

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Kasdi Merbah, Ouargla
Faculté des Nouvelles Technologies de l'Information et de la Communication
Département d'informatique et de technologie de l'information

N° d'ordre :



THÈSE

*Présentée en vue de l'obtention du diplôme de **Doctorat LMD** en Informatique*

Option : Intelligence artificielle et technologies de l'information

Titre

Intégration de la Sémantique dans le Big Data

Présenté Par :

Zaoui SAYAH

Soutenue publiquement le : 27/02/2021

Devant le jury composé de :

Président :	Mr. Fouad CHEBBARA	Prof.	Université de Ouargla.
Rapporteur :	Mr. Okba KAZAR	Prof.	Université de Biskra.
Examineur :	Mme. Fatima Zohra LAALLAM	Prof.	Université de Ouargla.
Examineur :	Mr. Amine KHALDI	M. C. (A)	Université de Ouargla.
Examineur :	Mr. Mohamed Redouane KAFI	M. C. (A)	Université de Ouargla.
Examineur :	Mr. Khaled REZEG	Prof.	Université de Biskra.

Année académique: 2020/2021

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Au nom de dieu le clément, le
miséricordieux*

*In the Name of Allah the Merciful, the
Compassionate*

Remerciements

Avant, toute personne, je tiens à remercier notre Dieu Tout Puissant pour m'avoir éclairci le chemin de ce travail.

*Mes vifs remerciements vont également à Monsieur **Okba KAZAR** professeur à l'université de Biskra, d'avoir assuré l'encadrement de cette thèse ainsi que pour ses précieux conseils et la confiance qu'il m'a accordée qui ont fortement contribué à mener à bien ce travail.*

*Je tiens à remercier vivement Monsieur **Fouad CHEBBARA M. C. (A)** à l'université de Ouargla, pour m'avoir fait l'honneur de présider le jury de ma soutenance.*

*Je tiens à exprimer toute ma gratitude à Mme. **Fatima Zohra LAALLAM** Professeur à l'université de Ouargla, Monsieur **Amine KHALDI M. C. (A)** à l'université de Ouargla, Monsieur **Mohamed Redouane KAFI M. C. (A)** à l'université de Ouargla, Monsieur **Khaled REZEG, M. C. (A)** à l'université de Biskra, pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail et faisant partie de jury de soutenance.*

Enfin, Un grand merci s'adresse à ceux qui m'ont aidé de près ou de loin.

Dédicaces

Je dédie ce modeste travail:

À ceux que j'ai de plus cher au monde : mes parents;

À ma femme et mes petits garçons Taha Yacine, Zidane et Louai

À mes frères et mes sœurs et mes neveux.

À tous mes amis et collègues.

À tous ceux qui m'ont aimé et me souhaitent le bonheur et la réussite.

ملخص

في الوقت الحاضر، يقوم مستخدمو الويب والأنظمة بإفراط في تحميل الويب من خلال إنتاج كبير من البيانات الضخمة مما يؤدي إلى جعل البيانات الكبيرة أكثر أهمية في العديد من المجالات مثل الشبكات الاجتماعية، إنترنت الأشياء ، الرعاية الصحية ، المدن الذكية ، سلامة الطيران ، إلخ. أصبح استخدام البيانات الكبيرة أمرًا بالغ الأهمية للشركات نظرًا للتطور الهائل لموفري المعلومات والمستخدمين على الويب. مع ذلك، البيانات الكبيرة ، ليست ذات معنى بدون دلالات. لفهم البيانات الضخمة جيدًا، نطرح أسئلة حول العلاقات بين البيانات الضخمة والدلالات وكيف يمكن أن تساعد الدلالات.

للتغلب على هذه المشكلة ، يصبح دمج الأنطولوجيا في البيانات الكبيرة حلاً فعالاً يضمن قابلية التشغيل البيئي الموثوقة بين الأنظمة ويجعل البيانات الكبيرة أكثر فائدة. هذه التكنولوجيا يمكن أن تخفي عدم تجانس موارد البيانات المختلفة وتمثيلها. في هذه الرسالة، نقدم لمحة عامة عن التكامل الدلالي في البيانات الكبيرة ونراجع الأعمال المختلفة التي تدرس توفير الطاقة في المنازل والمدن الذكية. لتحقيق ذلك ، نقدم بنية فعالة قائمة على الدمج بين الأنطولوجيا والبيانات الكبيرة ونظام الوكلاء المتعددين. بالإضافة إلى ذلك ، تظهر الطريقة المقترحة الإمكانيات التي توفرها هذه التقنيات لتقليل استهلاك الطاقة في المدن الذكية ، وتشجيع تغيير السلوك للمقيمين ، وتوفير الراحة المطلوبة وتنفيذ سياسة طاقة فعالة طويلة الأجل. في بيئة ذكية ومستدامة.

