour les données de métrologie, comme faible et donc n'a pas été prise en compte lors des choix d'architectures applicative et technique.
L'industriel a évalué d'autres contraintes non fonctionnelles :

- Temps réel ;
- Sensibilité ;
- Volumétrie ;
- Sécurité ;
- Sauvegarde ;
- Fiabilité ;
- Disponibilité ;

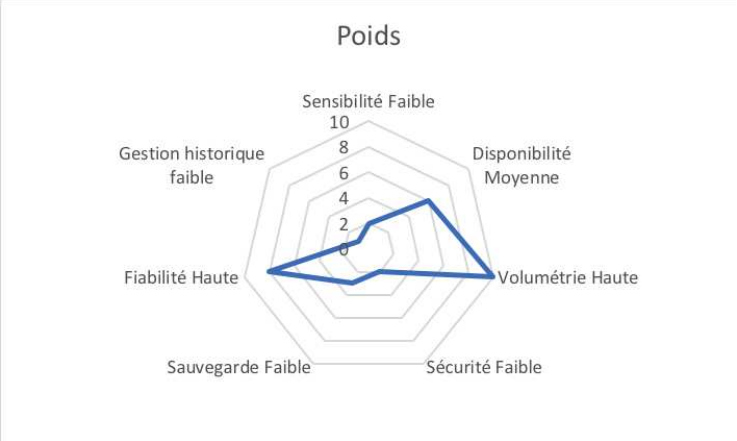


FIGURE *5.4: Évaluation initiale des contraintes non fonctionnelles pour le lac de données métrologie*

Fréquence	Haute	Poids
Sensibilité	Faible	2
Disponibilité	Moyenne	6
Volumétrie	Haute	10
Sécurité	Faible	2
Sauvegarde	Faible	3
Fiabilité	Haute	8
Gestion de l'historique	faible	1

TABLE *5.1: Tableau d'évaluation initiale des contraintes non fonctionnelles pour le lac de données métrologie*

— Gestion de l'historique.

Le graphe 5.4 et le tableau 5.1 synthétisent l'évaluation de l'industriel de ces contraintes non fonctionnelles.

En vue d'évaluer la gravité des données sur ce lac de données nous avons procédé à des mesures et estimations de ces trois paramètres, notamment la partie coût de déplacement des données qui n'a pas fait l'objet d'une étude lors des choix d'architectures. La section suivante expose ces résultats.

5.4.4 Évaluation de la gravité des données sur le lac de données métrologie

L'observation de chaque paramètre a été faite sur un mois afin d'intégrer les pics d'activités de l'industriel et avoir des valeurs représentatives.

5.4.4.1 Le volume

Le volume estimé est de 2, 3 Exabytes de données pour le lac de données métrologie. Il est classifié comme important mais n'impose pas à la donnée de ne pas pouvoir être déplacée.
Cependant nous alertons sur les hypothèses de calcul qui ont été faites par l'industriel. En effet une gestion du cycle de vie de la donnée a été imposée à un mois de conservation d'historique. À la vue des explorations souhaitées dans le lac de données, comme par exemple faire de la fouille de données ou du « machine learning », une conservation d'historique plus grande pourra être nécessaire, ce qui va impliquer une augmentation du volume de données. Cette augmentation va se traduire par une extension de la capacité de la plateforme HortonWorks, et donc impacter le coût de la solution. Ce coût supplémentaire, même s'il n'a pas été évalué, n'est pas considéré comme un frein aux architectures en place.

A ce stade de l'étude, le volume seul n'est pas jugé assez influant pour bloquer le déplacement des données mais nous émettons une alerte sur l'estimation qui en faite.

5.4.4.2 La sensibilité

La sensibilité des données a été classifiée comme faible lors de la conception de l'architecture fonctionnelle. Or l'organisme bancaire est soumis à la Loi de Programmation Militaire[1] et certaines données transitant par ses réseaux doivent être protégées car jugées sensibles. Les données de métrologie provenant notamment des serveurs de type mainframe sont elles classifiées hautement sensibles.
De plus le cas de la métrologie n'est pas représentatif du "poids" de la sensibilité, donnée initialement, pour les futurs autres lacs de données, notamment celui des données clients qui va être soumis à la régulation européenne RGPD sur les données personnelles.
Ce facteur n'a dons pas été évalué correctement lors de l'architecture fonctionnelle et peut remettre en cause le déplacement de certaines données. Le tableau suivant réévalue la sensibilité des données selon leur provenance :

Évaluation	Serveur X86	Serveurs Unix	Serveurs Mainframe	Baies de stockage	Réseaux
Sensibilité	2	6	10	8	9
Évaluation	Faible	Moyenne	Haute	Haute	Haute

Nous avons de même reproduit le diagramme de la sensibilité par type de serveur sur la figure 5.5.

Le paramètre de sensibilité doit donc être revu notamment pour les données provenant des serveurs

1. https ://www.agefi.fr/banque-assurance/actualites/hebdo/20170720/cybersecurite-enjeu-reglementaire-223387. Cette loi impacte les industriels classés comme opérateurs d'importance vitale (OIV) pour l'économie française.

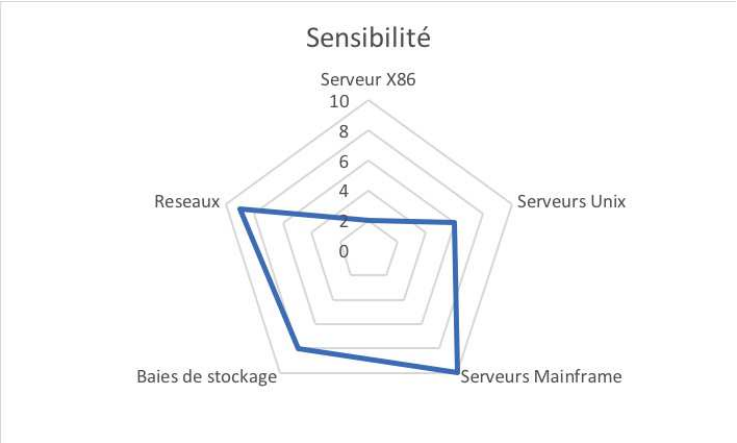


FIGURE *5.5: Évaluation de la sensibilité par type de serveur, pour le lac de données métrologie*

mainframe. Cette haute sensibilité pourrait bloquer le transfert de certaines données de ce serveur vers le lac de données HDP.

5.4.4.3 Le coût

Au vu de la sensibilité très haute des données provenant des serveurs mainframe, nous avons concentré l'étude du coût de déplacement des données sur les serveurs mainframe. Nous avons calculé le coût de déplacement de 1 TB de données par jour du serveur mainframe vers un autre serveur. Ce coût se mesure en "million instructions per second" (MIPS) qui est l'unité de facturation du serveur.
L'évaluation de la puissance machine hébergeant les données, du coût de déplacement de 1 TB de données sont les suivantes :

— Utilisation de 4 cœurs de processeurs sur un mainframe de type z13 qui est utilisé à 85% Cela correspond en unité de mesure mainframe à 519 MIPS par jour.

Le coût par jour est donc de 6756, 4$ (prix moyen observé pour 519 MIPS)[2]

— sur une année le coût est donc estimé à 2 466 103$

A ce coût doit être ajouté le coût d'administration et de maintenance du serveur, une étude fait état d'un coût moyen de 98 482$.

— Soit un coût total de réplication de 1 TB de donnée de 2, 55 $M

2. Ces données sont des données officielles, moyennes, indiquées par le constructeur. https ://searchdatacenter.techtarget.com/answer/Converting-CPU-hours-to-MIPS-Mainframe-capacity-planning-process

Le volume estimé par jour de données pour la métrologie à répliquer est estimé à 8, 6 TB selon le tableau de la section 5.4.3, soit un coût sur une année de plus de 22 M$.
Le déplacement des données des serveurs mainframe a donc un coût très important, qui n'avait pas été estimé initialement.

5.4.5 Conclusion du cas d'étude de lac de données métrologie

Nous avons donc évalué trois facteurs (volume-masse, sensibilité et coût) pour déterminer si la gravité des données avait un impact sur l'architecture applicative et technique du lac de données métrologie.
Si le volume ne semble pas être un facteur déterminant, le coût et la sensibilité peuvent potentiellement faire reconsidérer certains choix d'architecture applicative et technique. La relation donnée-traitement entre le lac de données et les serveurs de type mainframes (qui sont les principaux systèmes opérants de l'industriel) doit être reconsidérée. Une duplication systématique de ces données pourraient ne pas être possible, à moyen terme.
La gravité des données, notamment au travers la sensibilité et le coût, a donc une influence sur l'architecture du lac de données.

Le tableau suivant donne une synthèse de nos résultats :

Gravité	Serveur X86	Serveurs Unix	Serveurs Mainframe	Baies de stockage	Reseaux
Volume	8	4	2	7	6
Sensibilité	2	6	10	8	9
Cout	3	5	10	5	3

Nous pouvons en conclure que l'architecture initiale présente des failles. Dans notre cas d'étude, l'évaluation de la gravité des données devrait imposer une évolution de l'architecture du lac de données métrologie, sous l'influence des données provenant des serveurs de type mainframe.
Le scénario d'architecture que nous avons proposé dans la section 5.3.2 peut être une évolution de l'architecture applicative existante, sans la remettre en cause. Certaines donnés pourront être dupliquées dans le lac de données HDP et certaines rester sur leur système source, pour être accédées lorsque des explorations les solliciteront, en mode fédération. Le résultat du traitement de l'exploration pourra être transféré sur le lac de données HDP.

Le principal objectif poursuivi dans cette étude de cas est celui de répondre à la question : qu'est-ce

qui peut remettre en cause le choix d'une architecture fédératrice mono technologique des lacs de données ?
Notre hypothèse est liée à la gravité des données telle que nous l'avons étendue en intégrant trois facteurs : volume-masse, sensibilité et coût. Le cas d'étude met en lumière le fait que le coût du déplacement des données vers leurs traitements est à prendre en compte. Son importance, dans ce cas d'étude, implique une remise en cause de l'architecture fédératrice unique de toutes les données disponibles de la métrologie et qu'un scénario d'architecture hybride doit être envisagé.
Ce premier constat entraîne la modification d'accès et d'intégration des données métrologie du serveur mainframe, ce qui impacte l'architecture applicative mise en place. Une première solution d'accès en mode fédération, où les données restent en place, est à explorer 5.3.2.
Si les données sont jugées trop sensibles, les traitements du lacs de données utilisant ces données devront se faire où elles résident et donc c'est le traitement complet qui va devoir être déporté et seul le résultat, s'il n'est pas jugé sensible, pourra être exporté vers la plateforme HortonWorks.
Le lac de données pourrait donc avoir deux zones de stockage : une sur HortonWorks et une sur les serveurs mainframe.
Cela ouvre la porte aux architectures de lacs de données hybrides et non plus mono technologiques, qui peuvent intégrer différentes zones de stockage, sur des types de serveurs divers, utilisant des technologies différentes.

Dans le cadre de notre étude nous avons évalué l'impact de la gravité des données sur un lac de données «in situ». Une perspective à nos travaux de recherche est d'étudier l'impact de ce facteur dans la décision de positionner un lac de données «in situ» versus dans les «nuages». Dans ce cas, la sensibilité des données personnelles en particulier va nécessiter d'aborder les aspects de confidentialité via-à-vis du prestataire mais aussi du fournisseur d'accès. Cela va entraîner des problématiques supplémentaires et générer un coût de gestion qui devra être évalué.

5.5 Synthèse du chapitre 5

Dans ce chapitre nous avons introduit le concept de gravité des données, en tant que contrainte non fonctionnelle, pouvant influencer la conception d'architecture des lacs de données, notamment la relation donnée-traitement. A partir de travaux de MacCrory[46] et [3], qui incluaient le volume et la sensibilité comme facteurs compris dans la gravité, nous avons proposé d'y inclure l'évaluation du coût du déplacement des données.

Après avoir évalué l'impact de ce que ces trois facteurs pouvaient induire sur l'architecture des lacs de données, nous avons proposé un scénario d'architecture alternative, où toutes les données ne sont pas dupliquées physiquement dans le lac de données mais où certaines peuvent rester sur le système source qui les produit.
Cette proposition entraîne une architecture non plus mono technologique, mais hybride, dans laquelle d'autres zones de stockage, virtuelles, car non matérialisées dans le même environnement techniques mais référencées dans le catalogue de métadonnées, peuvent être englobées dans le lac de données au sens logique.

Nous avons étayé cette proposition par une étude de l'influence de la gravité des données, sur un lac de données industriel. L'étude que nous avons faite a démontré qu'effectivement, si la gravité des données est importante, pour certaines données sur certains serveurs, il faut envisager une alternative au scénario d'architecture en place. La duplication systématique des données pour alimenter le lac de données ne doit pas être la méthode systématique pour constituer un lac de données.

Au travers de cette étude, nous avons remis en cause la conception des lacs de données, qui repose sur certains postulats, qui s'avèrent défaillants. Cette étude fait apparaître le manque de conceptualisation dans la constitution des lacs de données.

Notre intérêt s'est donc porté sur l'exploration des approches de conceptualisation ou représentation pouvant s'appliquer aux lacs de données.
Nous nous sommes intéressés au méthodes de représentation qui s'affranchissent des produits logiciels, comme l'approche MDA (Model Driven Architecture), qui semble correspondre à notre recherche.

Le chapitre suivant est dédié à l'amorce de cette approche pour tenter de proposer une représentation des lacs de données.

Chapitre 6

Contribution à une démarche de formalisation des lacs de données via une approche ligne de produits

Nous souhaitons explorer une approche de formalisation constituant une aide à la mise en place de projets d'architecture (s'inscrivant dans une stratégie de développement d'architecture des systèmes d'information, conduisant à des produits pour un usage donné).

L'ingénierie des lignes de produits constitue une approche qui permet la formalisation d'une série de produits ou systèmes logiciels semblables qui ne différent que par des composants optionnels. Elle s'affranchit des logiciels[1] mais prend en compte les principaux composants ou fonctionnalités que nous avons identifiés (voir chapitre 4) et par suite la formalisation obtenue permet un gain considérable en termes de coût, de temps et de qualité.

6.1 Nos attentes

Le concept de lac de données est né de la mouvance des données massives et de la technologie Apache Hadoop. Sa conception est partie des logiciels présents sur le marché industriel, et s'est donc focalisée sur une technologie essentiellement. Nous avons vu dans les précédents chapitres que l'association lac de données - Apache Hadoop était très limitative et ne correspondait plus vraiment aux attentes des organisations, ainsi qu'au concept de lac de données. Nous avons vu ses évolutions en terme d'architecture avec l'apparition des architectures hybrides.

L'idée des lignes de produits logiciels vient de la perception que dans beaucoup de domaines, les applications ne sont pas des systèmes isolés, mais partagent entre elles des besoins, des fonctionnalités et

1. mais peut aller jusqu'au niveau logiciel si nécessaire.

des propriétés. L'idée générale des lignes de produits logiciels est de profiter de ces points communs pour définir une architecture de base à partir de laquelle de nouvelles applications pourront être construites plus facilement, plus rapidement et avec un meilleur niveau de qualité.
A ce jour, il n'y a pas eu de travaux sur la formalisation des lacs de données, dans la littérature scientifique. Les industriels ont posé la problématique, proposé essentiellement des solutions logicielles mais n'ont jamais abordé la formalisation des lacs de données [10].
Dans le cadre de nos travaux, nous voulons expérimenter une approche ligne de produit et évaluer la pertinence de cette approche pour une formalisation des lacs de données.
Nos attentes autour de cette approche ligne de produit sont donc de :

— proposer une liste de composants minimum à mettre en place pour faire fonctionner un lac de données sans que ce dernier soit transformé en marécage ;

— d'amorcer une démarche de formalisation pour les lacs de données.

Notre objectif dans cette approche ligne de produits est d'arriver à la production d'un *feature model* (FM) (ou modèle de caractéristiques).
Les *feature models* sont des modèles de variabilité [8]. Leur but est de caractériser quels éléments d'une LPL sont communs à tous les produits, lesquels peuvent varier d'un produit à un autre et comment ces éléments peuvent varier. En d'autres termes, ils modélisent les exigences communes et variables dans la LPL. Les FMs sont utilisés comme support pour plusieurs tâches de l'ingénierie des LPLs, principalement pour la représentation d'informations, mais aussi pour définir le périmètre de la LPL, son évolution et sa maintenance, la réalisation d'opérations de conception ou bien la dérivation de produits. La figure 6.1 représente le FM que nous avons obtenu lors de nos travaux.
L'obtention des FMs peuvent permettre d'indiquer le caractère obligatoire ou optionnel (par exemple) de certaines fonctionnalités et donc faciliter la tâche de l'architecte qui pourra ainsi indiquer le caractère obligatoire ou optionnel (par exemple) de certaines fonctionnalités des composants à mettre en oeuvre pour les lacs de données

Afin d'obtenir ce "FM", nous utilisons nos connaissance industrielles, sur plusieurs lacs de données industriels, pour constituer une base de connaissance nécessaire à la formalisation ligne de produit, pour cela un référentiel des fonctionnalités d'un lac de données doit être créé, ce que nous expliquons dans la section suivante.

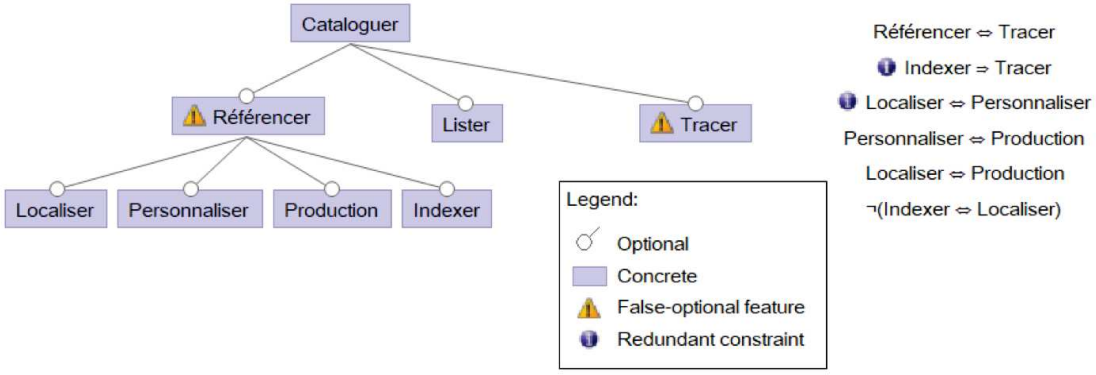


FIGURE *6.1: Feature Model de la fonctionnalité cataloguer*

6.2 Modélisation des fonctionnalités d'un lac de données

Notre premier travail dans cette approche ligne de produits a été de proposer un référentiel de fonctionnalités. Pour cela, nous somme partis des composants établis dans le chapitre 4 : l'acquisition, le catalogage, le stockage, l'exploitation, l'exploration, la gouvernance (cycle de vie, sécurité, qualité). Pour chacun de ces composants, nous avons établi quelles tâches possibles pouvaient être effectuées autour de la fonctionnalité et en avons fait une modélisation (voir figure 6.2).
Lors de ces travaux nous avons établi six grandes fonctionnalités :

— Acquérir

— Cataloguer

— Gérer le cycle de vie

— Exploiter

— Sécuriser

— Stocker

Nous considérons ces six fonctionnalités comme fondamentales (mais non exhaustives) dans notre démarche expérimentale, et basons notre référentiel sur celles-ci.

Pour chacune de ces fonctions, nous avons donc recensé les tâches/actions possibles dans le lac de données, basé sur la littérature scientifique (assez limitée sur cet aspect [17][26][3][9][48][53]) et sur diverses sources du monde industriel ([66][10][18][67]).