أظهرت النتائج التجريبية أن الطريقة المقترحة تمكن من تحسين استهلاك الطاقة إلى حد كبير وتقديم مستوى عالٍ من الاستقلالية وضمان قابلية التوسع. بالإضافة إلى ذلك، تم تقليل وقت التنفيذ نسبيًا. بالإضافة إلى ذلك، أثبتت طريقتنا فعاليتها ضد العديد من الأساليب الحديثة.

الكلمات الرئيسية: البيانات الضخمة ، توفير الطاقة ، **Hadoop** ، التشغيل المتداخل ، نظام متعدد الوكلاء ،

الأنطولوجيا ، التكامل الدلالي ، المدن الذكية.

Résumé

De nos jours, les utilisateurs et les systèmes du web surchargent le web avec une génération exponentielle d'une énorme quantité de données. Ce qui a rendu les Big Data plus importantes dans plusieurs domaines tels que les réseaux sociaux, l'internet des objets, les soins de santé, les villes intelligentes, la sécurité aérienne, etc. L'utilisation des Big Data est devenue de plus en plus cruciale pour les entreprises en raison de l'énorme évolution des fournisseurs et des utilisateurs d'informations sur le web. Cependant, les Big Data ne sont pas significatives sans sémantique. Pour bien comprendre les Big Data, nous posons des questions sur les relations entre les Big Data et la sémantique et sur la façon dont la sémantique peut aider.

Pour surmonter ce problème, l'intégration d'ontologies dans les Big Data devient une solution efficace qui assure une interopérabilité fiable entre les systèmes et rend les Big Data plus utiles. Cette technologie peut cacher l'hétérogénéité des différentes ressources de données et de leurs représentations. Dans cette thèse, nous donnons un aperçu de l'intégration sémantique dans les Big Data et passons en revue les divers travaux qui étudient l'économie d'énergie dans les maisons et les villes intelligentes. Pour y parvenir, nous proposons une architecture efficace basée sur la combinaison entre l'ontologie, les Big Data et le système multi-agents. En outre, l'approche proposée montre les potentialités offertes par ces technologies pour réduire la consommation d'énergie dans les villes intelligentes, promouvoir le changement de comportement des habitants, offrir le confort requis et mettre en œuvre une politique énergétique efficace à long terme dans un environnement intelligent et durable.

Les résultats expérimentaux ont montré que l'approche proposée a permis d'optimiser considérablement la consommation d'énergie, offrir un niveau élevé d'autonomie et garantir l'évolutivité. En outre, le temps d'exécution a été comparativement réduit. De plus, notre approche a prouvé son efficacité face à plusieurs méthodes citées dans l'état de l'art.

Mots-clés: Big Data, Économie d'énergie, Hadoop, Interopérabilité, Système Multi-Agents, Ontologie, Intégration Sémantique, Villes Intelligentes.

Abstract

Nowadays, web users and systems overload the web with an exponential generation of huge amount of data that leads to making Big Data more important in several domains such as social networks, internet of things, health care, smart cities, aviation safety, etc. The use of Big Data has become increasingly crucial for companies due to the huge evolution of information providers and users on the web. However, Big Data remains not meaningful without semantic, to get a good comprehension about Big Data we ask questions about how Big Data and semantic are related to each other and how Semantic may help.

To overcome this issue, ontology integration in Big Data becomes an efficient solution that ensures a reliable interoperability between systems and makes Big Data more useful. This technology can hide the heterogeneity of different data resources and their representation. In this thesis, we provide an overview of semantics integration in Big Data and review various works that investigate energy saving in smart homes and cities. To achieve this, we propose an efficient architecture based on the combination between ontology, Big Data, and Multi-Agent System. Furthermore, the proposed approach shows the opportunities of these technologies to reduce energy consumption in smart cities, promotes the change of inhabitant's behavior, offer the required comfort and implement an effective long-term energy policy in a smart and sustainable environment.

Experimental results have shown that the proposed approach has optimized significantly the energy consumption, provide a high level of autonomy and ensure scalability. Furthermore, the execution time was comparatively reduced. Moreover, our approach has proven its strength against several methods from the state-of-the-art.