Par exemple pour la fonctionnalité "acquérir", nous avons fait la distinction entre données structurées et non structurées (nous n'avons pas distingué les données semi-structurées, ces deux types étant assez représentatifs pour la démarche expérimentale) mais pour chacune les tâches restent quasi identiques au niveau modélisation, ce qui les différencient ce sont les techniques utilisées pour réaliser les tâches dans le lac de données.

Nous avons aussi fait une distinction entre temps réel et temps différé (mode batch). Nous aurions pu aussi reprendre les trois modes d'acquisition cités par [53], mais dans notre démarche expérimentale, deux distinctions pouvaient suffire. Tout comme la distinction entre données structurées et non structurées, les tâches restent quasi identiques au niveau modélisation, ce qui peut les différencier ce sont les techniques utilisées pour réaliser les tâches dans le lac de données.
Nous avons donc identifié comme tâches principales : Extraire, Dupliquer, Connecter. Ce qui a fixé notre premier référentiel pour la fonction **acquérir**.
Nous avons procédé de même pour les cinq autres fonctions :

- **Cataloguer** : Lister, Typer (sensibilité en particulier), Référencer, Localiser, Personnaliser, Responsabiliser, Production (Produire), Indexer, Lister Facettes, Qualifier (Qualité), Tracer.
- **Gérer le cycle de vie** : Fonctionner, Effacer, Agréger, Résumer, Purger (Effacer pour des raisons techniques), Sauvegarder, Archiver.
- **Exploiter** : Préparer, Enrichir, Agréger, Étiqueter, Reconnaissance de forme, Classifier, Nettoyer, Filtrer, Réconcilier, Corréler, Transformer, Explorer/Produire, Naviguer, Décrire, Statistiques, Motifs ou Règles, Indicateur/Reporting, Segmenter, Prédire, Prescrire, Inférer (Systèmes Experts), Requêter, Publier, Administrer.
- **Sécuriser** : Protéger, Crypter, Confidentialité, Anonymiser, Auditer, Mettre en conformité.
- **Stocker** : Physique, Virtuel.

Une fois ce référentiel expérimental de fonctionnalités établi, nous avons construit notre base de connaissance, à partir de nos expériences industrielles.

6.3 Constitution de la base de connaissance des lacs de données industriels

Nous avons étudié six cas industriels de lac de données, sous l'angle de ce référentiel est indiqué pour chacun d'entre eux, quelles fonctionnalités étaient présentes et quelles "tâches" étaient effectuées dans le lac de données.

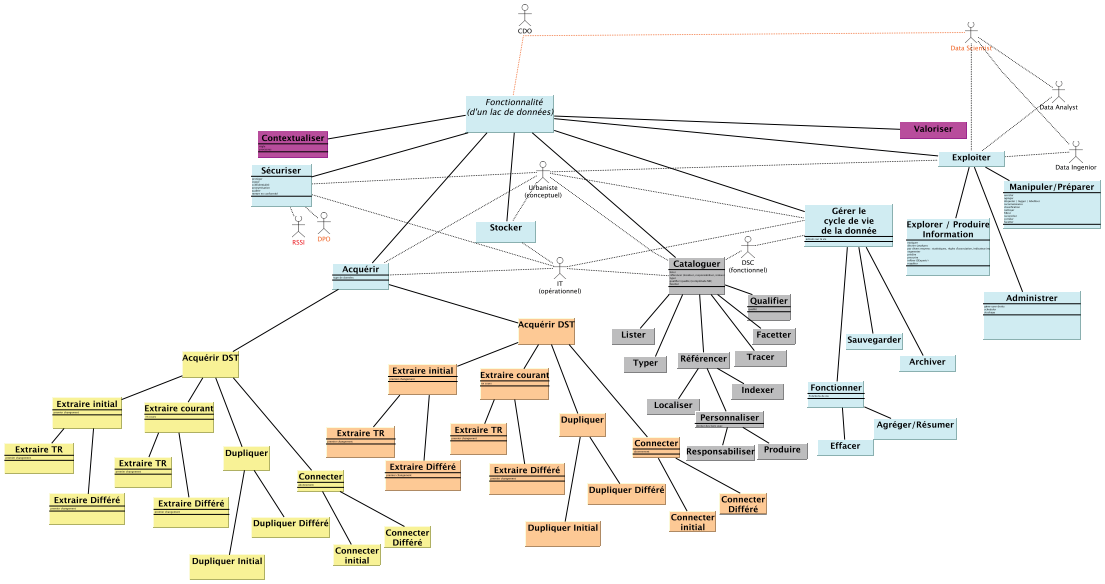


FIGURE *6.2: Modélisation d'un lac de données par fonctionnalités*

Pour cela, nous avons constitué dans un tableur une matrice par fonctionnalité et par onglet. Un extrait est présenté dans la figure 6.3, le détail des autres fonctions se trouve en annexe de ce manuscrit.

En ayant constitué un référentiel des fonctionnalités des lacs de données et une base de connaissance, nous pouvons commencer notre démarche de formalisation basée sur le modèle ligne de produit.
Nous détaillons notre démarche dans la section suivante.

6.4 Notre démarche

La démarche que nous avons appliquée est la suivante :

— génération du contexte formel ;

— production de l'AFC ;

— obtention d'un treillis version simplifiée, complète sous forme d'ACposet ;

— constitution automatique de l'ECFD ;

— obtention manuelle d'un FM.

Nous avons appliqué cette méthodologie sur les six tables dérivées de notre base de connaissance.

					Client 1	Client 2	Client 3.i	Client 3.n	Client 4	Client 5	Client 6
Acquérir					x	x	x	x	x	x	x
	Données structurées				x	x	x	x	x	x	x
		Extraire			x	x	x	x	x	x	x
			Initial		x	x	x	x	x	x	x
				Temps réel							
				Temps différé	x	x	x	x	x	x	x
			Courant		x	x	x	x		x	x
				Temps réel	x	x	x	x		x	x
				Temps différé	x	x	x	x			x
		Dupliquer			x	x	x	x	x	x	x
			Initial		x	x	x	x	x	x	x
				Temps réel					x	x	x
				Temps différé	x	x	x	x	x	x	x
			Courant		x	x	x	x		x	x
				Temps réel		x				x	x
				Temps différé	x	x	x	x		x	x
		Connecter (flux)			x	x			x	x	
			Initial			x			x	x	
				Temps réel		x			x		
				Temps différé					x	x	
			Courant		x	x				x	
				Temps réel		x				x	
				Temps différé	x					x	
	Données non structurées				x	x			x		
		Extraire				x			x	x	
			Initial			x			x	x	
				Temps réel						x	
				Temps différé		x			x	x	
			Courant			x				x	
				Temps réel		x				x	
				Temps différé		x				x	
		Dupliquer			x	x			x	x	
			Initial		x	x			x	x	
				Temps réel					x	x	
				Temps différé	x	x			x	x	
			Courant		x	x				x	
				Temps réel	x					x	
				Temps différé	x	x				x	
		Connecter			x	x				x	
			Initial			x				x	
				Temps réel		x				x	
				Temps différé						x	
			Courant		x	x				x	
				Temps réel	x	x				x	
				Temps différé						x	

FIGURE *6.3: Base de connaissance - fonctionnalité Acquérir*

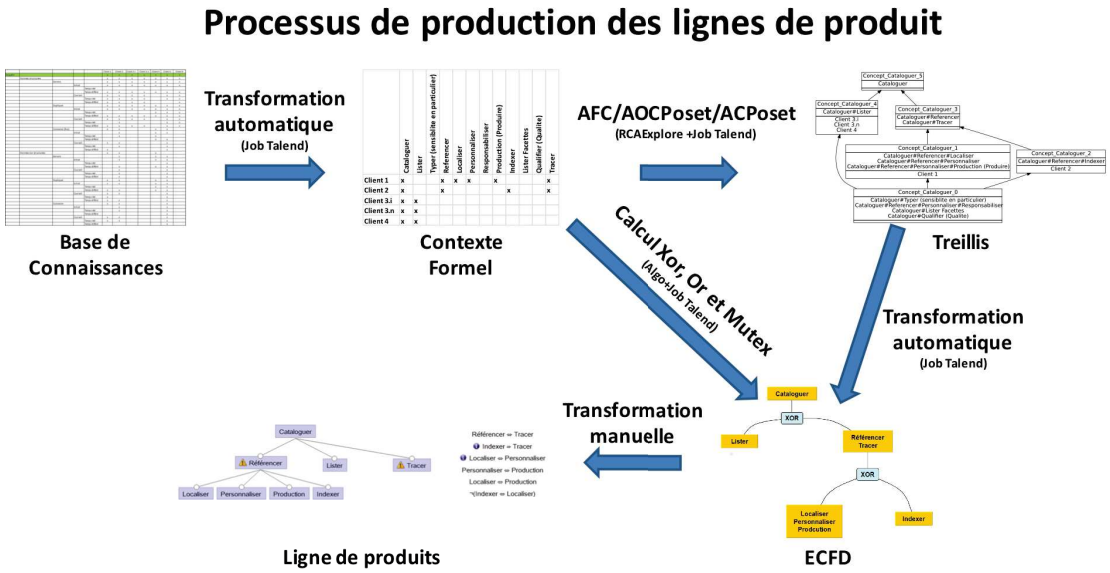


FIGURE *6.4: Processus de production des lignes de produit appliqués dans ces travaux.*

Lors de la constitution de la base de connaissance nous avons créé en réalité deux axes de constitution de la base de connaissance : un par fonctionnalités et un par composant logiciel. Dans le cadre de notre démarche expérimentale sur la démarche ligne de produits, appliquée aux lacs de données, nous avons opté pour l'exploitation seulement de l'analyse via les fonctionnalités. Pour autant, nous amorçons quelques pistes, dans nos perspectives, via l'exploitation de la base de connaissance par composant, que nous souhaitons explorer dans nos travaux de recherche ultérieurs.
Nous avons ensuite fragmenté en six grandes fonctionnalités, afin d'obtenir des FMs lisibles. Ce qui nous a donné six points d'entrée pour la démarche ligne de produits, que nous avons appliquée.

La base de connaissance est continuellement enrichie par les nouvelles collaborations industrielles. Nous avons d'ailleurs démarré la constitution de cette base avec quatre cas, puis en avons eu six. Cet enrichissement de la base de connaissance nous a permis d'observer, sans les analyser de façon détaillée, une évolution des FM. Le nombre de cas dans notre base de connaissance peu paraître encore peu élevé et de ce fait nos interprétations des treillis, pourraient être discutées de part leur faible représentativité. Pour compenser ce manque de données de masse pour constituer notre base de connaissance, nous avons systématiquement pratiqué une validation métier/expérience des résultats.

Dans notre démarche, nous avons réutilisé un processus existant (semi-automatisé) que nous illustrons

dans la figure 6.4.
Pour chaque étape nous avons manipulé quelques termes utilisés en ingénierie des produits logiciels, tels que : l'analyse formelle de concept (AFC), le concept et le treillis de concept. Nous décrivons, de façon très de simplifiée, ce qu'ils représentent dans la prochaine section.

6.5 L'approche ligne de produit - éléments de vocabulaire

Notre acquisition du vocabulaire minimum pour appliquer une démarche "ligne de produits" se base sur les travaux de J. Carbonnel [37] et [8] sous la direction de M. Huchard qui présentent une synthèse des principaux termes de ce domaine, que nous reprenons dans ce chapitre.
Nous nous sommes basés aussi sur les travaux de Clements et Northrop [1] qui définissent le concept de ligne de produit logicielle (LPL) comme **un ensemble de systèmes à logiciel prépondérant partageant de manière cohérente un ensemble de caractéristiques communes. Ces dernières satisfont aux besoins spécifiques d'un segment particulier du marche ou à une mission, et sont développées depuis un ensemble commun d'actifs clés de manière prescriptive.**

Software product line is a set of software-intensive systems sharing a common, managed set of features that satisfy the specifc need of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

Clements et Northrop [1] (traduction p. 206)

L'analyse formelle de concepts (AFC) [20] est un formalisme mathématique qui permet de structurer un ensemble d'objets décrits par des attributs. Un contexte formel représente une relation binaire entre les objets et les attributs qui définit le fait que "l'objet O possède l'attribut A" (voir figure 6.8). A partir d'un contexte formel, l'AFC permet d'extraire un ensemble ordonné de concepts.

Un concept est un ensemble maximal d'objets partageant un ensemble maximal d'attributs. Muni d'un ordre partiel, l'ensemble des concepts extraits du contexte formel forme une structure appelée **treillis de concepts**.La figure 6.5 schématise un concept.

Un treillis est un ensemble partiellement ordonné dans lequel chaque paire d'éléments possède une borne inférieure et une borne supérieure. La figure 6.6 schématise un treillis de concept.

Un "ECFD"ou Equivalence Class Feature Diagram en anglais :

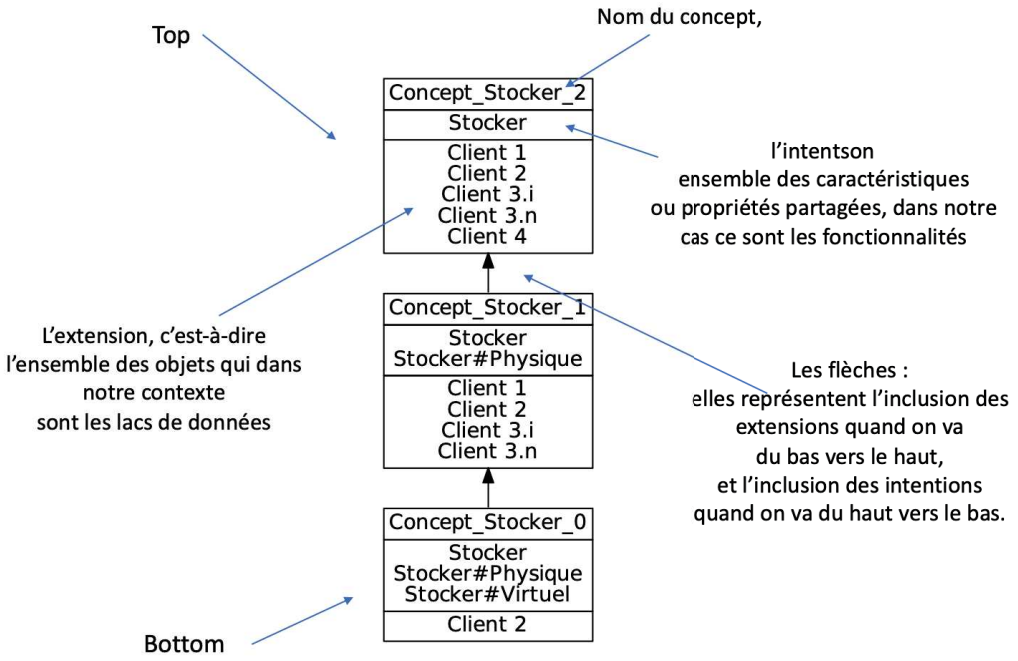


FIGURE *6.5: Un concept*

Description d'un treillis

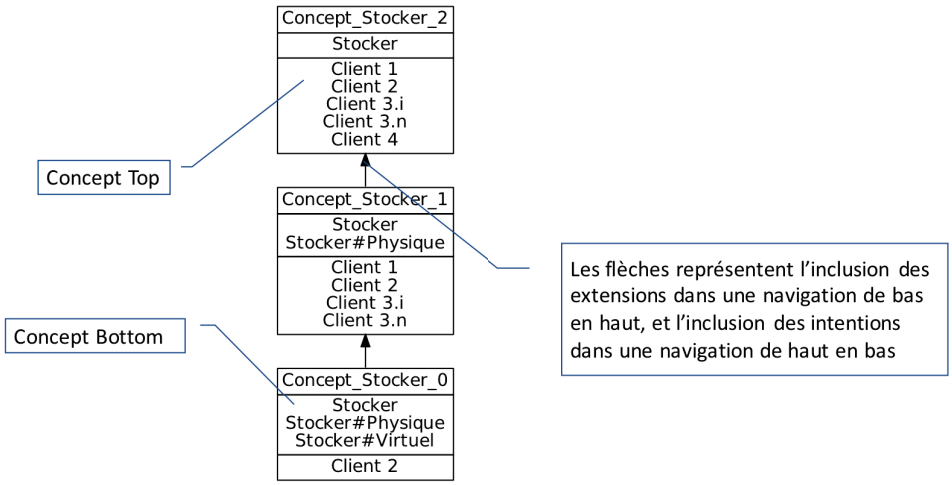


FIGURE *6.6: Un treillis de concepts*

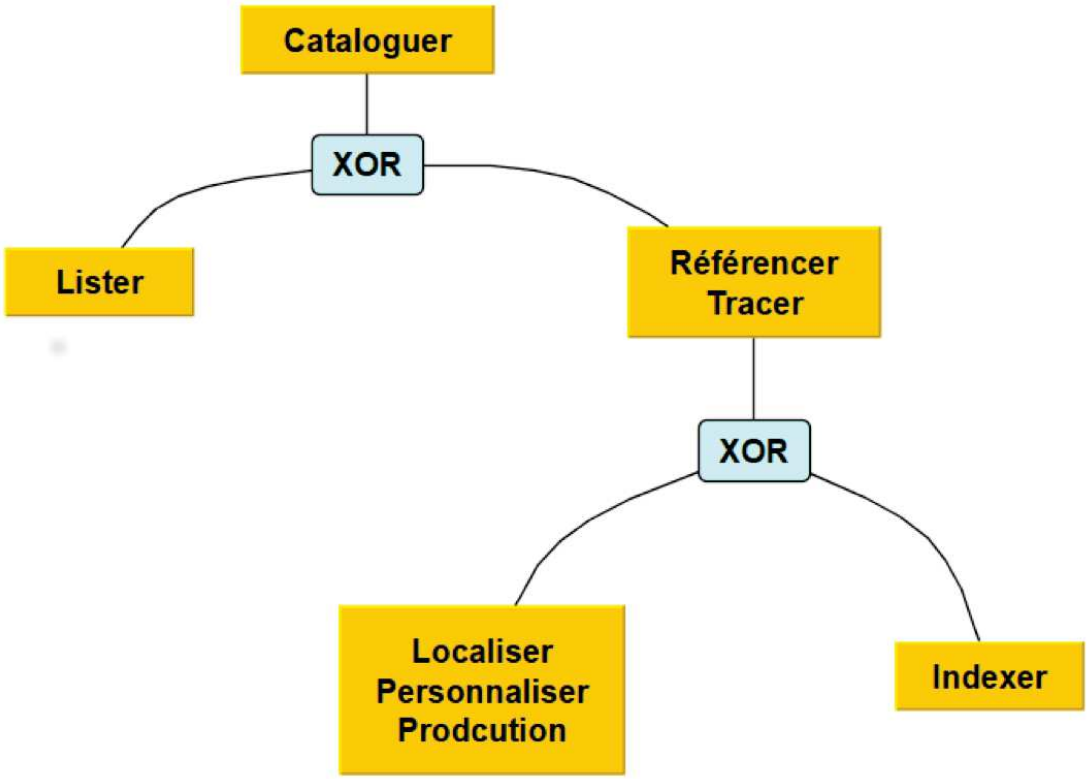


FIGURE *6.7: Equivalent Class Feature Diagram - ECFD*

Un EFCD est un ensemble de relations logiques obtenu à partir de descriptions de variantes.
Les ECFD sont une forme normale qui donne une représentation unique de cette variabilité et engendre plusieurs FMs. La figure 6.7 illustre un des ECFD que nous avons obtenus lors de nos travaux.