Keywords: Big Data, Energy Saving, Hadoop, Interoperability, Multi-Agent System, Ontology, Semantic Integration, Smart Cities.

Table des matières

Remerciements	I
Resume	IV
Table des matieres	VI
Liste des figures.....	VIII
Liste des tableaux	IX

INTRODUCTION GENERALE.....	1
----------------------------	---

CHAPITRE I. ETAT DE L'ART SUR LES BIG DATA ET LE SYSTEME MULTI AGENT

I.1 INTRODUCTION	6
I.2 C'EST QUOI LE BIG DATA ?.....	6
I.2.1 Volume (Données dans les supports de stockage)	7
I.2.2 Variété (Données sous plusieurs formes).....	8
I.2.3 Vitesse (données en mouvement)	8
I.2.4 Valeur (la valeur des données extraites)	9
I.2.5 Véracité (la qualité ou la fiabilité des données).....	10
I.3 HISTORIQUE ET DEVELOPPEMENT DU BIG DATA	11
I.3.1 Première génération : Big Data 1.0 (1994-2004)	12
I.3.2 Deuxième génération : Big Data 2.0 (2005-2014)	12
I.3.3 Troisième génération : Big Data 3.0 (2015-...).....	13
I.4 DOMAINES DE BIG DATA	14
I.5 ANALYSE DE BIG DATA	15
I.5.1 Analyse du texte.....	15
I.5.2 Analyse audio	16
I.5.3 Analyse vidéo	16
I.5.4 Analyse des réseaux sociaux	16
I.5.5 Analyse prédictive.....	17
I.6 TECHNOLOGIES DE BIG DATA	17
I.6.1 Hadoop.....	18
I.6.2 HDFS.....	20
I.6.3 MapReduce.....	21
I.7 LES SYSTEMES MULTI-AGENTS (SMA)	23
I.7.1 Concept d'agent	24
I.7.2 Caractéristiques du Système multi-agents	25
I.7.3 La plateforme de développement des SMA	26
I.7.4 Les raisons du choix du paradigme d'agent	27
I.8 CONCLUSION.....	27

CHAPITRE II. LE WEB SEMANTIQUE ET LES ONTOLOGIES

II.1 INTRODUCTION	28
II.2 WEB SEMANTIQUE.....	28
II.3 ARCHITECTURE DU WEB SEMANTIQUE.....	29
II.4 L'ONTOLOGIE.....	31
II.4.1 Composants d'ontologie	33
II.5 RAISONS DE L'UTILISATION DES ONTOLOGIES.....	34
II.6 LES TYPES D'ONTOLOGIES.....	35
II.6.1 Les ontologies terminologiques	35
II.6.2 Les ontologies d'information	35
II.6.3 Les ontologies de la connaissance.....	35
II.6.4 Les ontologies d'applications.....	36
II.6.5 Les ontologies de domaine	36
II.6.6 Les ontologies génériques (haut niveau).....	36
II.6.7 Les ontologies de représentation.....	37
II.7 CONSTRUCTION D'ONTOLOGIES.....	37

II.8 FORMALISMES ET LANGAGES DE REPRESENTATION D'ONTOLOGIES	39
II.8.1 Formalismes de représentation d'ontologie	39
II.8.2 Langages de représentation d'ontologies	40
II.8.3 Éditeur d'ontologies	49
II.9 CONCLUSION.....	52
CHAPITRE III. INTEGRATION DE LA SEMANTIQUE DANS LE BIG DATA	
III.1 INTRODUCTION	54
III.2 DES SYSTEMES D'INTEGRATION DE DONNEES VERS L'INTEGRATION SEMANTIQUE DE DONNEES	54
III.3 NECESSITE DE L'ONTOLOGIE DANS LE BIG DATA	54
III.3.1 L'intégration de la sémantique est-il un défi ?	55
III.3.2 Intégration d'ontologie au cours du processus de construction	55
III.3.3 Que signifie l'intégration d'ontologies	56
III.3.4 Processus d'intégration d'ontologies.....	57
III.3.5 Comment peut-on améliorer l'intégration.....	58
III.4 APPROCHES DE CONSTRUCTION DES ONTOLOGIES.....	59
III.4.1 Approche mono-ontologie (single-ontology approach).....	59
III.4.2 Approche multi-ontologies.....	60
III.4.3 Approche hybride.....	61
III.5 ROLE DES ONTOLOGIES DANS LE BIG DATA.....	62
III.5.1 Le mapping comme solution pour l'intégration d'ontologies	63
III.5.2 Comment l'intégration d'ontologie peut relever les défis du Big Data ?	63
III.6 COMPARAISON DES TRAVAUX BASES SUR L'INTEGRATION D'ONTOLOGIES DANS LE BIG DATA.....	65
III.7 DISCUSSION	69
III.8 CONCLUSION.....	70
CHAPITRE IV. MODELISATION DE L'APPROCHE	
IV.1 INTRODUCTION	71
IV.2 CONTEXTE DE L'ETUDE	71
IV.2.1 Intégration sémantique dans les Big Data pour les villes intelligentes.....	71
IV.3 ÉCONOMIE DE L'ENERGIE DANS LES VILLES INTELLIGENTES	73
IV.3.1 Travaux connexes.....	73
IV.4 PROPOSITION D'ARCHITECTURE MULTICOUCHE	76
IV.4.1 Objectif du système	76
IV.4.2 Architecture générale du système	77
IV.4.3 Diagramme de séquence AUML de l'architecture proposée	79
IV.4.4 Fonctionnement du système	80
IV.4.5 Ontologie de maisons intelligentes Onto-SB	81
IV.4.6 Module moteur de raisonnement (REM)	84
IV.5 PRESENTATION DE L'ENVIRONNEMENT.....	85
IV.5.1 Description des scénarios.....	87
IV.6 MISE EN ŒUVRE	90
IV.7 CONCLUSION.....	94
CHAPITRE V. EXPERIMENTATION ET RESULTATS	
V.1 INTRODUCTION	96
V.2 PREMIERE SIMULATION.....	96
V.3 DEUXIEME SIMULATION.....	98
V.4 DISCUSSION	101
V.5 CONCLUSION.....	102
CONCLUSION GENERALE	104
LISTE DES TRAVAUX	106
BIBLIOGRAPHIE	107

Liste des figures

CHAPITRE I. ETAT DE L'ART SUR LES BIG DATA ET LE SYSTEME MULTI AGENT

FIGURE 1 CARACTERISTIQUES DE BIG DATA	7
FIGURE 2 CROISSANCE DES DONNEES PAR AN (EXAOCTETS)	8
FIGURE 3 BIG DATA EN CONSTANTE EVOLUTION	10
FIGURE 4 LES DIFFERENTES SOURCES DE BIG DATA.	15
FIGURE 5 ARCHITECTURE HDFS DE HADOOP	21
FIGURE 6 SCHEMA DE FONCTIONNEMENT DE MAPREDUCE	22
FIGURE 7 HADOOP ECOSYSTEM.	23

CHAPITRE II. LE WEB SEMANTIQUE ET LES ONTOLOGIES

FIGURE 8 ARCHITECTURE DU WEB SEMANTIQUE PROPOSE PER TIM BERNERS-LEE	31
FIGURE 9 CLASSIFICATION DES ONTOLOGIES	35
FIGURE 10 TYPOLOGIE D`ONTOLOGIE SELON LE NIVEAU DE CONCEPTUALISATION	36
FIGURE 11 CLASSES ET PROPRIETES DEFINIES PAR LE RDFS	45
FIGURE 12 LES FAMILLES DE LANGAGES QUI ONT CONDUIT A L'OWL.	48

CHAPITRE III. INTEGRATION DE LA SEMANTIQUE DANS LE BIG DATA

FIGURE 13 APPROCHE MONO-ONTOLOGIE.	60
FIGURE 14 APPROCHE MULTI-ONTOLOGIES.	60
FIGURE 15 APPROCHE HYBRIDE.	61

CHAPITRE IV. MODELISATION DE L'APPROCHE

FIGURE 16 UNE ARCHITECTURE MULTICOUCHE POUR ISEM-SC.	78
FIGURE 17 DIAGRAMME DE SEQUENCE AUML DE L'ARCHITECTURE ISEM-SC.	80
FIGURE 18 LA TAXONOMIE UTILISEE DANS LES CONCEPTS DE L'ONTO-SB	82
FIGURE 19 REGLES SWRL UTILISEES POUR GERER L'ENERGIE DES APPAREILS.	84
FIGURE 20 LA TEMPERATURE REELLE DU 26 AOUT 2019	89
FIGURE 21 L'IMPLEMENTATION JADE DE DIFFERENTES COUCHES D'AGENTS.	91