Nous avons adopté une démarche très simplifiée dans notre expérimentation de l'approche ligne de produits, le domaine étant très vaste et riche de plusieurs centaines de travaux de recherche, pour lesquels nous n'ambitionnons pas d'apporter une extension de connaissance au travers nos travaux.

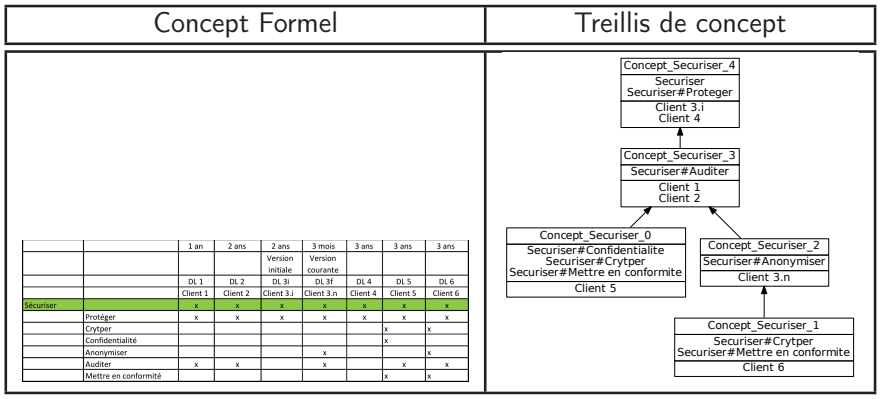
6.6 Application de notre démarche

Grâce au processus semi-automatisé et notre base de connaissance nous avons pu générer les contextes formels associés à chacune des fonctionnalités.

				Client 1	Client 2	Client 3.i	Client 3.n	Client 4	Client 5	Client 6
Cataloguer				x	x	x	x	x	x	x
	Lister					x	x	x	x	x
	Typer (sensiblité en particulier)								x	x
	Référencer			x	x				x	x
		Localiser		x						
		Personnaliser		x					x	x
			Responsabiliser						x	x
			Production (Produire)	x					x	
		Indexer			x				x	
	Lister Facettes									
	Qualifier (Qualité)								x	x
	Tracer			x	x					x

	cataloguer	Lister	typer	referencer	localiser	personaliser	responsabiliser	produire	Indexer	lister	qualifier	tracer
Client 1	X			X	X	x		x				x
Client 2	X			X					X			X
Client 3.i	X	X										
Client 3.n	X	X										
Client 4	X	X										
Client 5	X	X	X	X		X	X	X	X		X	
Client 6	X	X	X	X								X

FIGURE *6.8: Création du contexte formel pour la fonctionnalité Cataloguer, à partir de la base de connaissance*



		1 an	2 ans	2 ans	3 mois	3 ans	3 ans	3 ans
				Version initiale	Version courante			
		DL 1	DL 2	DL 3i	DL 3f	DL 4	DL 5	DL 6
		Client 1	Client 2	Client 3.i	Client 3.n	Client 4	Client 5	Client 6
Sécuriser		x	x	x	x	x	x	x
	Protéger	x	x	x	x	x	x	x
	Crytper						x	x
	Confidentialité						x	
	Anonymiser				x			x
	Auditer	x	x		x		x	x
	Mettre en conformité						x	x

FIGURE *6.9: Contexte formel et treillis de concepts associés à la caractéristique "Sécuriser" des lacs de données*

Le tableau 6.8 montre le contexte, pour chaque cas industriel, de la fonction "cataloguer". Nous avons fait cela pour toutes les fonctionnalités, le détail est donnée dans les Annexes. Le contexte formel nous a permis d'obtenir des treillis de concept, en appliquant l'AFC.

Dans la figure 6.9 nous illustrons un treillis de concept pour la fonctionnalité **Securiser**, avec son contexte formel associé.

Nous avons à notre disposition trois types d'algorithmes pour sous-structurer des treillis, chacun apportant des informations différentes et complémentaires. Cette étape étant automatisée (outil RCA),

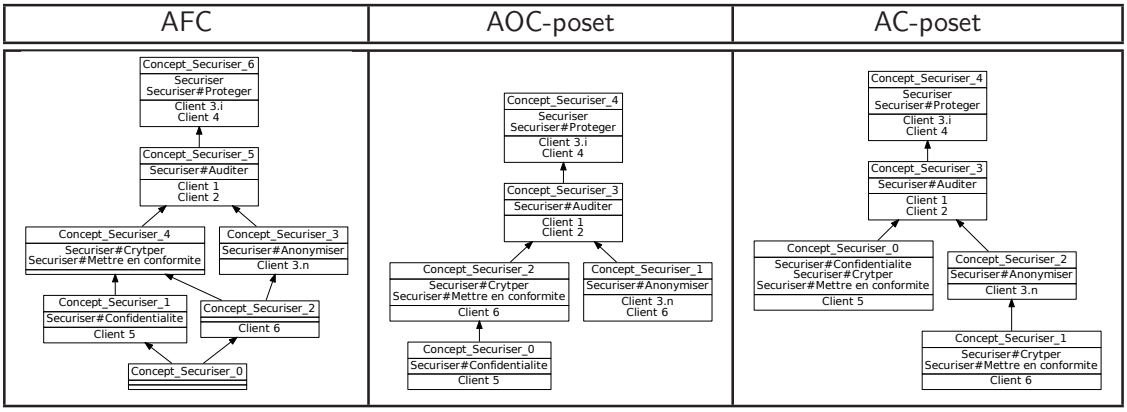


FIGURE *6.10: Comparaison de trois formats de treillis sur la fonction Sécuriser*

nous avons donc généré les trois treillis sous deux formes chacun, une forme simplifiée et une forme complète. Nous avons donc obtenu six types de treillis par fonctionnalité : AFC, Ac-Poset et AOC-poset, illustrés dans la figure 6.10.

La figure 6.10 illustre la génération de trois treillis avec ces trois versions. En annexes, nous insérons les différents treillis pour chaque fonctionnalité du lac de données.

Nous avons appliqué à l'association contexte formel et treillis, le processus de génération automatique (créé par André Mirallés, à base de l'outil E.T.L Talend) qui permet de générer l'ECFD pour chaque fonctionnalité du lac de données (voir figure 6.4).
Dans la figure 6.7, nous illustrons l'obtention d'un ECFD pour la fonctionnalité **Cataloguer**.
Nous avons ensuite, de façon manuelle, obtenu le Feature Model (FM) (voir figure 6.1 pour la fonction **Cataloguer**). En effet cette opération ne peut pas être automatisée à ce jour car elle demande l'intervention d'un expert qui va vérifier la cohérence du résultat obtenus. Les règles expertes ne sont pas pour l'instant formalisées.

Au travers cette méthodologie semi-automatisée, qui ré-utilise des outils déjà existants pour des travaux de recherche sur les lignes de produits, nous avons pu obtenir, à partir de notre base de connaissances un "Feature Model" (FM) pour chacune des fonctionnalités du lac de données.

6.7 L'analyse des premiers résultats

Nous avons focalisé nos travaux sur l'établissement et la validation de la démarche de formalisation des lacs de données par l'approche ligne de produit, et non sur l'analyse poussée de tous les éléments que nous avons obtenu à toutes les étapes (Base de connaissance, Contexte formel, treillis, ECFD, treillis), ceux ci étant extrêmement nombreux et fournis en information.
Nous avons cependant au travers d'une "macro" analyse générale de tous les FM et treillis pu avoir une indication sur la maturité de mise en place des cas industriels inclus dans notre base de connaissance, les uns par rapport aux autres, ainsi que par rapport à notre référentiel expérimental.
Ces analyses nous ont permis de créer un questionnaire de maturité des lacs de données ainsi qu'une prévention au risque de transformation en marécage des lacs de données, si certaines tâches de certaines fonctionnalités n'étaient pas présentes.
Extraits des questions, ayant trait à la fonctionnalité acquérir :

- Les données-sources de données du lac sont-elles listées ?
- Avec quel outil ? Sous quelle forme (base de données, catalogue de métadonnées. . .) ?
- Les données-sources de données du lac sont-elles typées (sensibilité en particulier) ?
- Les données-sources de données du lac sont-elles référencées ? localisées ? personnalisées ?
- Y a-t-il un outil, une base de gestion des données référentielles (Master data) dans le lac ou en dehors du lac ?
- Y a-t-il un responsable (data stewardship) assigné aux données/sources ?
- Y a-t-il un responsable de production des données/sources qui est renseigné ?
- Les données-sources de données du lac sont-elles indexées, "taggées" ?
- Les données-sources de données du lac sont-elles facettées ? c'est à dire arrivent-elles déjà avec un "point de vue" ?
- Les données-sources de données du lac sont-elles qualifiées (Qualité) ?
- Les données-sources de données du lac sont-elles tracées (lineage) ?
- Y a-t-il un outil, une base de gestion de métadonnées du lac ?

Nous avons pu appliquer le questionnaire obtenu sur deux cas industriels supplémentaires qui vont donc enrichir notre base de connaissance et améliorer nos analyses futures.

6.8 Synthèse du chapitre 6

Dans ce chapitre, nous avons expérimenté une approche de modélisation qui s'affranchit des technologies, en vue de produire un modèle pour aider et accélérer la création des lacs de données. Pour cela, nous nous sommes inspirés d'une approche ligne de produits.

Afin de tester cette approche nous avons constitué un référentiel des fonctionnalités d'un lac de données et créé une base de connaissance des lacs de données, basée sur nos collaborations industrielles. A partir de cette base de connaissance, nous avons pu ré-utliser des processus semi-automatiques pour générer un Feature Model (FM) qui donne une ébauche de modèle formalisant les fonctionnalités des composants à mettre en œuvre dans l'architecture des lacs de données.

Nous avons fait une analyse macroscopique des premiers résultats, qui nous a permis de proposer un questionnaire pour évaluer la maturité des lacs de données et prévenir du potentiel de transformation en "marécage".

Ces premières analyses nous confirment dans l'approche proposée, même si cette approche reste encore à affiner et les résultats à analyser plus finement, en collaboration avec des experts de l'approche ligne de produits.

Les premières pistes d'application des modèles FM aux lacs de données ouvrent des perspectives intéressantes qui seront poursuivies dans les travaux futurs.

Chapitre 7

Conclusion et perspectives

7.1 Conclusions

Dans nos travaux, nous nous sommes intéressés à l'évolution du système d'information dans le cadre des données massives et l'exploitation de celles-ci.
Nous avons étudié en détails l'évolution des systèmes décisionnels durant ces vingt dernières années, plus spécifiquement les facteurs qui ont ou qui vont influencer les systèmes décisionnels et leurs impacts sur l'architecture du système d'information.
Des facteurs tels que le format des données, leur volumétrie, les usages mais aussi les logiciels, les infrastructures et l'apparition de la technologie Apache Hadoop.
Lors de cette étude nous avons mis en évidence l'apparition d'un nouveau concept dans le système informatique des organisations : les lacs de données.

Nous avons focalisé notre travail sur l'étude de ce nouveau concept, les lacs de données dans le cadre du phénomène des données massives et avons proposé de le considérer comme un nouveau composant du système d'information, au même titre que les systèmes décisionnels.

Nous avons fait un état des lieux des connaissances, plus qu'un état de l'art sur les lacs de données, car ce sujet est encore très récent que ce soit dans la littérature scientifique ou dans le monde industriel. Au regard de cet état des lieux et des enjeux que représentent les lacs de données pour les organisations, nous avons proposé de les positionner dans le système d'information, en tant que nouvel composant. Nous avons ensuite clairement établi la complémentarité de ce composant vis-à-vis du système décisionnel, autre composant du SI et avons proposé notre définition pour les lacs de données qui est la suivante :
Le lac de données est une collection de données (/ ou un ensemble de donnée) qui est :

- **Indépendante d'un schéma d'information pré établi ;**
- **De formats non contraints (tous formats acceptés) ;**
- **Non transformée ;**

- **Conceptuellement rassemblée en un endroit unique mais potentiellement non matérialisée ;**
- **Destinée à un ou des utilisateurs experts en science des données ;**
- **Munie d'un catalogue de méta-données ;**
- **Munie d'un ensemble de règles et méthodes de gouvernance de données.**

L'objectif premier du lac de données est de permettre l'exploration, sans a priori, des données qu'il rassemble, en vue de découvrir des nouvelles pistes d'information à exploiter dans le contexte d'une valorisation des données d'une organisation. C'est un système dirigé par les données *data driven* qui vient compléter le système décisionnel *information driven* mis en place.

Nous avons ensuite donné notre point de vue sur la vision actuelle de l'architecture des lacs de données, fortement associée à la technologie Apache Hadoop, qui se traduit par une architecture mono technologique. Nous avons montré les limites de cette approche et avons proposé d'introduire plus d'hybridation dans les choix technologiques, tels que les bases NoSQL, les bases relationnelles, qui peuvent compléter les architectures des lacs de données.

Pour appuyer ces scénarios "hybrides", et remettre en cause la vision de l'architecture mono technologique des lacs de données, nous avons étudié un facteur d'influence sur la conception des architectures des lacs de données : la gravité des données. Après avoir décrit les éléments qui composent la gravité des données, tels que le volume, la sensibilité et le coût de déplacement des données nous avons démontré l'impact de la prise en compte de ce facteur sur l'architecture des lacs de données, au travers d'un cas d'usage industriel.

En vue d'amorcer une formalisation des lacs de données, nous avons exploré une démarche lignes de produits logiciel pour proposer une ébauche d'approche conceptuelle des lacs de données. Pour cela, nous avons constitué une base de connaissance des lacs de données provenant de nos collaborations avec le monde industriel.
Cette base de connaissance continue de s'enrichir et va continuer à d'être exploitée, en collaboration avec des experts sur le domaine de l'approche ligne de produits en vue d'analyser plus finement les premiers résultats que nous avons obtenu (36 treillis ont déjà été produits).
Loin de se terminer par l'achèvement de ce manuscrit, nos travaux vont se poursuivre pour approfondir et améliorer cette approche. La formalisation sous forme de Feature Model (modèle de ligne de produits) des fonctionnalités des lacs de données constitue une aide précieuse pour l'architecte d'information.
Dans la prochaine, et dernière section de ce mémoire nous présentons quelques perspectives à nos travaux.

7.2 Perspectives

Suite aux travaux présentés, plusieurs perspectives intéressantes s'ouvrent à nous.

— Lors de notre étude de l'influence de la gravité des données sur la relation donnée-traitement nous avons pris position pour un accès en mode fédération des données ne pouvant se déplacer dans le lac de données, cela suppose que les systèmes qui hébergent les données voulant être accédées acceptent l'exécution du traitement sur leur environnement. L'apport d'accélérateur d'exécution de ces traitements afin de diminuer l'impact sur le système source est à explorer.

— Une amélioration des outils d'accès en fédération et leur intégration au niveau du catalogue des métadonnées, par exemple, pourrait rendre plus transparent ce mode d'accès.

— les métadonnées n'ont pas été intrinsèquement traitées, nous pensons que l'impact des terminologies et ontologies pourraient aussi être investiguées.

— Les perspectives les plus ouvertes concernent le travail que nous avons amorcé sur la formalisation des lacs de données via une approche de ligne de produits logiciels. Avec dans un premier temps l'exploitation d'une partie de notre base de connaissance via les composants techniques. Pour la validation de notre approche nous n'avons exploité que la partie fonctionnalités. De même l'apport de moteur d'extraction de règle, autour des associations triadiques pourra donner des analyses différentes sur cette base de connaissance.

Enfin, en adoptant la démarche du biomimétisme (Du grec, Bio : vie et Mimesis : imiter), nous pourrions analyser le composant lac de données via la vision des lacs naturels notamment lors du phénomène d'eutrophisation. "L'eutrophisation des milieux aquatiques est un déséquilibre du milieu provoqué par l'augmentation de la concentration d'azote et de phosphore dans le milieu. Elle est caractérisée par une croissance excessive des plantes et des algues due à la forte disponibilité des nutriments". (Wikipedia). Quelle signification aurait ce phénomène dans le contexte des architectures des systèmes d'information ?

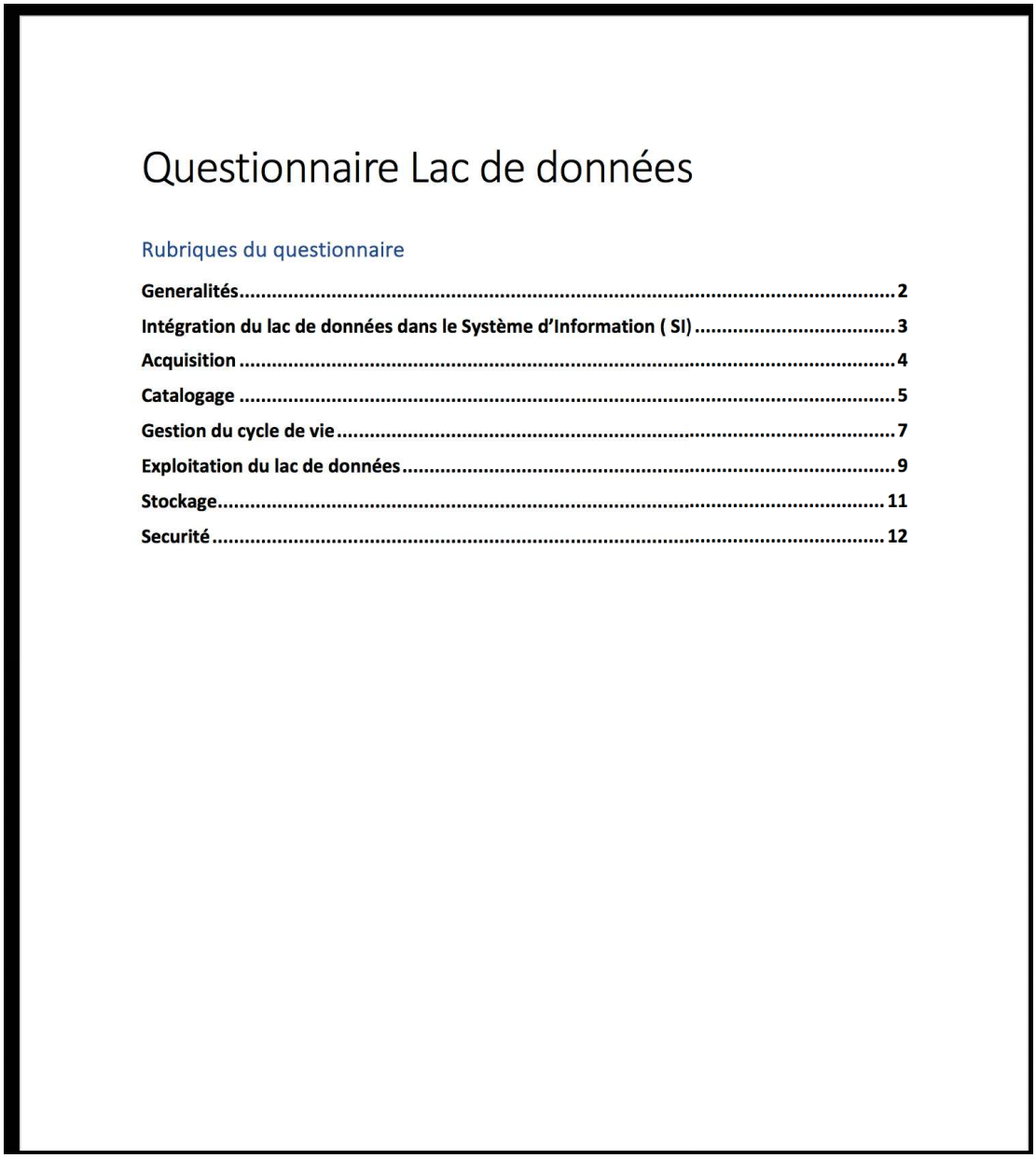
Les perspectives sont donc nombreuses et présentées ici de manière non exhaustive sur un sujet encore très récent où beaucoup d'explorations sont encore à faire et pour lesquelles nous allons être pro actifs.