CHAPITRE V. EXPERIMENTATION ET RESULTATS

FIGURE 22 TEMPS DE FONCTIONNEMENT DES APPAREILS AVEC ET SANS REGLES.	97
FIGURE 23 CONSOMMATION D'ENERGIE DANS LA VILLE INTELLIGENTE POUR DIFFERENTS SCENARIOS.	98
FIGURE 24 INITIALISATION DES PARAMETRES DU SYSTEME DE SIMULATION.	99
FIGURE 25 SIMULATION DE SCENARIOS DE TEMPS D'EXECUTION DANS 10 MAISONS INTELLIGENTES.	100
FIGURE 26 SIMULATION DE SCENARIOS DE TEMPS D'EXECUTION DANS 20 MAISONS INTELLIGENTES.	100
FIGURE 27 LE TEMPS D'EXECUTION DANS DIFFERENTS SCENARIOS.	101

Liste des tableaux

CHAPITRE I. ETAT DE L'ART SUR LES BIG DATA ET LE SYSTEME MULTI AGENT

CHAPITRE II. LE WEB SEMANTIQUE ET LES ONTOLOGIES

TABLEAU 1 COMPARAISON DE DIFFERENT EDITEURS D` ONTOLOGIES	52
TABLEAU 2 UNE COMPARAISON ENTRE LES TROIS APPROCHES D'INTEGRATION.	62
TABLEAU 3 COMPARAISON DES TRAVAUX BASES SUR L'INTEGRATION DE LA SEMANTIQUE DANS LE BIG DATA.	65

Chapitre IV. Modélisation de l'approche

TABLEAU 4 LES COUCHES DE L'ARCHITECTURE ET LES TECHNOLOGIES ASSOCIEES UTILISEES.	77
TABLEAU 5 CONCEPTS, INSTANCES, ATTRIBUTS ET RELATIONS DE LA MAISON INTELLIGENTE.	83
TABLEAU 6 ILLUSTRATION DES APPAREILS UTILISES DANS LE MODELE DE MAISON INTELLIGENTE.	86
TABLEAU 7 LES CAPTEURS UTILISES ET LEURS ROLES DANS LA MAISON INTELLIGENTE.	87
TABLEAU 8 UN SCENARIO SIMPLE DE ROUTINES QUOTIDIENNES DANS LA MAISON INTELLIGENTE.	88
TABLEAU 9 CONSOMMATION D'ENERGIE DANS UNE MAISON INTELLIGENTE PENDANT LA PERIODE DE 24 HEURES.	90
TABLEAU 10 ILLUSTRATION DES MATERIELS UTILISES POUR LA MISE EN ŒUVRE.	91
Chapitre V. Expérimentation et Résultats	
TABLEAU 11 ÉNERGIE ECONOMISEE DANS DIVERS APPAREILS AVEC LES REGLES UTILISEES.	97
TABLEAU 12 COMPARAISON DES TECHNOLOGIES UTILISEES ET LEURS AVANTAGES DANS LES TRAVAUX EXPERIMENTES.	102
TABLEAU 13 COMPARAISON DES CARACTERISTIQUES PRINCIPALES D'ECONOMIE D'ENERGIE DANS LES TRAVAUX EXPERIMENTES.	1022

Introduction générale

Au cours des deux dernières décennies, les individus et les systèmes génèrent quotidiennement une énorme quantité de données à partir de sources hétérogènes et surchargent le web avec un volume exponentiel de données. Ces données numériques sont continuellement produites à partir de millions d'appareils et d'applications (les réseaux sociaux, téléphones intelligents, capteurs, etc.). Dans cette période, la notion de Big Data est apparue pour la première fois dans un article scientifique en 1997 à ACM (Association for Computing Machines), qui a souligné les défis technologiques de la visualisation de grands ensembles de données. Cette notion est introduit en informatique par **Roger Magoulas** en 2005 (Oussous et al., 2017). Le Big Data est défini par **Gartner** « *Les Big Data sont des ressources d'information volumineuses, à haute vitesse et / ou de haute qualité qui nécessitent de nouvelles formes de traitement pour améliorer la prise de décision, la découverte et le processus d'optimisation* » (Eine et al., 2017). Dans les dernières années la majorité des compagnies sont trouvés émergés dans le paradigme Big Data y compris (EMC, Oracle, IBM, Microsoft, Google, Amazon, and Facebook, etc.). L'importance du Big Data est très appréciée dans de nombreux domaines vitaux tels que les villes intelligentes, L'IoT, la santé, la gestion du trafic aérien, etc.