Ce manuscrit n'est pas une fin, mais un préambule à la poursuite de nos recherches sur les évolutions des systèmes et architectures d'information.

Chapitre 8

Annexes

8.1 Questionnaire lac de données

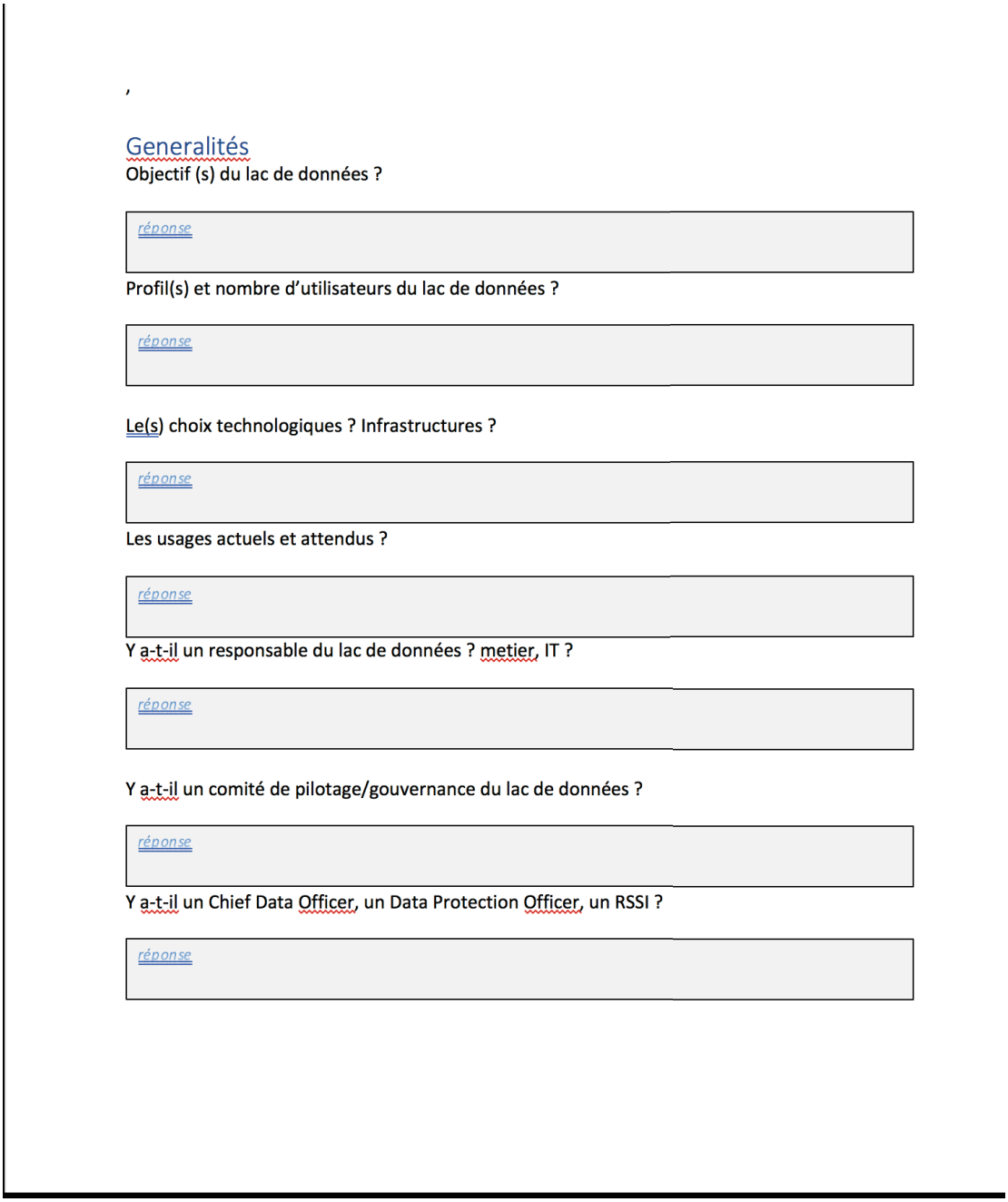


Questionnaire Lac de données

Rubriques du questionnaire

Generalités **2**
Intégration du lac de données dans le Système d'Information (SI) **3**
Acquisition **4**
Catalogage **5**
Gestion du cycle de vie **7**
Exploitation du lac de données **9**
Stockage **11**
Securité **12**

FIGURE *8.1: Sommaire du questionnaire lac de données*



,

Generalités

Objectif (s) du lac de données ?

réponse

Profil(s) et nombre d'utilisateurs du lac de données ?

réponse

Le(s) choix technologiques ? Infrastructures ?

réponse

Les usages actuels et attendus ?

réponse

Y a-t-il un responsable du lac de données ? metier, IT ?

réponse

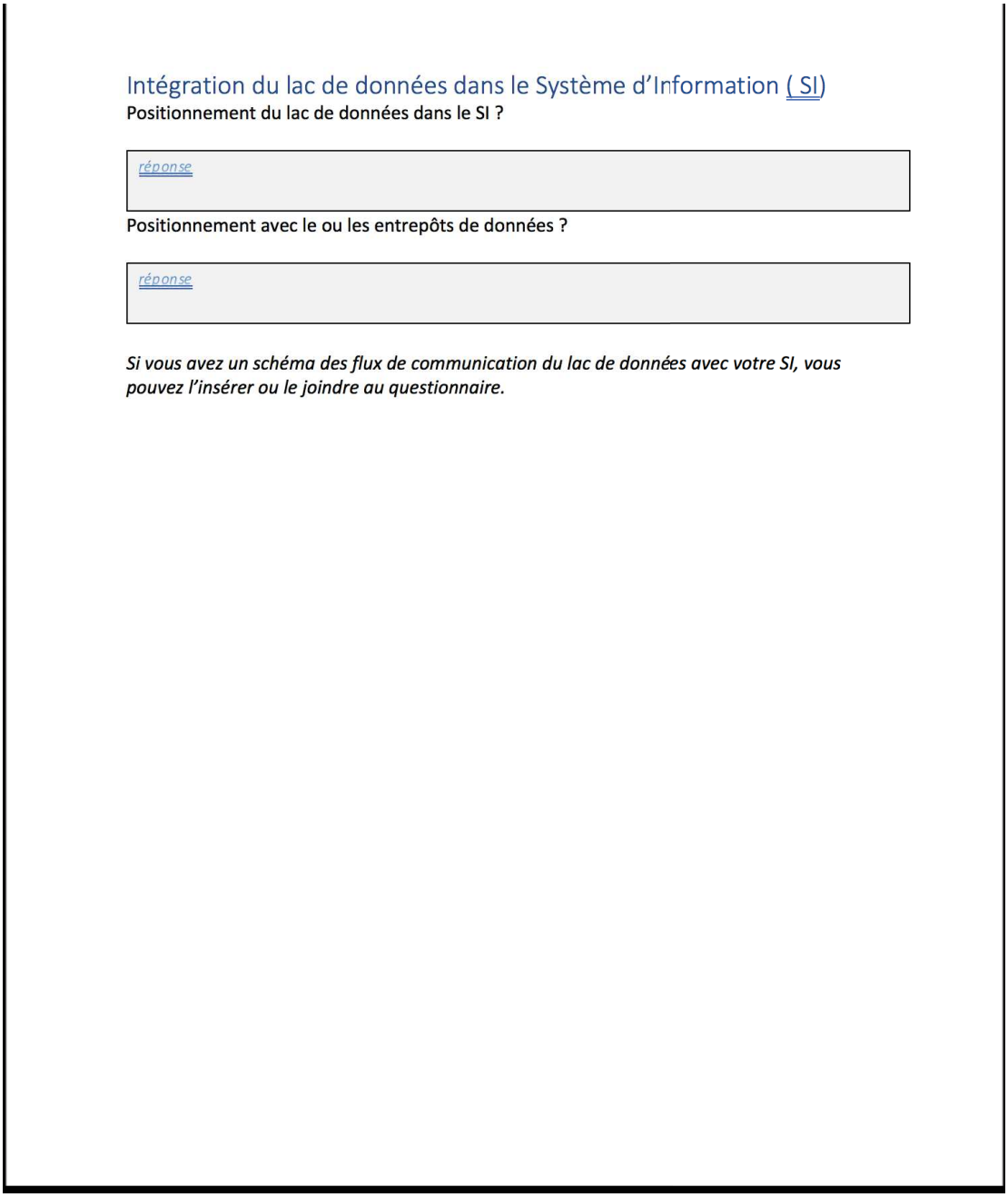
Y a-t-il un comité de pilotage/gouvernance du lac de données ?

réponse

Y a-t-il un Chief Data Officer, un Data Protection Officer, un RSSI ?

réponse

FIGURE 8.2: *Généralités 1/2 du questionnaire lac de données*



Intégration du lac de données dans le Système d'Information (SI)

Positionnement du lac de données dans le SI ?

réponse

Positionnement avec le ou les entrepôts de données ?

réponse

Si vous avez un schéma des flux de communication du lac de données avec votre SI, vous pouvez l'insérer ou le joindre au questionnaire.

FIGURE *8.3: Généralités 2/2 du questionnaire lac de données*

Acquisition

Types de données dans le lac ? structurées, non structurées, interne, externes ?

réponse

Granularité des sources de données ? brute (primaire), agrégées, résumées, facettées ?

réponse

Quels types de modes d'acquisition : Temps réel, temps différé (batch) ?

réponse

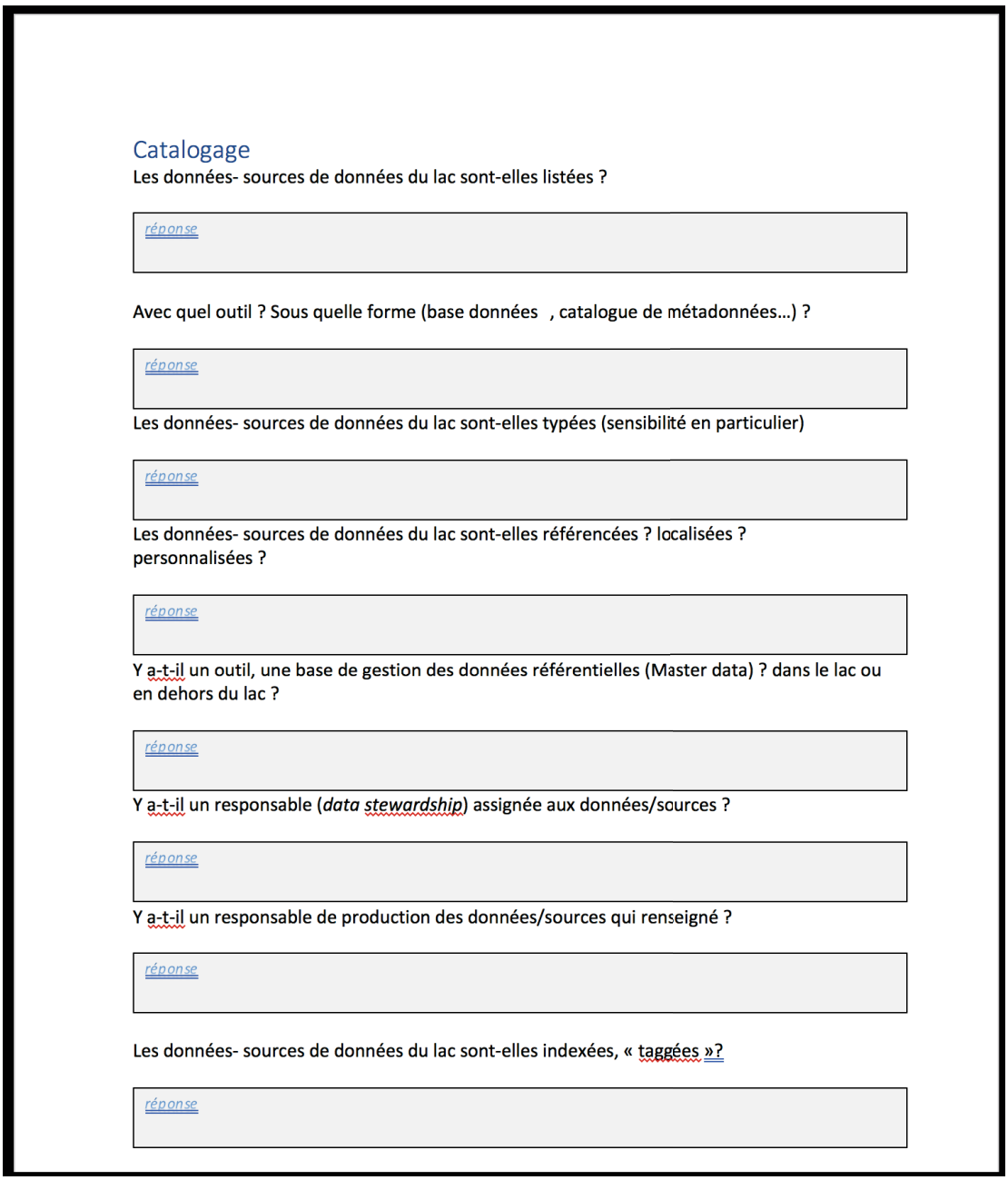
Quels types d'acquisition de données ? Extraction, Duplication, connexion (abonnement à des flux) ? mode push ou/et pull ? virtualisation (les données ne sont pas physiquement déplacées dans le lac)

réponse

Quels outils/ langage(s) pour faire l'acquisition des données dans le lac ?

réponse

FIGURE *8.4: Fonctionnalité Acquisition- questionnaire lac de données*



Catalogage

Les données- sources de données du lac sont-elles listées ?

réponse

Avec quel outil ? Sous quelle forme (base données , catalogue de métadonnées…) ?

réponse

Les données- sources de données du lac sont-elles typées (sensibilité en particulier)

réponse

Les données- sources de données du lac sont-elles référencées ? localisées ? personnalisées ?

réponse

Y a-t-il un outil, une base de gestion des données référentielles (Master data) ? dans le lac ou en dehors du lac ?

réponse

Y a-t-il un responsable (*data stewardship*) assignée aux données/sources ?

réponse

Y a-t-il un responsable de production des données/sources qui renseigné ?

réponse

Les données- sources de données du lac sont-elles indexées, « taggées »?

réponse

FIGURE *8.5: Fonctionnalité Catalogage 1/2- questionnaire lac de données*

Les données- sources de données du lac sont-elles facettées ? *cad arrivent –elles déjà avec un « point de vue ».*

réponse

Les données- sources de données du lac sont-elles qualifiées (Qualité) ?

réponse

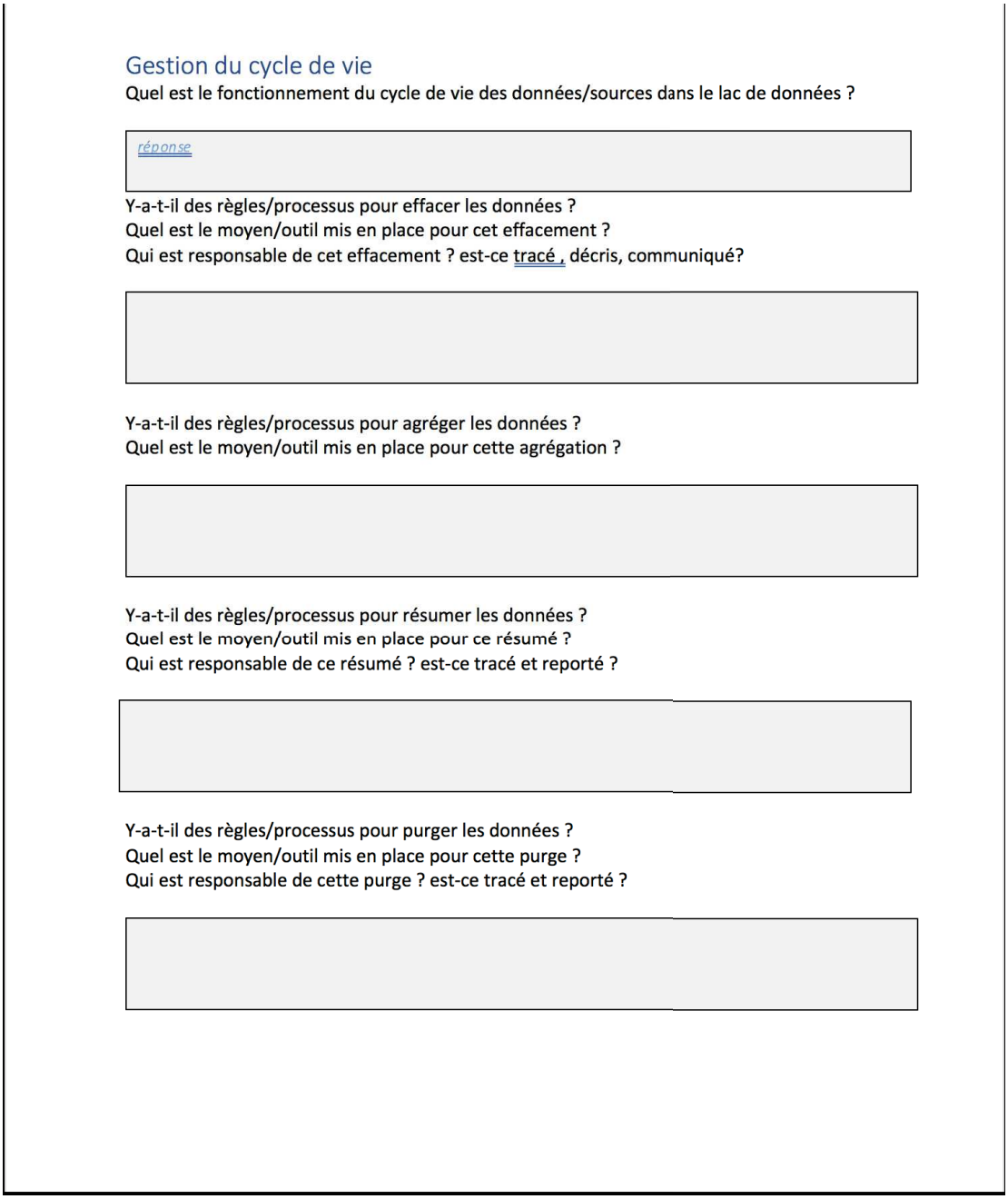
Les données- sources de données du lac sont-elles tracées (lineage) ?

Réponse

Y a-t-il un outil, une base de gestion de métadonnées du lac ?

réponse

FIGURE *8.6: Fonctionnalité Catalogage 2/2- questionnaire lac de données*



Gestion du cycle de vie

Quel est le fonctionnement du cycle de vie des données/sources dans le lac de données ?

réponse

Y-a-t-il des règles/processus pour effacer les données ?
Quel est le moyen/outil mis en place pour cet effacement ?
Qui est responsable de cet effacement ? est-ce tracé , décris, communiqué?

Y-a-t-il des règles/processus pour agréger les données ?
Quel est le moyen/outil mis en place pour cette agrégation ?

Y-a-t-il des règles/processus pour résumer les données ?
Quel est le moyen/outil mis en place pour ce résumé ?
Qui est responsable de ce résumé ? est-ce tracé et reporté ?

Y-a-t-il des règles/processus pour purger les données ?
Quel est le moyen/outil mis en place pour cette purge ?
Qui est responsable de cette purge ? est-ce tracé et reporté ?