En 2011, selon un rapport de l'International Data Corporation (IDC), le volume global de données générées et stockées dans le monde était de 1,8 ZB. Ce chiffre a été multiplié par neuf environ au bout de cinq ans, ce nombre étant estimé à doubler tous les deux ans dans un proche avenir. Par exemple, Google traite des données de centaines de PB, Facebook génère également plus de 10 PB de données par mois (Chen et al., 2014). Cet aspect de grand volume de données apparaît comme un défi pour qu'il soit manipulé par des outils de traitement de données classiques ainsi avec des moyens existants, ce qui a conduit à la recherche des nouveaux moyens et technologies capables de traiter ce gros volume de donnée qui dépasse l'échèle de dizaines de PB dans un temps raisonnable. Non seulement l'aspect de gros volume qui caractérise le Big Data, selon **lany** qui a donné une définition bien connue (appelé 3Vs) pour expliquer le sens de Big Data : le volume, la vitesse et la variété (Chen et al., 2014). Cette définition de 3 Vs signifie respectivement que la taille de donnée est grande (échelle de 1 téraoctet et plus) (Lee, 2017), le processus de création de données est très rapide et les données sont existées sous différents types et sont capturées à partir de différentes sources. Autres études (Tsai et al., 2015) ont ajouté

Introduction Générale

que la définition de 3Vs est insuffisante pour bien couvrir la notion de Big Data d'aujourd'hui. Ainsi, la véracité, la validité, la valeur, la variabilité, le lieu, le vocabulaire et le vague (imprécision) ont été ajoutés au 3Vs pour bien éclaircir le terme de Big Data.

Sur le plan de marché de Big Data, la commercialisation est atteinte d'environ 16,1 milliards de dollars 2014 et prévoit qu'il atteindra 512 milliards de dollars d'ici 2026, selon toujours (IDC). Par exemple, Google va investir 100 millions de dollars dont 8,5 millions seront offerts à une dizaine des institutions utilisant le Big Data ou l'intelligence artificielle pour affronter le Covid-19. En plus de la valeur de marché occupé par le Big Data, l'importance se figure dans d'autres domaines vitaux tels que le contrôle et la prévention des maladies épidémique, la commerce électronique, les villes intelligentes et la gestion du trafic aérien, etc. (Oussous et al., 2017). Grâce au Big Data, les responsables des entreprises peuvent réellement mesurer et améliorer la connaissance globale sur leurs activités et traduire directement ces connaissances à des décisions plus sûres, ce qui permet aux managers de prendre des décisions sur la base des preuves plutôt que d'intuitions. Ce qui conduit à une baisse considérable de la perte de temps, efforts et moyens d'un côté et un gain significatif en métiers de profits.

Problématique

Avec le progrès exponentiel de Big Data et la nécessité inévitable pour une bonne maîtrise de ces domaines que ce soit pour des raisons commerciales, stratégiques ou bien politiques et sociaux, les outils et les technologies traditionnels affichent un grand fiasco pour manipuler le volume colossal de données et la variété de différents types données afin de chercher, analyser, stocker et extraire des connaissances exploitables par les décideurs (Labrinidis and Jagadish, 2012). Pour remédier à ce déficit, des nombreux techniques et outils matériels et logiciels sont développés pour fournir davantage une capacité de stockage, de traitement en parallèle et d'analyse en temps réel de différentes sources hétérogènes et des données complexes généralement semi structurés ou non structurés ainsi la garantie de la confidentialité et la sécurité des données (Oussous et al., 2017). L'avantage acquis par ces solutions est d'offrir plus de fiabilité, de flexibilité, d'évolutivité et de performance avec un coût que se diminue continuellement en raison de l'avancée technologique. Ces outils se caractérisent essentiellement par le Hadoop qui a donné une bonne opportunité de grande capacité de stockage et de traitement rapide de

données grâce à ces deux composants principaux HDFS et MapReduce (Shvachko et al., 2010).

Néanmoins, l'exploitation des Big Data non seulement un défi de stockage et de gestion d'une variété de données, mais le manque d'interopérabilité entre les ressources hétérogènes engendre des problèmes inhérents, ce qui rend le partage des données et la réutilisation des connaissances une tâche difficile dans les applications de Big Data (Rani et al., 2017). En conséquent, la compréhension et l'extraction de connaissance cohérente de ces données devient une exigence primordiale pour répondre aux besoins excessifs des clients et des entreprises notamment les décideurs (Abbes and Gargouri, 2018). Pour surmonter ce défi, il devient impératif de doter les Big Data de la sémantique et de permettre une vue standard pour assurer une interopérabilité efficace entre les applications (Bansal and Kagemann, 2015). De plus, les données deviennent lisibles, compréhensives et interprétables, elles pourraient être aussi partagées et échangées entre les individus, les systèmes et les organisations sans effort particulier (Eine et al., 2017).

Ces difficultés énoncées ci-dessus, se figurent essentiellement dans le domaine des villes intelligentes. De fait, qu'elles génèrent un énorme volume de données hétérogènes qui parvient de million des capteurs. La gestion de l'énergie est considérée comme un axe de recherche important vise à trouver des solutions efficaces pour empêcher la perte d'énergie et optimiser la consommation. Cependant, la difficulté de l'interopérabilité, le grand volume et diversité de données apparaît comme un défis majeur pour instaurer un tel system de gestion efficace.

Motivation

Selon le volume et la complexité des informations produites et les bases de données et leurs interconnexions notamment les données générées par les capteurs et les appareils dans les villes intelligentes. Des nouvelles techniques et approches doivent être développés en raison de faire sortir et profiter la valeur des données à partir du Big Data et extraire des nouvelles connaissances, qui ne sont pas aussi facile à connaître par des outils traditionnels. Cela implique le développement des méthodes et des approches pour la gestion et la manipulation de ces données qui inclut le stockage, le partage, l'analyse et la visualisation ou la mise à jour des données. Et surtout, des formalismes approprient pour la conceptualisation, la modélisation et la représentation des connaissances ainsi que des mécanismes personnalisés pour l'interopérabilité et la liaison de données entre les

différents ensembles dynamiques du system pour gérer la consommation d'énergie dans les villes intelligentes. Ces exigences sont pertinentes pour développer l'approche proposée et elles doivent être prises en considération lors d'élaboration de chaque outil.

Contribution

La contribution de cette étude se concentre sur les villes intelligentes comme exemple contemporain. Nous proposons une solution intelligente pour faire face aux défis urbains dans la vie réelle des citoyens. La réduction de la consommation d'énergie est l'un des domaines les plus pertinents pour le développement et la durabilité des villes intelligentes. Elle vise à permettre une stratégie efficace pour économiser l'énergie et réduire les factures de consommation électrique pour les résidents (Ejaz et al.,2017).

Dans ce contexte, nous proposons une architecture de diverses technologies coopérant ensemble pour fournir une optimisation énergétique efficace. Non seulement pour les maisons indépendamment, mais aussi pour la ville entière. L'architecture proposée est essentiellement basée sur l'ontologie des maisons intelligents (Onto-SB), qui est un outil puissant pour représenter les connaissances du domaine et fournit un cadre structurel pour organiser les données sur les maisons intelligents (Degha et al., 2019). L'Onto-SB offrir un cadre qui permet le raisonnement en représentant formellement les connaissances du domaine. En outre, nous utilisons le système multi-agents (SMA) pour garantir un rôle fiable, qui fournit une coopération efficace et l'autonomie nécessaire pour gérer les énormes données générées et échangées entre les différents acteurs du système (Ma et al., 2019). L'utilisation des technologies Big Data assure un mécanisme fiable pour le stockage et le traitement des données. Par conséquent, l'approche proposée réussit à réduire la consommation d'énergie et à garantir le confort des habitants dans le contexte d'une durabilité environnementale intelligente.

Organisation de la thèse

Cette thèse est organisée en cinq chapitres. En commençant par une introduction générale dont on a présenté le contexte du travail, la problématique et la motivation.

Premier chapitre, vise à introduire notre thèse par un état de l'art du Big Data qui présente les concepts de bases liés à ce domaine. Les caractéristiques, les domaines et les technologies utilisées pour surmonter sa complexité.

Introduction Générale

Le deuxième chapitre est consacré à la technologie de Web sémantique et les ontologies. Ce chapitre présente les différents éléments, les langages, les éditeurs d'ontologie, le processus de construction et la raison pour laquelle l'ontologie est utilisée.

Le troisième chapitre est un survol sur les travaux de l'intégration de la sémantique dans le Big Data. Une explication générale pour le processus d'intégration, les approches existantes, les avantages et les défis rencontrés dans ce contexte. Ce chapitre est conclu par une étude comparative de différents travaux d'intégration à travers une variété des domaines.

Une vue conceptuelle de système est décrite dans le quatrième chapitre, ce chapitre présente notre contribution qui est une proposition d'un système de gestion de l'énergie dans les villes intelligentes qui se base sur l'intégration de l'ontologie avec l'utilisation de la technologie Big Data et le système multi agent.