FIGURE *8.7: Fonctionnalité du Cycle de vie 1/2- questionnaire lac de données*

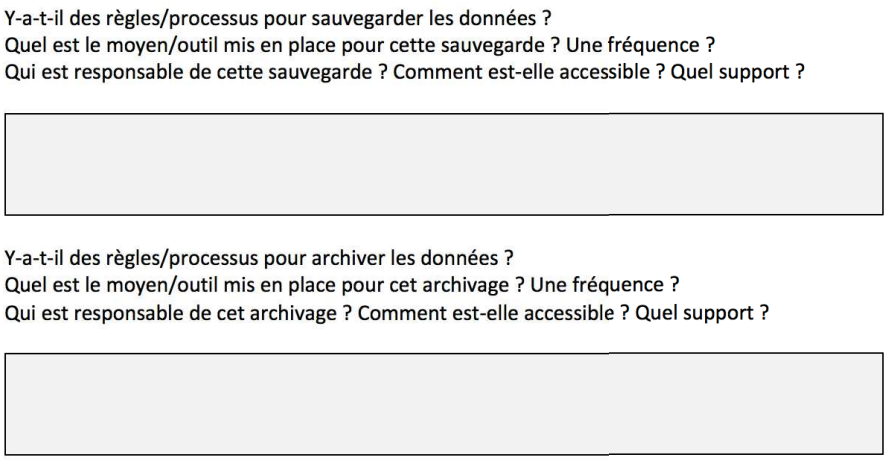
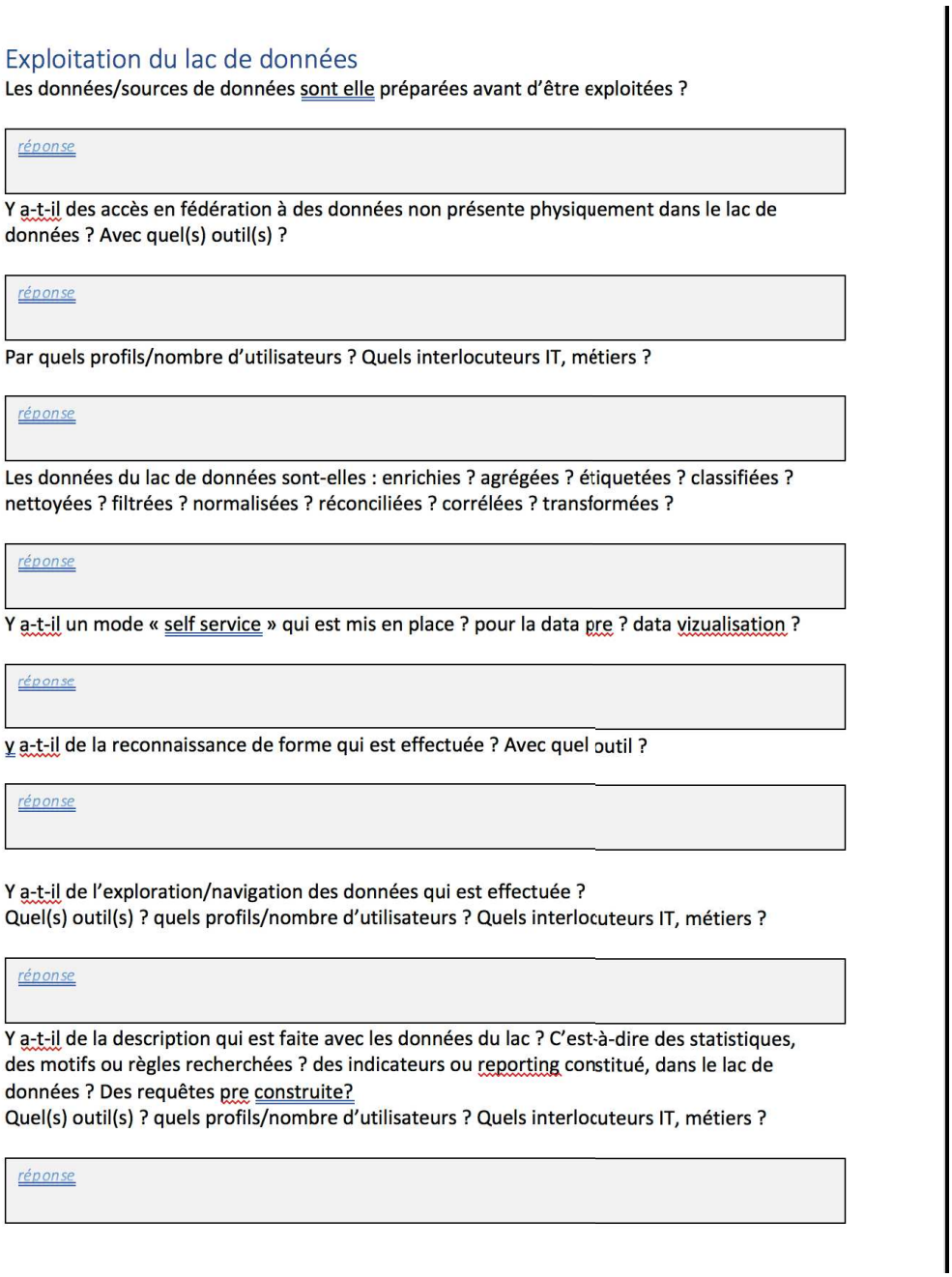


FIGURE *8.8: Fonctionnalité du Cycle de vie 2/2- questionnaire lac de données*



Exploitation du lac de données

Les données/sources de données sont elle préparées avant d'être exploitées ?

réponse

Y a-t-il des accès en fédération à des données non présente physiquement dans le lac de données ? Avec quel(s) outil(s) ?

réponse

Par quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Les données du lac de données sont-elles : enrichies ? agrégées ? étiquetées ? classifiées ? nettoyées ? filtrées ? normalisées ? réconciliées ? corrélées ? transformées ?

réponse

Y a-t-il un mode « self service » qui est mis en place ? pour la data pre ? data vizualisation ?

réponse

y a-t-il de la reconnaissance de forme qui est effectuée ? Avec quel outil ?

réponse

Y a-t-il de l'exploration/navigation des données qui est effectuée ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Y a-t-il de la description qui est faite avec les données du lac ? C'est-à-dire des statistiques, des motifs ou règles recherchées ? des indicateurs ou reporting constitué, dans le lac de données ? Des requêtes pre construite?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

FIGURE *8.9: Fonctionnalité Exploitation 1/2- questionnaire lac de données*

Y a-t-il de la segmentation avec les données du lac qui est effectuée ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Y a-t-il de la prédiction avec les données qui est effectuée ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Y a-t-il de la prescription avec les données qui est effectuée ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Y a-t-il mise en place de système expert ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Y a-t-il des accès par des API sur le lac de données ? Lesquelles ?

réponse

Y a-t-il publication des résultats des explorations/analyses faites dans le lac ? communication mise à disposition de « data set préparés ou résultants » ?

réponse

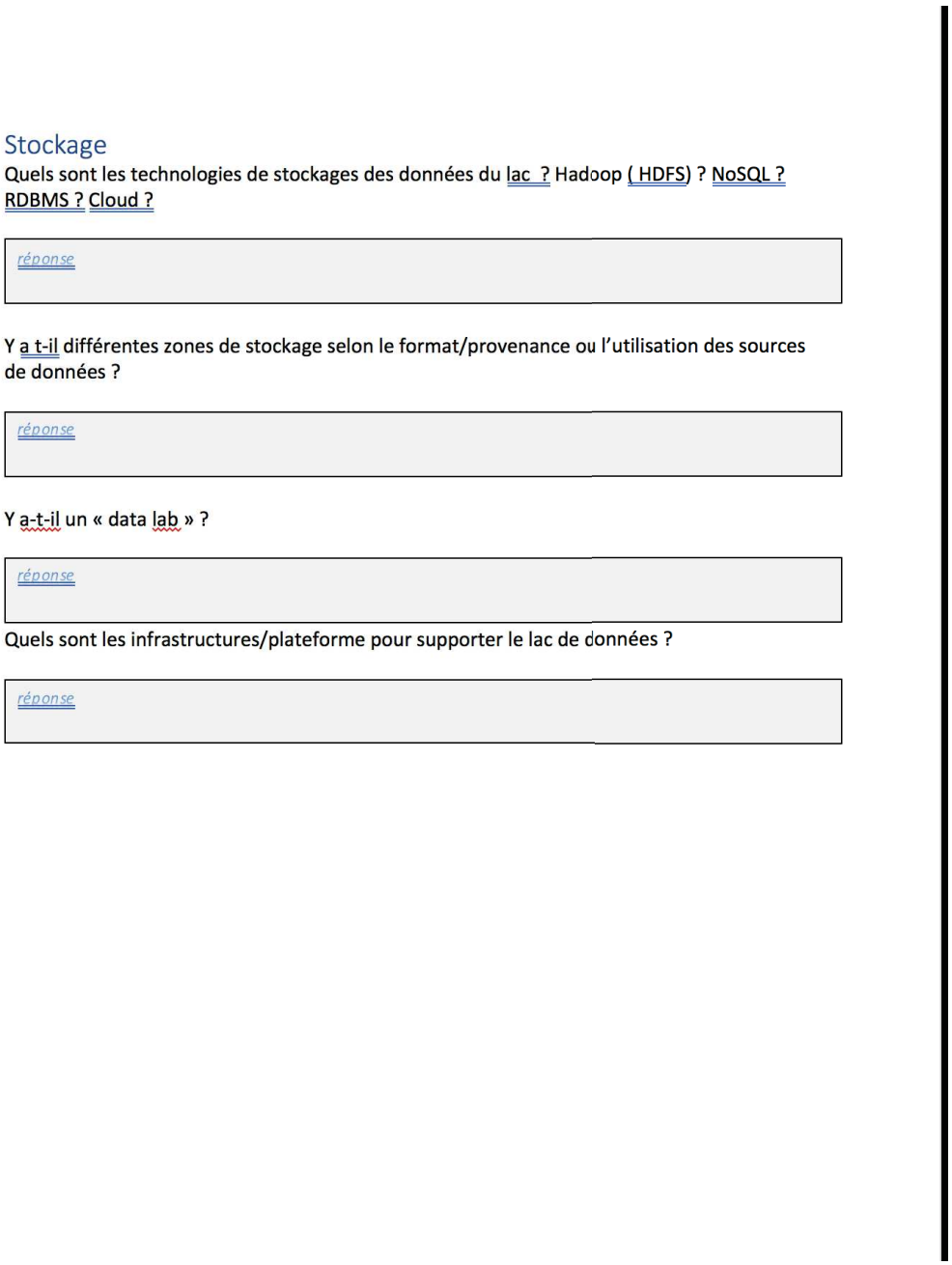
Un mode collaboratif est-il en place ? entre les data engineers, data analysts, data scientistes ?
Quel(s) outil(s) ? quels profils/nombre d'utilisateurs ? Quels interlocuteurs IT, métiers ?

réponse

Comment le lac de données est-il administré ? Rôles ? Profils ? Processus ? outils ?

réponse

FIGURE *8.10: Fonctionnalité Exploitation 2/2- questionnaire lac de données*



Stockage

Quels sont les technologies de stockages des données du lac ? Hadoop (HDFS) ? NoSQL ? RDBMS ? Cloud ?

réponse

Y a t-il différentes zones de stockage selon le format/provenance ou l'utilisation des sources de données ?

réponse

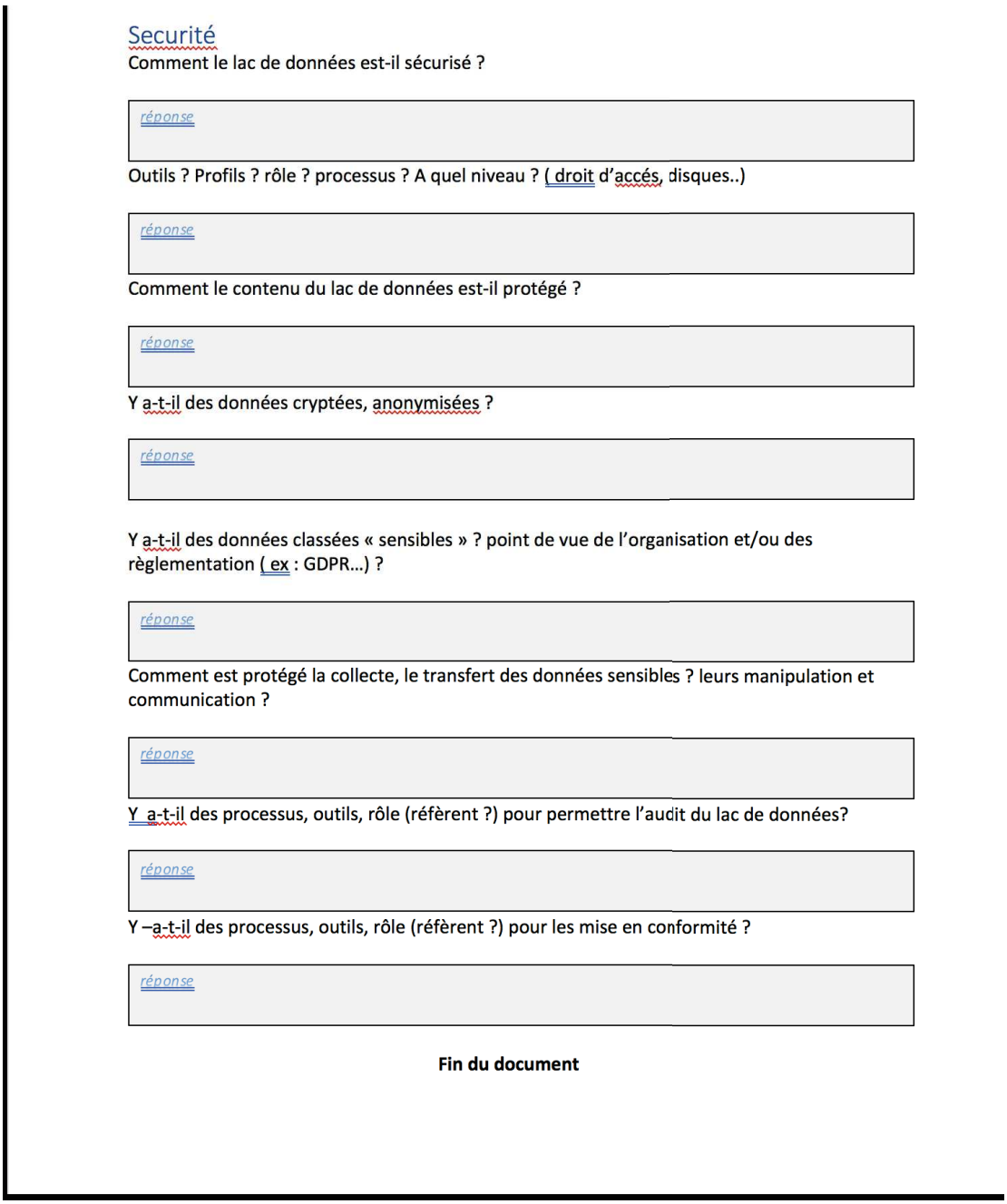
Y a-t-il un « data lab » ?

réponse

Quels sont les infrastructures/plateforme pour supporter le lac de données ?

réponse

FIGURE *8.11: Fonctionnalité Stockage- questionnaire lac de données*



Securité

Comment le lac de données est-il sécurisé ?

réponse

Outils ? Profils ? rôle ? processus ? A quel niveau ? (droit d'accés, disques..)

réponse

Comment le contenu du lac de données est-il protégé ?

réponse

Y a-t-il des données cryptées, anonymisées ?

réponse

Y a-t-il des données classées « sensibles » ? point de vue de l'organisation et/ou des règlementation (ex : GDPR...) ?

réponse

Comment est protégé la collecte, le transfert des données sensibles ? leurs manipulation et communication ?

réponse

Y a-t-il des processus, outils, rôle (réfèrent ?) pour permettre l'audit du lac de données?

réponse

Y –a-t-il des processus, outils, rôle (réfèrent ?) pour les mise en conformité ?

réponse

Fin du document

FIGURE *8.12: Fonctionnalité Securité- questionnaire lac de données*

8.2 Ligne de Produit treillis

8.2.1 Fonction Cataloguer

8.2.1.1 Fonction Cataloguer-concept formel

			maturité	1 an	2 ans	2 ans	3 mois	3 ans	3 ans	3 ans
			évolution			Version initiale	Version courante			
			data lake	DL 1	DL 2	DL 3i	DL 3f	DL 4	DL 5	DL 6
Cataloguer				x	x	x	x	x	x	x
	Lister					x	x	x	x	x
	Typer (sensiblité en particulier)								x	x
	Référencer			x	x				x	x
		Localiser		x						
		Personnaliser		x					x	x
			Responsabiliser						x	x
			Production (Produire)	x					x	
		Indexer			x				x	
	Lister Facettes									
	Qualifier (Qualité)								x	x
	Tracer			x	x					x

FIGURE *8.13: Concept formel-Cataloguer*

8.2.1.2 Fonction Cataloguer-AC-poset

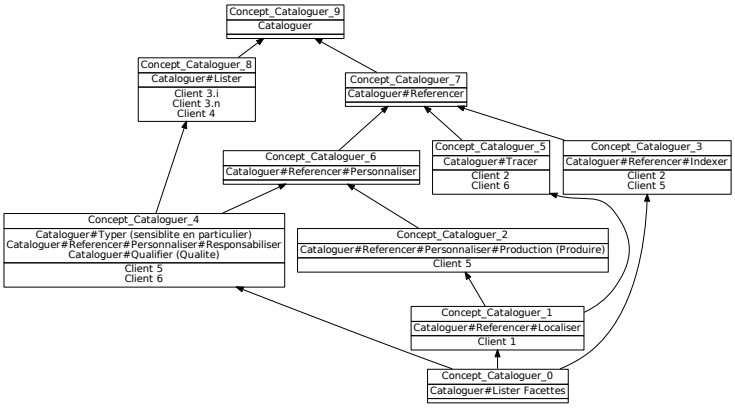


FIGURE *8.14: AC-Cataloguer-simple*

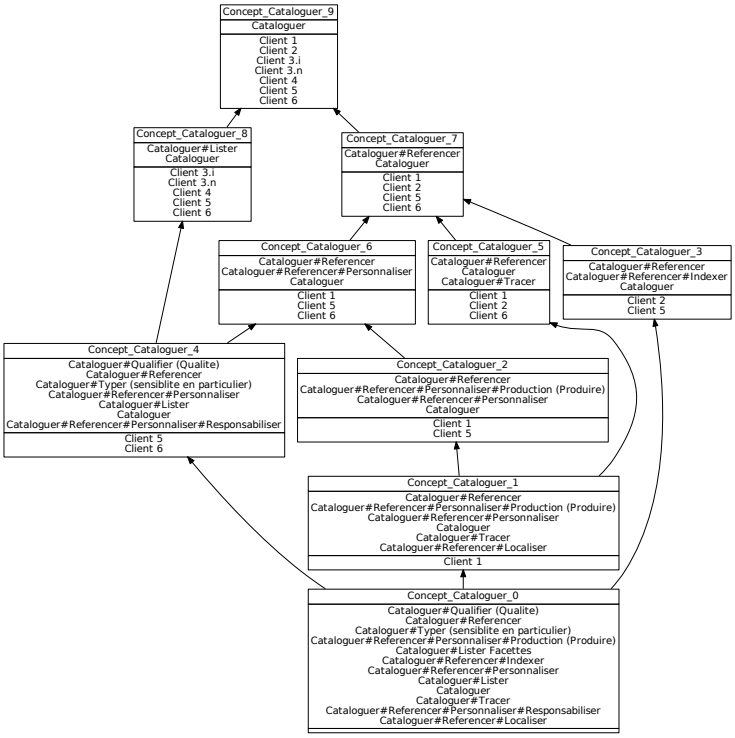


FIGURE *8.15: AC-Cataloguer-plein*

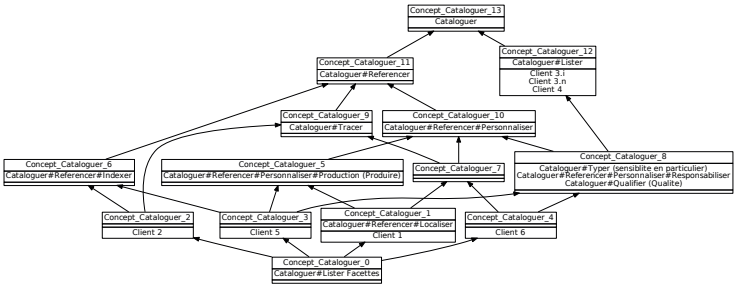


FIGURE *8.16: FCA-Cataloguer-simple*

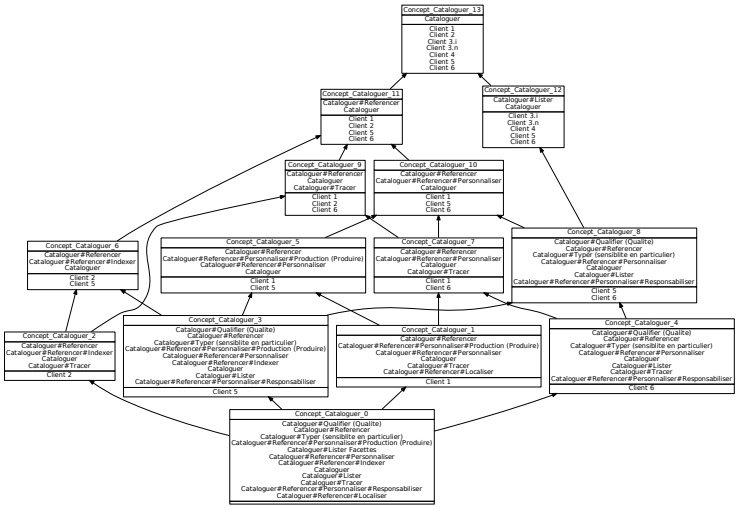


FIGURE *8.17: FCA-Cataloguer-plein*

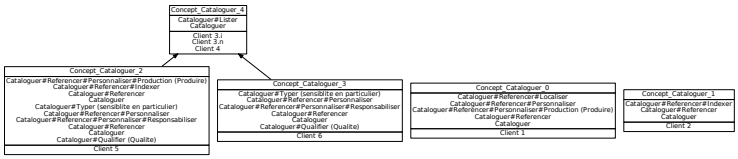


FIGURE *8.18: AOC-Cataloguer-simple*

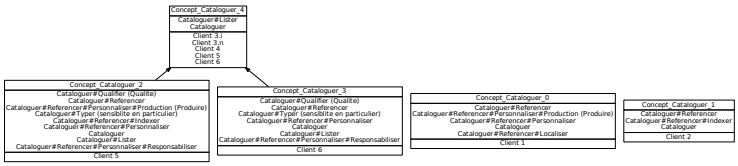


FIGURE *8.19: AOC-Cataloguer-plein*

	maturité	1 an	2 ans	2 ans	3 mois	3 ans	3 ans	3 ans
	évolution			Version initiale	Version courante			
	data lake	DL 1	DL 2	DL 3i	DL 3f	DL 4	DL 5	DL 6
Stocker		x	x	x	x	x		
	Physique	x	x	x	x		x	x
	Virtuel		x				x	x

FIGURE *8.20: Concept formel-Stocker*

8.2.1.3 Fonction Cataloguer-FCA

8.2.1.4 Fonction Cataloguer-AOC-poset

8.2.2 Fonction Stocker

8.2.2.1 Fonction Stocker-concept formel

8.2.2.2 Fonction Stocker-AC-poset

8.2.2.3 Fonction Stocker-FCA

8.2.2.4 Fonction Stocker-AOC-poset

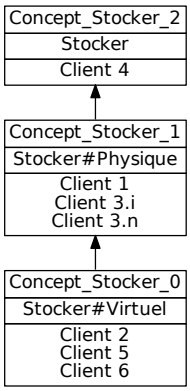


FIGURE *8.21: AC-stocker-simple*

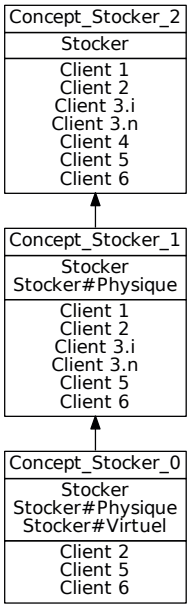


FIGURE *8.22: AC-stocker-plein*

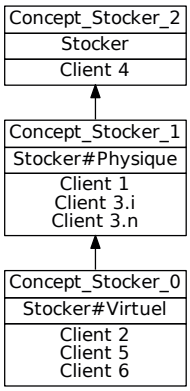


FIGURE *8.23: FCA-stocker-simple*

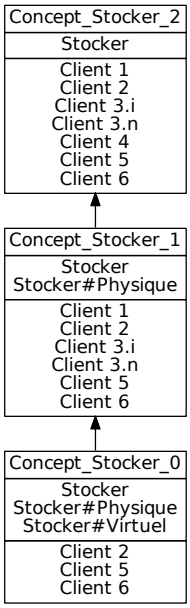


FIGURE *8.24: FCA-stocker-plein*

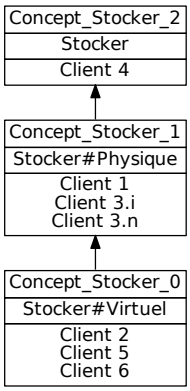


FIGURE *8.25: AOC-stocker-simple*

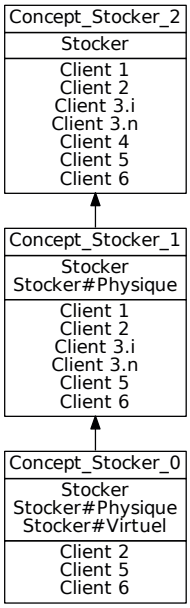


FIGURE *8.26: AOC-stocker-plein*

			maturité	1 an	2 ans	2 ans	3 mois	3 ans	3 ans	3 ans
			évolution			Version initiale	Version courante			
			data lake	DL 1	DL 2	DL 3i	DL 3f	DL 4	DL 5	DL 6
Exploiter										
	Préparer			x	x		x	x	x	x
		Enrichir			x		x	x	x	
		Agréger						x	x	
		Etiqueter				x	x			
		Reconnaissance de forme								
		Classifier						x	x	x
		Nettoyer								
		Filtrer		x	x					
		Reconcilier		x					x	
		Corréler								x
		Transformer							x	x
	Explorer/Produire			x	x			x		
		Naviguer						x		
		Décrire							x	
			Statistiques						x	x
			Motifs ou Régles							
			indicateur/reporting							
		Segmenter		x					x	x
		Prédire		x			x		x	
		Prescrire								
		Inférer (Systèmes Experts)								
		Requêter							x	x
	Publier			x	x	x	x			x
	Administrer			x	x	x	x	x	x	x

FIGURE *8.27: Concept formel-Exploiter*

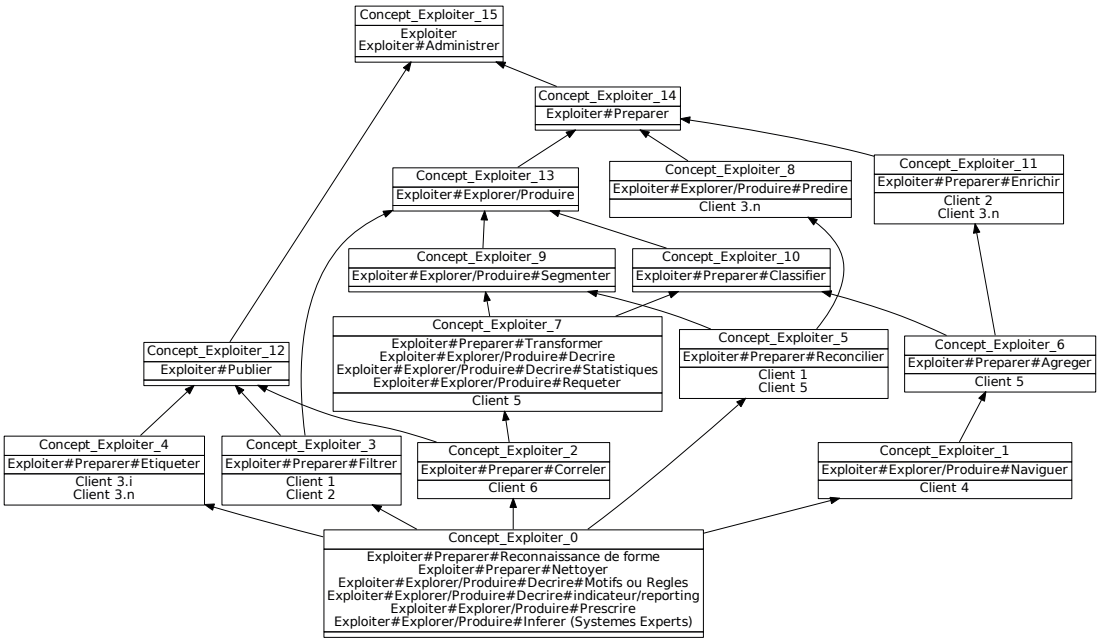


FIGURE *8.28: AC-Exploiter-simple*

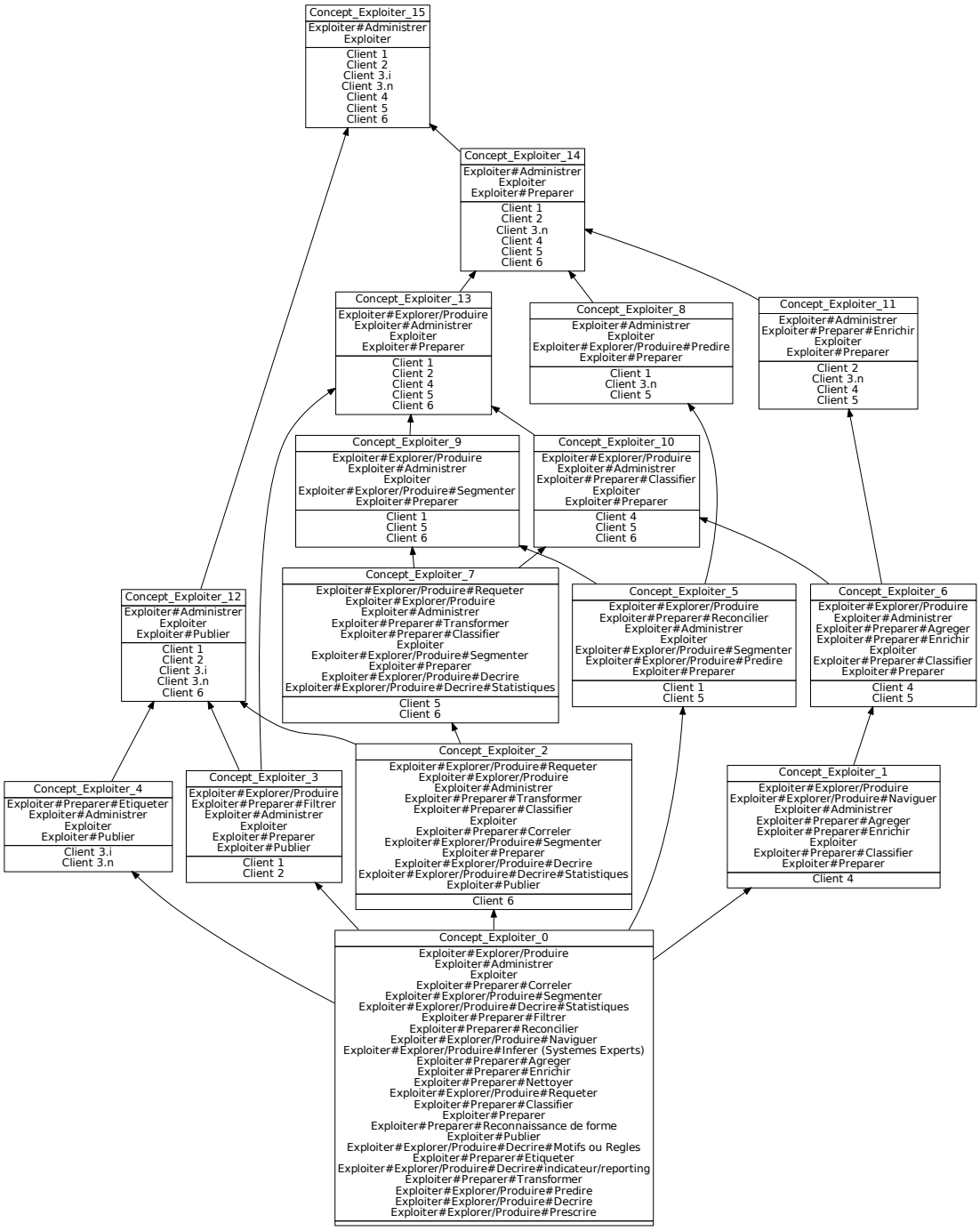


FIGURE *8.29: AC-Exploiter-plein*

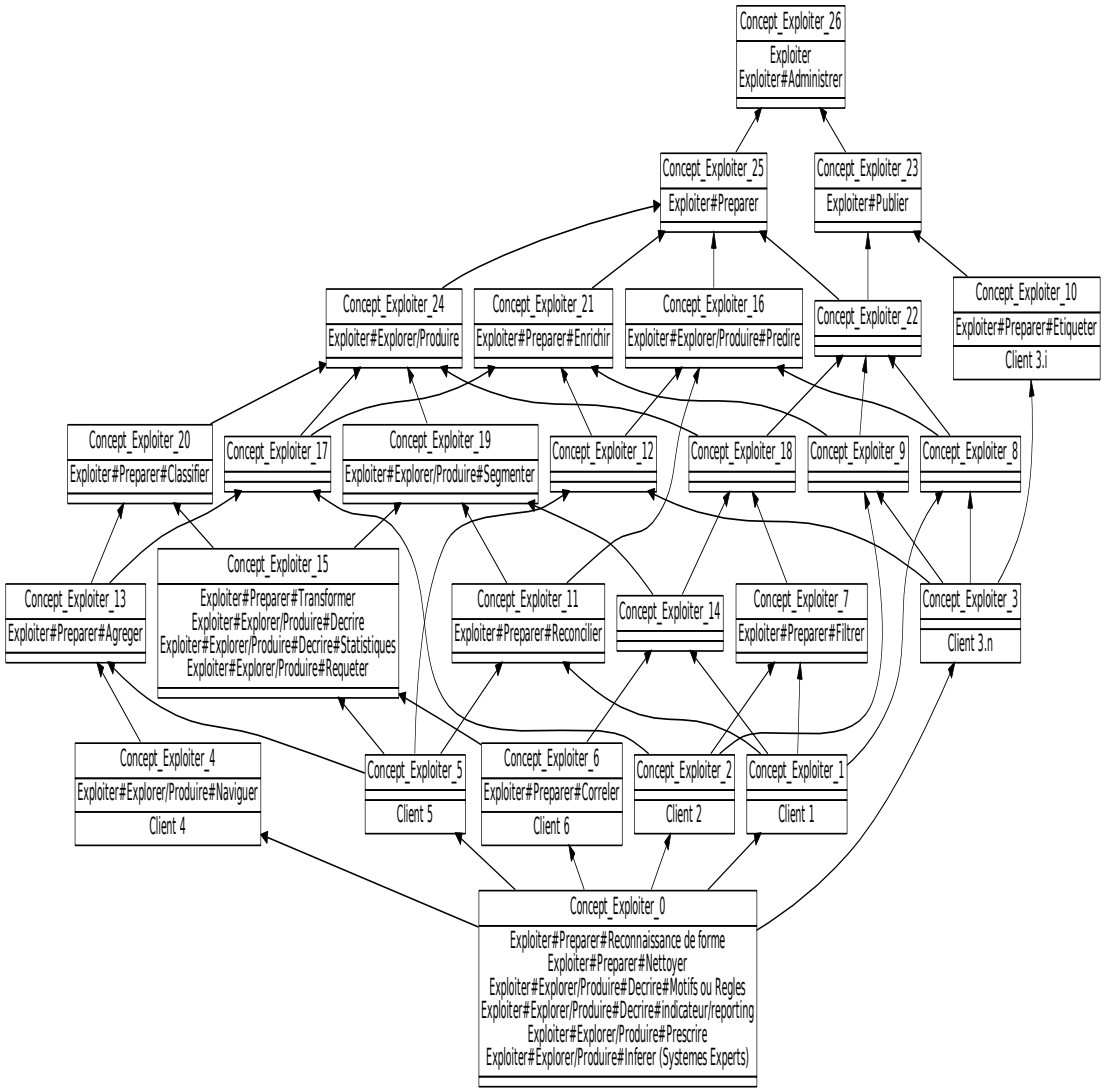


FIGURE *8.30: FCA-Exploiter-simple*

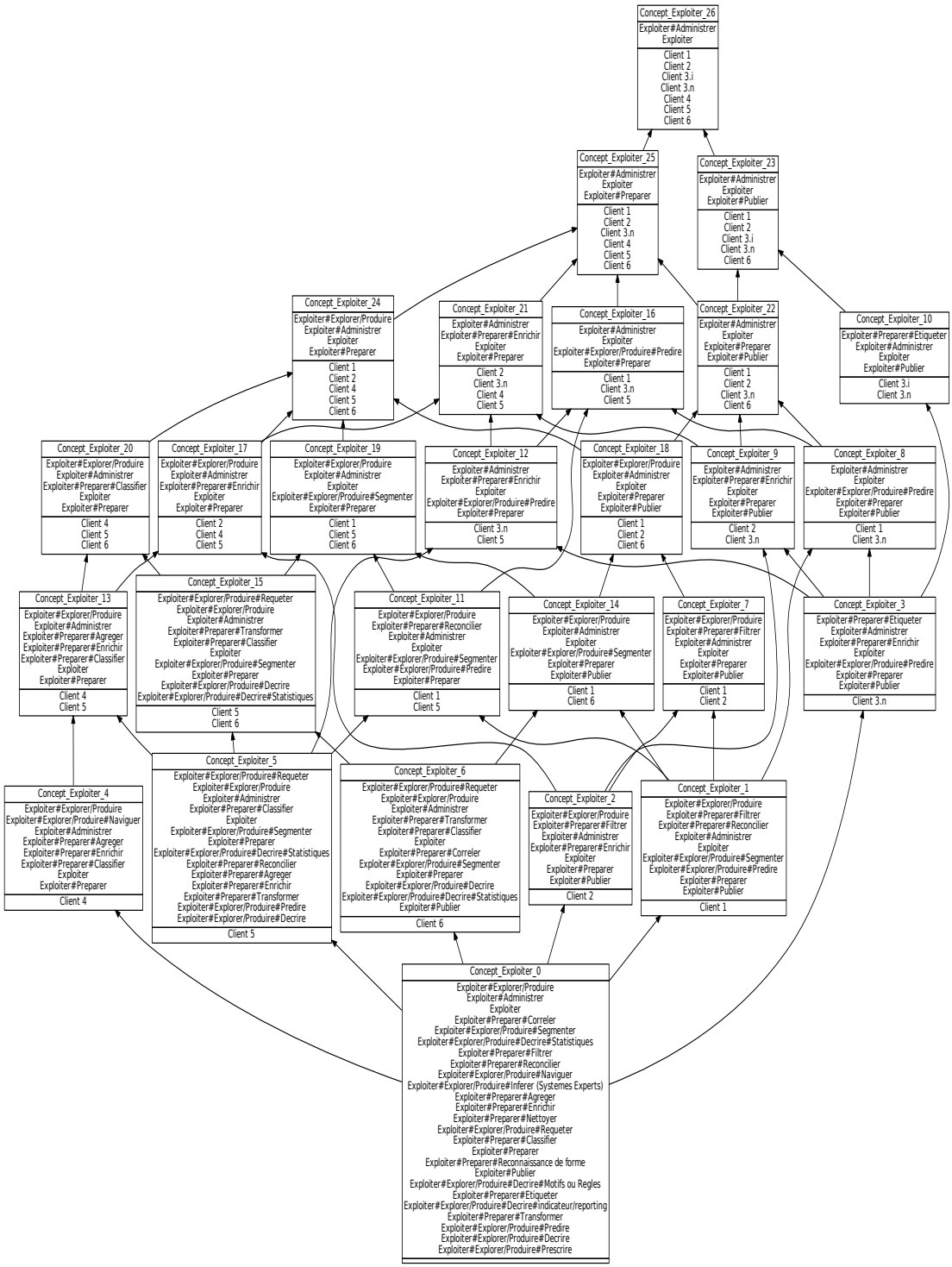


FIGURE *8.31: FCA-Exploiter-plein*

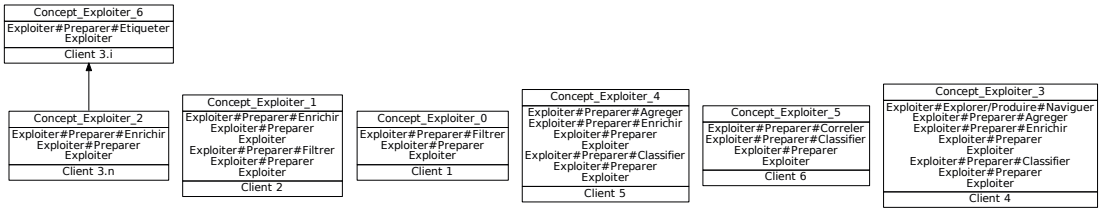


FIGURE *8.32: AOC-Exploiter-simple*

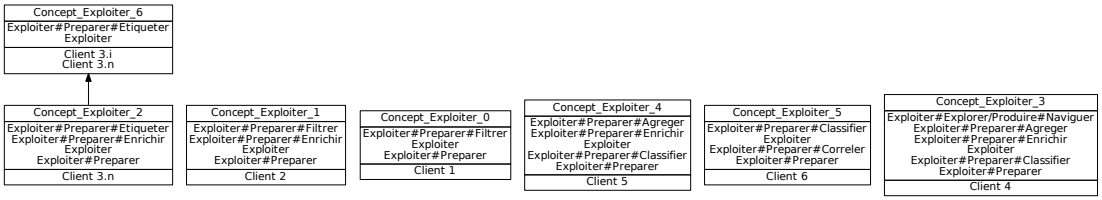


FIGURE *8.33: AOC-Exploiter-plein*

		maturité	1 an	2 ans	2 ans	3 mois	3 ans	3 ans	3 ans
		évolution			Version initiale	Version courante			
Gérer cycle de vie		**data lake**	DL 1	DL 2	DL 3i	DL 3f	DL 4	DL 5	DL 6
	Fonctionner		x	x		x		x	x
		Effacer							
		Agréger	x					x	
		Résumer	x	x		x			x
		Purger (Effacer pour des raisons techniques)						x	
	Sauvegarder		x	x	x	x	x	x	x
	Archiver		x			x		x	

FIGURE *8.34: Concept formel-Gérer*

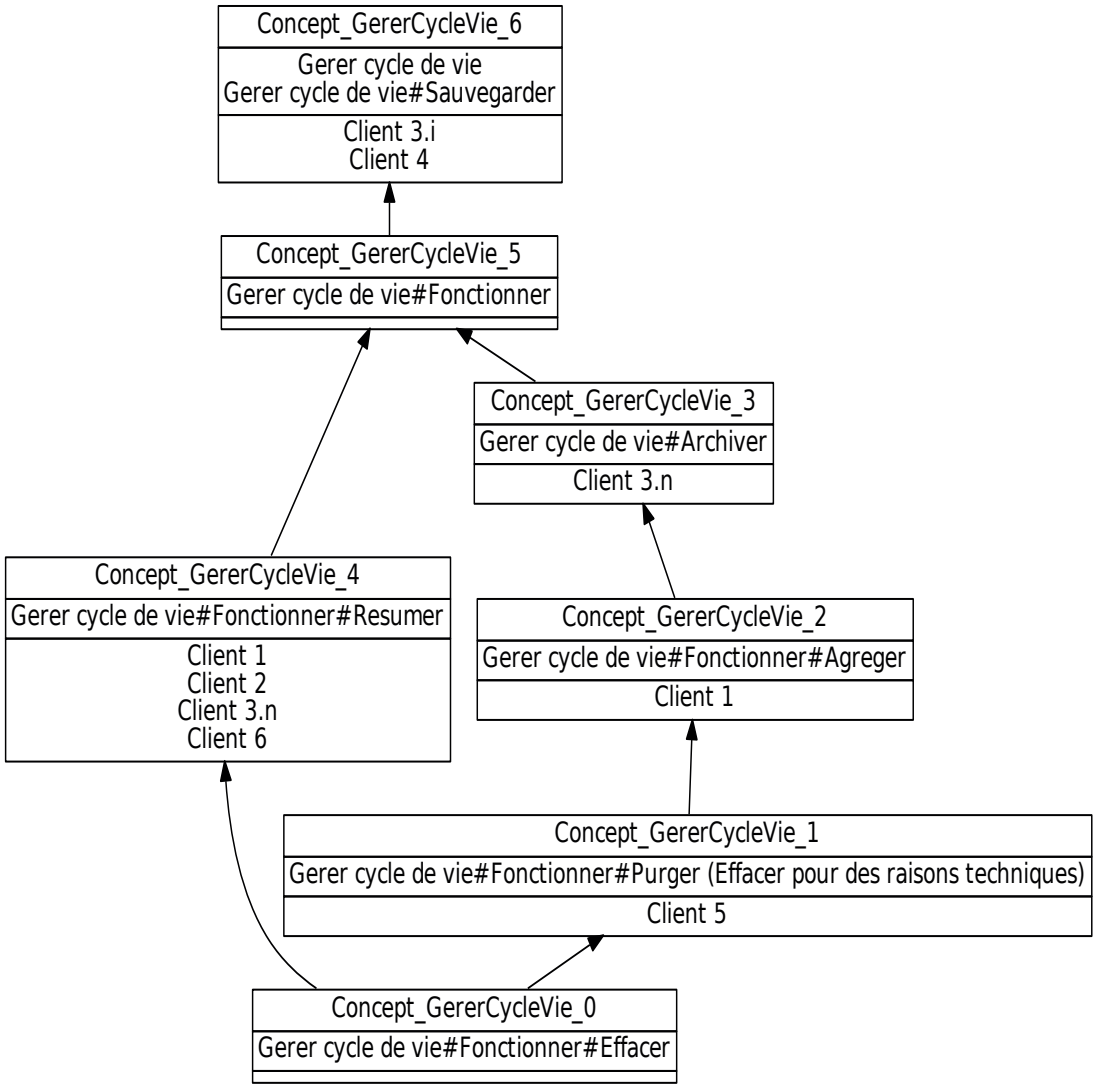


FIGURE *8.35: AC-Gérer-simple*

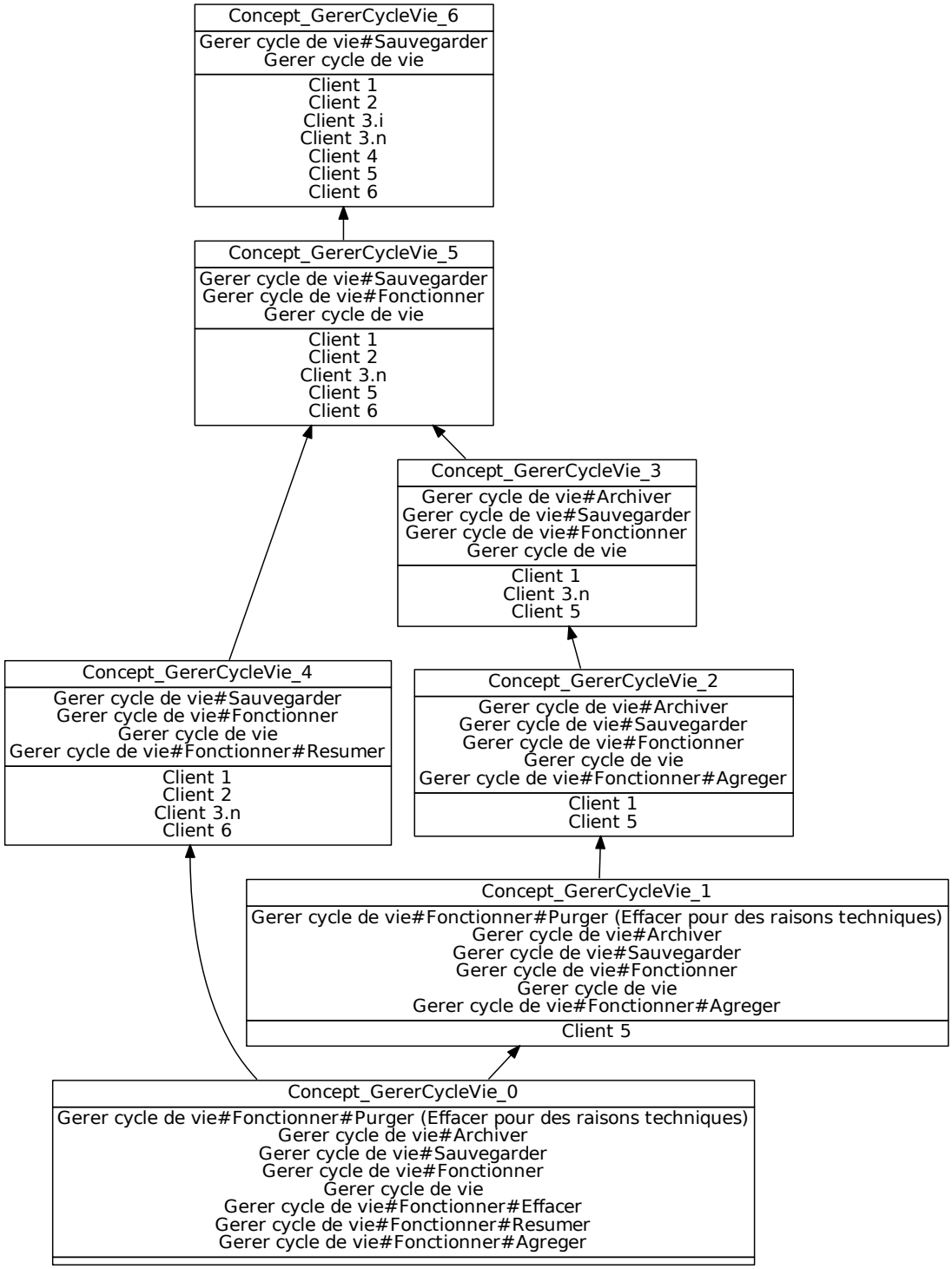


FIGURE *8.36: AC-Gérer-plein*

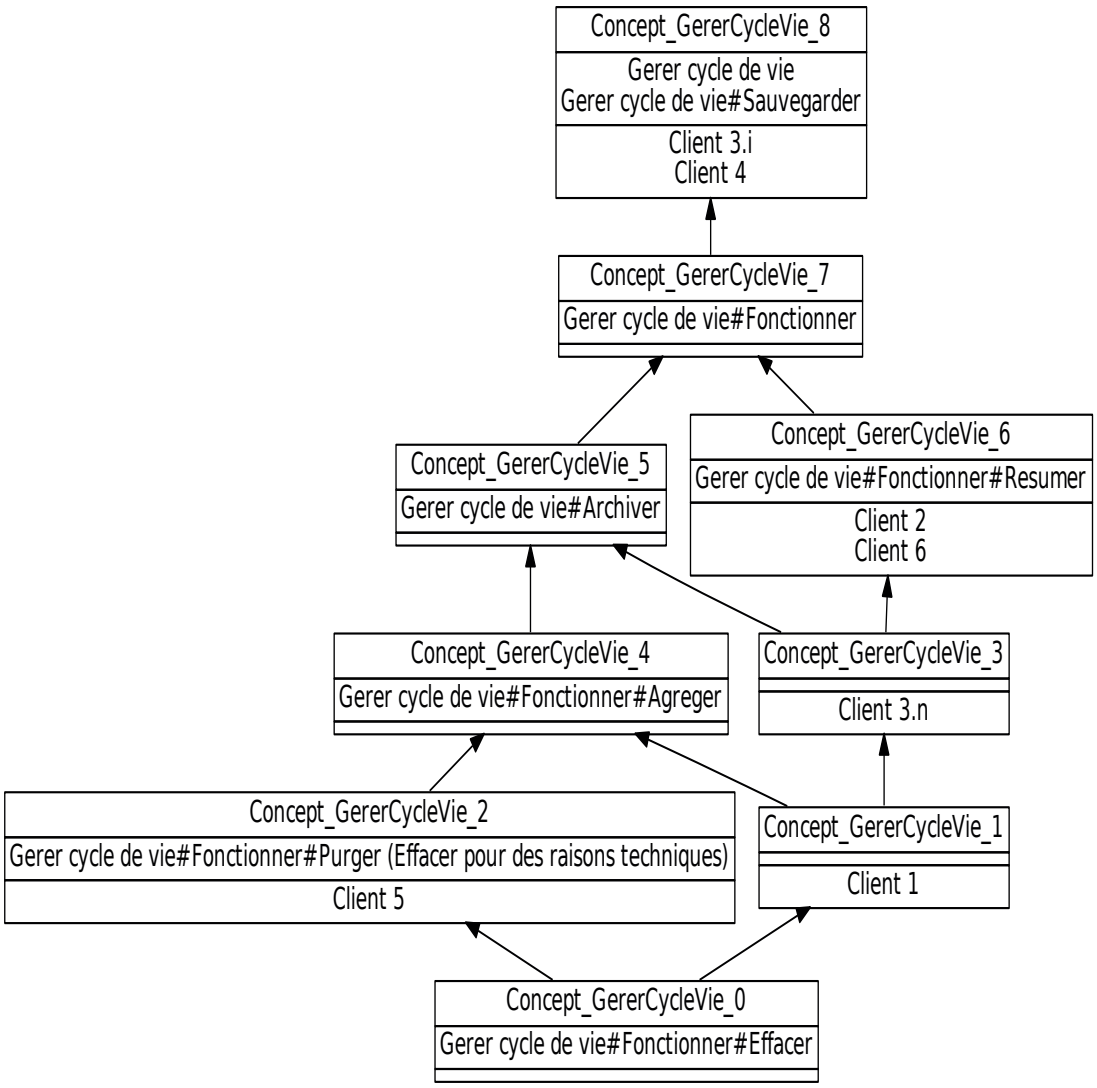


FIGURE *8.37: FCA-Gérer-simple*

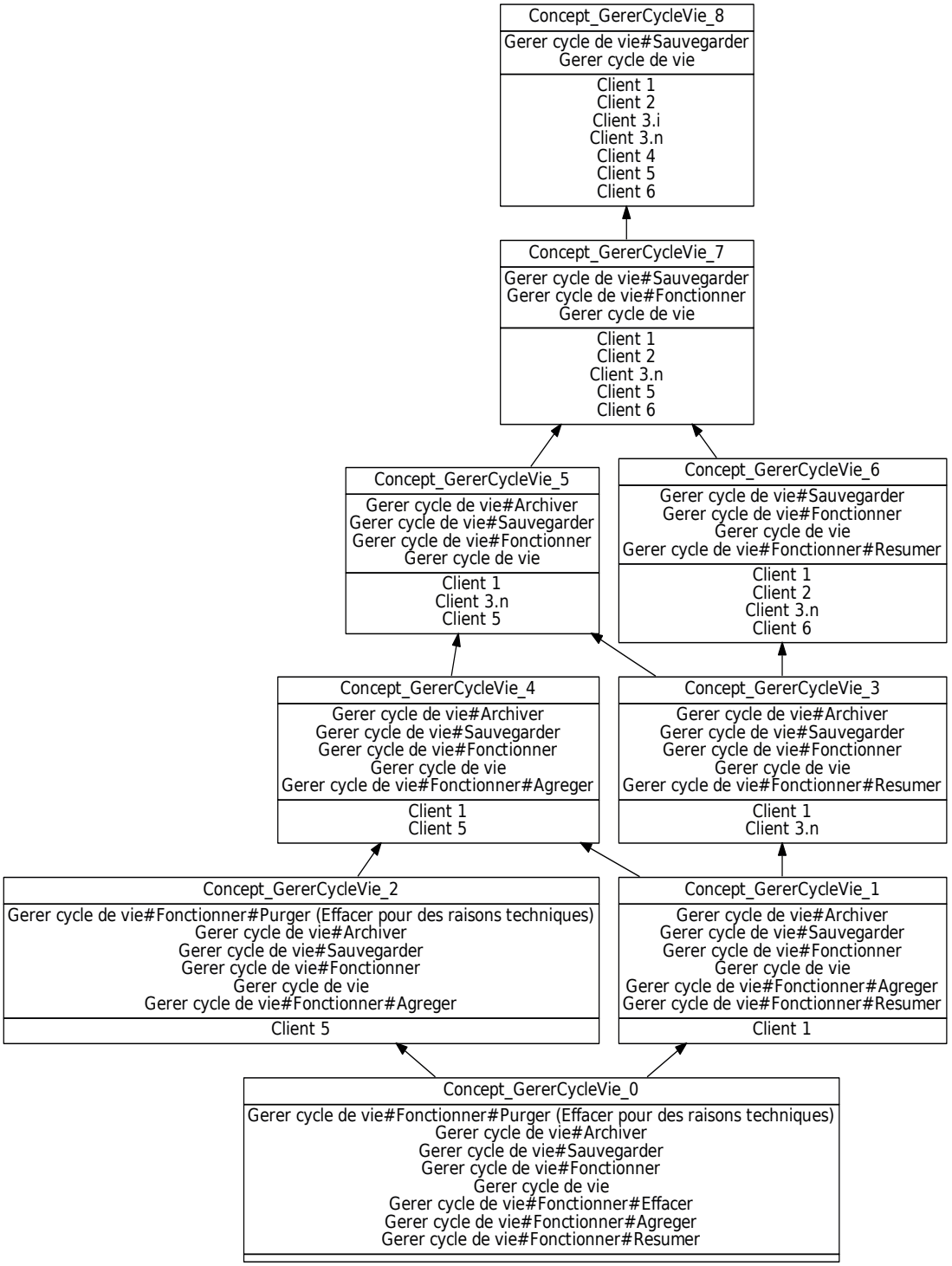


FIGURE *8.38: FCA-Gérer-plein*

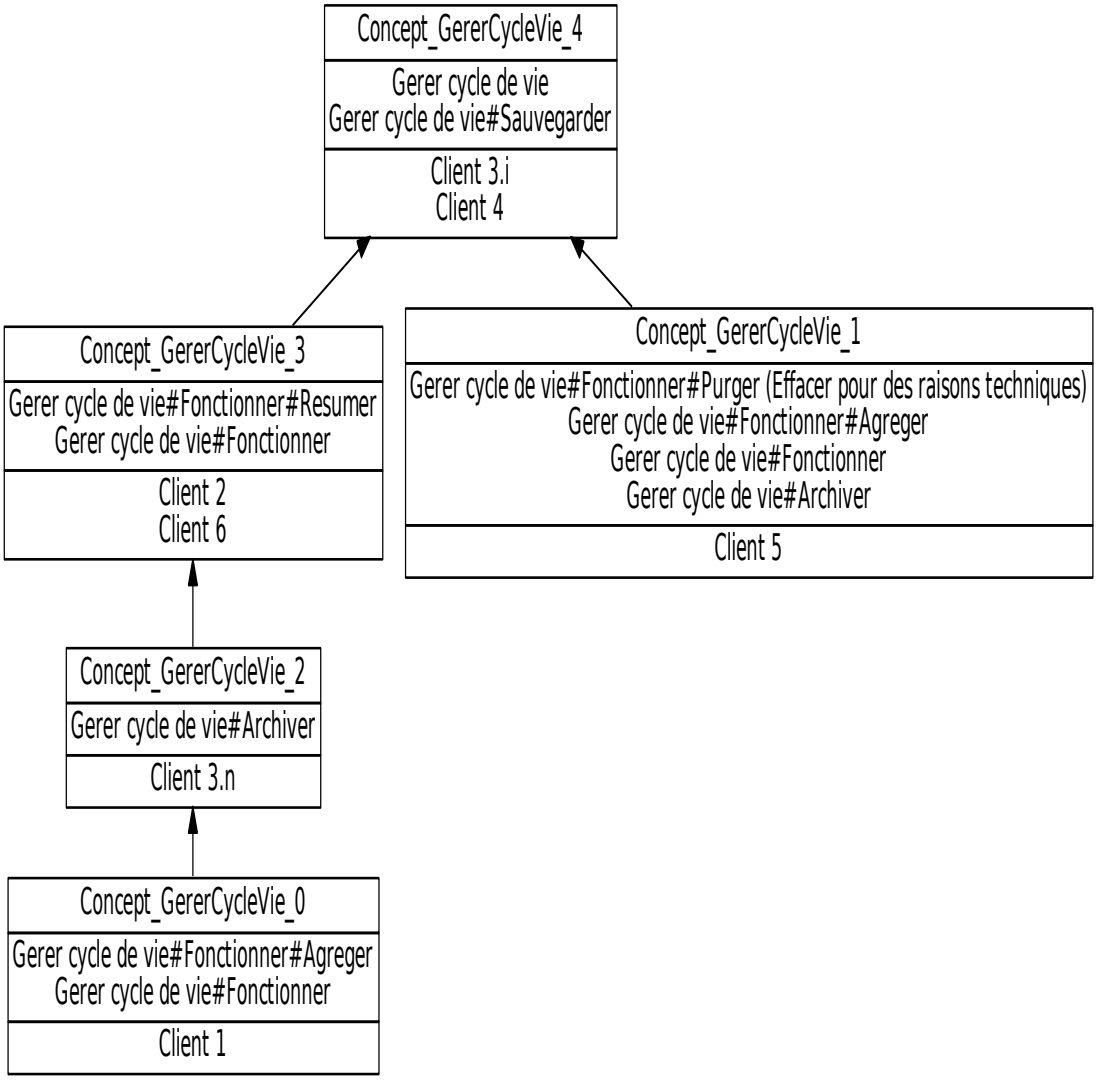


FIGURE *8.39: AOC-Gérer-simple*

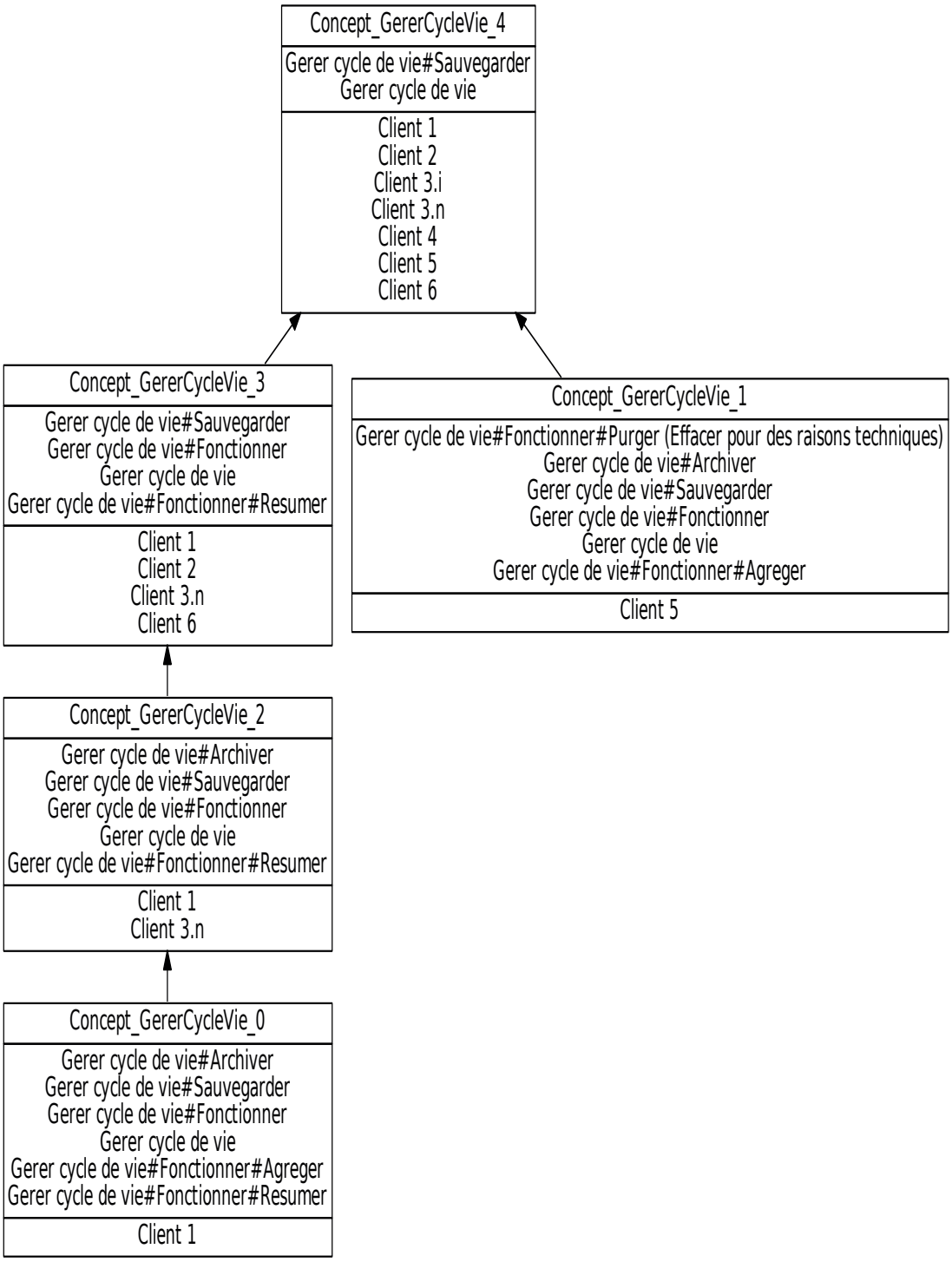


FIGURE *8.40: AOC-Gérer-plein*

Bibliographie

[1] *Software Product Lines : Practices and Patterns*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001.

[2] AEROW. Comparaison des deux approches kimball versus inmon. https://www.aerow.group/a16u1509/, 2016.

[3] H. Alrehamy and C. Walker. Personal data lake with data gravity pull. *Proceedings 2015 Ieee Fifth International Conference on Big Data and Cloud Computing Bdcloud 2015*, pages 160–167, 2015.

[4] E. Annoni, F. Ravat, O. Teste, and G. Zurfluh. Méthode de Développement des Systèmes d'Information Décisionnels : Roue de Deming. In *Congrès Informatique des Organisations et Systèmes d'Information et de Décision - INFORSID'06*, pages 657–673, Hammamet, Tunisia, May 2006.

[5] J. W. Ansari. *Semantic Profiling in Data Lake*. PhD thesis, RWTH Aachen University, 2018.

[6] P.-E. Arduin, M. Grundstein, and C. Rosenthal-Sabroux. *Système d'information et de connaissance*. ISTE Éditions, June 2015.

[7] M. Bala and Z. Alimazighi. Modélisation de processus etl dans un modéle mapreduce, 05 2013.

[8] J. Carbonnel. *L'analyse formelle de concepts : un cadre structurel pour l'étude de la variabilité de familles de logiciels*. PhD thesis, LIRMM, Montpellier, 2018.

[9] A. Castelltort and c. Madera. De l'apport des lacs de donnees pour les observatoires scientifiques. In *Atelier SAGEO*, page 7, 6 Décembre 2016.

[10] M. Chessell, F. Scheepers, N. Nguyen, R. van Kessel, and R. van der Starre. Governing and managing big data for analytics and decision makers. http://www.redbooks.ibm.com/abstracts/redp5120.html?Open, 2014.

[11] CIGREF. valorisation des données. https://www.cigref.fr/publications-numeriques/ebook-cigref-entreprise-2020-enjeux-defis/index.html, 2014.

[12] CIGREF. valorisation des données. https://www.cigref.fr/wp/wp-content/uploads/2016/11/CIGREF-Valorisation-des-donnees-Pratiques-Modele-2016.pdf, 2016.

[13] H. H. D Lindstedt, K Graziano. *The business of data vault modeling*. Morgan Kaufmann, Massachusset, 2009.

[14] Y. Demchenko, C. de Laat, and P. Membrey. Defining architecture components of the big data ecosystem. In *2014 International Conference on Collaboration Technologies and Systems (CTS)*, pages 104–112, May 2014.

[15] J. Dixon. Pentaho, hadoop, and data lakes. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/, 2010.

[16] J. du Net. https ://www.journaldunet.com/solutions/dsi/1194284-base-nosql-laquelle-choisir-pour-quels-besoins/, 2017.

[17] H. Fang. Managing data lakes in big data era : What's a data lake and why has it became popular in data management ecosystem. In *International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 820–824. IEEE, 2015.

[18] H. Fang. 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, 2015.

[19] C. Favre. *Data warehouses' schema evolution : dimension hierarchies updating for analyses personalization*. Theses, Université Lumière - Lyon II, Dec. 2007.

[20] B. Ganter and R. Wille. *Formal concept analysis*. mathematical foundations, 1999.

[21] Gartner. Big data 3vs. https://www.gartner.com/newsroom/id/1731916, 2011.

[22] Gartner. Gartner says beware of the data lake fallacy. http://www.gartner.com/newsroom/id/2809117, 2014.

[23] GARTNER. Real-time insights and decision making using hybrid streaming, in-memory computing analytics and transaction processing. https://www.gartner.com/imagesrv/media-products/pdf/Kx/KX-1-3CZ44RH.pdf, 2016.

[24] N. Gillet. *Optimization of queries over large data in a distributed environment*. Theses, Université de Bordeaux, Mar. 2017.

[25] S. M. Gorry. A framework for management information systems. https://dspace.mit.edu/bitstream/handle/1721.1/47936/frameworkformana00gorr.pdf, 1971.

[26] R. Hai, S. Geisler, and C. Quix. Constance : An intelligent data lake system. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, pages 2097–2100, New York, NY, USA, 2016. ACM.

[27] H. Hashem. *Modélisation intégratrice du traitement BigData*. PhD thesis, Université de Paris Saclay, 2016. Thèse de doctorat dirigée par Cavalli, Ana Réseaux, information et communications Paris Saclay 2016.

[28] A. Hive. https ://cwiki.apache.org, 2018.

[29] Hortonworks. https ://docs.hortonworks.com, 2018.

[30] HULTGREN. https ://hanshultgren.files.wordpress.com/2012/09/data-vault-modeling-guide.pdf, 2012.

[31] Hultgren. *Modeling the Agile Data Warehouse with Data Vault*. Genesee Academy, LCC, 2012.

[32] IBM. Reference architecture : The best of best practices. https://www.ibm.com/developerworks/rational/library/2774.html, 2002.

[33] IDC. Dataage2025. https://www.seagate.com/www-content/our-story/trends/files/Seagate-WP-DataAge2025-March-2017.pdf, 2017.

[34] B. Inmon. *Data Lake architecture*. TechnicsPub.com, Basking Ridge, NJ, USA, 2014.

[35] W. H. Inmon. *Building the Data Warehouse*. QED Information Sciences, Inc., Wellesley, MA, USA, 1992.

[36] Y. P. Janusz Bucki. Pour un renouveau du concept de système d'information, 1994.

[37] J.Carbonnel. *Modelisation de la variabilité des lignes de produits*. Université de Montpellier, 2015.

[38] L. M. J.L. Théorie du système général, 1984.

[39] Jovanovic. https ://aisel.aisnet.org/sais2012/22, 2012.

[40] J. O. Kaldeich C. Data warehouse methodology : A process driven approach. *Lecture Notes in Computer Science*, 3084(1) :342–351, 2004.

[41] R. Kimball and M. Ross. *The Data Warehouse Toolkit : The Definitive Guide to Dimensional Modeling*. Wiley Publishing, 3rd edition, 2013.

[42] M. B. P. J. P. W. L. Rönnbäck, O. Regardt. *Anchor modeling — Agile information modeling in evolving data environments*. Elsevier,Data and Knowledge Engineering, B.V, 2010.

[43] LeBigdata. Qu´ est-ce qu´ un infocentre informatique ? Définition. https://www.lebigdata.fr/infocentre-definition, 2018.

[44] J.-F. Lebraty. Les systèmes décisionnels. In C.-W. I. Akoka, A, editor, *Encyclopédie de l'informatique et des systèmes d'information*, pages 1338–1349. Vuibert, 2006.

[45] MacCrory. Data gravity blog mccrory . https://blog.mccrory.me/2010/12/07/data-gravity-in-the-clouds/, 2010.

[46] MacCrory. Data gravity. https://datagravity.org/about/, 2014.

[47] C. Madera. Le business intelligence, generalités et techniques. https://docplayer.fr/2293463-Le-business-intelligence-generalites-et-techniques.html, 2009.

[48] C. Madera and A. Laurent. The next information architecture evolution : The data lake wave. In *Proceedings of the 8th International Conference on Management of Digital EcoSystems*, MEDES, pages 174–180, New York, NY, USA, 2016. ACM.

[49] C. Madera, A. Laurent, T. Libourel, and A. Miralles. How can the data lake concept influence information system design for agriculture ? In *EFITA CONGRESS* , Montpellier, France, July 2017.

[50] marketsandarkets.com. Data lakes market by software, September 2016. https ://www.marketsandmarkets.com/Market-Reports/data-lakes-market.

[51] MarketsandMarkets. Data lakes market, jun 2016. http://www.marketsandmarkets.com/PressReleases/data-lakes.asp.

[52] Microsoft. Microsoft reference architecture. https://azure.microsoft.com/en-us/blog/technical-reference-implementation-for-enterprise-bi-and-reporting/, 2017.

[53] A. T. Natalia Miloslavskaya. Big data, fast data and data lake concepts. *Procedia Computer Science*, 88(1) :300–305, 2016.

[54] NIST. Nist reference architecture. http://bigdatawg.nist.gov/-uploadfiles/M0226-v10-1554566513.docx, 2012.

[55] I. Nogueira, M. Romdhane, and J. Darmont. Modélisation des métadonnées d'un data lake en data vault. In *18e conférence sur l'Extraction et la Gestion de Connaissances (EGC 2018)*, volume E-34 of *Revue des Nouvelles Technologies de l'Information*, pages 257–262, Paris, France, Jan. 2018.

[56] Parcell. https ://www.whitepapers.em360tech.com, 2012.

[57] A. Perrot. *Information Visualization in the Big Data era : tackling scalability issues using multiscale abstractions*. Theses, Université de Bordeaux, Nov. 2017.

[58] D. J. Power. Understanding data-driven decision support systems. *Information Systems Management*, 25(2) :149–154, 2008.

[59] D. J. Power. Using 'big data' for analytics and decision support. *Journal of Decision Systems*, 23(2) :222–228, 2014.

[60] PWC. https ://www.cio.com/article/3003538/big-data/study-reveals-that-most-companies-are-failing-at-big-data.html, 2015.

[61] F. Ravat. *Models and tools for designing and using decision support systems*. Habilitation à diriger des recherches, Université des Sciences Sociales - Toulouse I, Dec. 2007.

[62] F. Ravat, O. Teste, and Z. Gilles. Modélisation et extraction de données pour un entrepôt objet. In *Bases de données avancées (BDA 2000)*, pages 119–138, Blois, France, Oct. 2000.

[63] O. Regardt, L. Rönnbäck, M. Bergholtz, P. Johannesson, and P. Wohed. Anchor modeling. In A. H. F. Laender, S. Castano, U. Dayal, F. Casati, and J. P. M. de Oliveira, editors, *Conceptual Modeling - ER 2009*, pages 234–250, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[64] F. Rodhain, B. Fallery, A. Girard, and S. Desq. Une histoire de la recherche en Systèmes d'Information, à travers 30 trente ans de publications. *Entreprises et Histoire*, (61) :78–97, 2010.

[65] F. Role. Panorama des travaux en cours dans le domaine des métadonnées. Research Report RR-3628, INRIA, 1999.

[66] Russom. Data Lakes : Purposes, Practices, Patterns, and Platforms. https://tdwi.org/research, 2017.

[67] P. Russom. Best practices report | data lakes : Purposes, practices, patterns, and platforms. Technical report, TDWI, March 29 2017.

[68] J. Sansen. *Information visualization for big data : a data abstraction approach*. Theses, Université de Bordeaux, July 2017.

[69] S. Servigne. Conception, architecture et urbanisation des systèmes d'information. *Encyclopædia Universalis*, pages 1–15, June 2010.

[70] S. Servigne. Conception, architecture et urbanisation des systèmes d'information. *Encyclopædia Universalis*, pages 1–15, 2010.

[71] K. Shvachko, H. Kuang, S. Radia, and R. Chansler. The hadoop distributed file system. In *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, MSST '10, pages 1–10, Washington, DC, USA, 2010. IEEE Computer Society.

[72] B. Stein and P. Alan Morrison. The enterprise data lake : Better integration and deeper analytics, 2014.

[73] I. Suriarachchi and B. Plale. Provenance as essential infrastructure for data lakes. In M. Mattoso and B. Glavic, editors, *Provenance and Annotation of Data and Processes*, pages 178–182, Cham, 2016. Springer International Publishing.

[74] Teradata. Teradata. https://www.teradata.fr/, 2018.

[75] I. G. Terrizzano, P. M. Schwarz, M. Roth, and J. E. Colino. Data wrangling : The challenging yourney from the wild to the lake. In *CIDR*, 2015.

[76] O. Teste. *Modélisation et manipulation d'entrepôts de données complexes et historisées*. Theses, Université Paul Sabatier - Toulouse III, Dec. 2000.

[77] P. Tyagi and H. Demirkan. http ://analytics-magazine. org, 2016.

[78] usine digitale. https ://www.usine-digitale.fr/article/la-transformation-digitale-a-2-milliards-de-dollars-de-hsbc.n431452, 2016.

[79] P. Vassiliadis, A. Karagiannis, V. Tziovara, and A. Simitsis. Towards a benchmark for etl workflows, 01 2007.

[80] C. Vitari and E. Raguseo. Données massives : l'évaluation de leur valeur économique et l'impact sur les performances des entreprises. Research report, Grenoble Ecole de Management, 2016.

[81] D. L. WH Inmon. *Data architecture : a primer for the data scientist : big data, data warehouse and data vault*. Morgan Kaufmann, Massachusset, 2015.