Le dernier chapitre présente les résultats obtenus au cours des expérimentations effectués pour évaluer l'approche proposée, une comparaison avec des autres travaux est donnée à la fin de ce chapitre. Nous terminons notre thèse par une conclusion générale et des perspectives possibles à ce travail.

Chapitre I

Etat de l'art sur le Big Data et le système multi agent

I.1 Introduction

Dans ce chapitre nous allons définir la notion de Big Data tout en citons leurs caractéristiques qui présentent les aspects fondamentaux. Un bref aperçu sur l'histoire d'évolution de Big Data est important pour mieux comprendre les différentes générations qui ont conduit à leur essor. Ensuite, nous allons illustrer les domaines principaux qui sont émergés dans le paradigme de Big Data. A la fin de ce chapitre on abordera les technologies employées pour exploiter et manipuler le Big Data efficacement.

I.2 C'est quoi le Big Data ?

Dans la littérature, le terme Big Data a eu plusieurs définitions, selon Davis et Patterson "Les Big Data sont des données trop grandes pour être manipulées et analysé par des protocoles de base de données traditionnels tels que SQL ", ce point de vue est partagé avec (Emani et al., 2015), (Zikopoulos and Eaton, 2011) et autres. Dans le même sens *Manyika et al* a défini les Big Data comme « un ensemble de données dont la taille dépasse la capacité des outils logiciels de base de données typiques pour capturer, stocker, gérer et analyser » (Manyika et al., 2011). Ces deux groupes d'auteurs se convenir que seulement la taille de données est le facteur qui caractérise les Big Data. *Edd Dumbill* montre l'aspect multidimensionnel du Big Data lorsqu'il a ajouté que les données sont trop grandes, se déplacent trop vite, ou ne correspondent pas les restrictions de architectures de base de données (Emani et al., 2015). À travers cela, nous comprenons clairement que nous devrions ajouter d'autres caractéristiques afin qu'un gros volume de données soit considéré comme Big Data. Les auteurs (Emani et al., 2015) et (Zikopoulos and Eaton, 2011) utilisent les 3Vs (Volume, Variété et Vitesse) pour caractériser les Big Data. Par ailleurs, *Lomotey* a décrit le model 5Vs (Volume, Variété, Vitesse, Valeur et la Véracité) illustré dans la figure 1, qu'est une extension du model précédent du 3Vs (Rodríguez-mazahua et al., 2016). autres auteurs (Emani et al., 2015) et institutions comme IEEE s'appuient en plus de la valeur et la véracité sur la visualisation. Ce dernier V montre leur importance par les nouveaux outils utilisés à la compréhension des données et l'analyser des résultats. Pour bien saisir ces Vs, on explique les caractéristiques de Big Data :

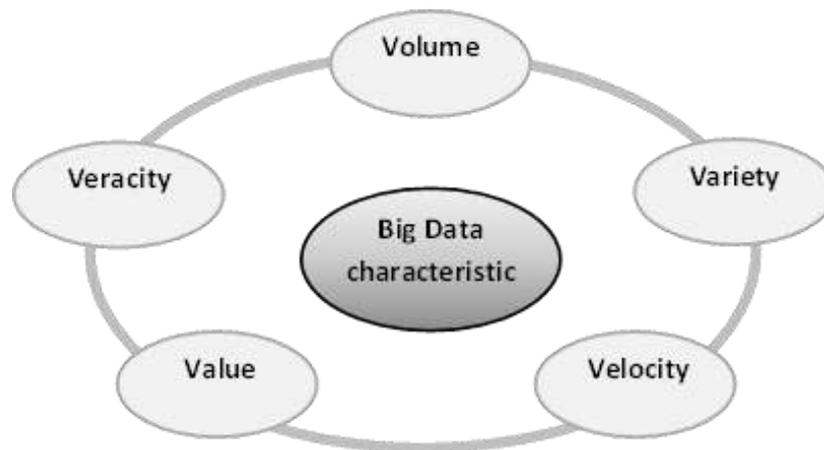


Figure 1 Caractéristiques de Big Data (Rodríguez-mazahua et al., 2016).

I.2.1 Volume (Données dans les supports de stockage)

Il s'agit d'une grande quantité de données générée et collectée par des entreprises à une échelle de pétaoctets et même de téraoctets, ces données de différents types continuent à inonder quotidiennement les bases de données de ces entreprises. Par exemple, Twitter transforme 12 téraoctets de tweets par jour en raison d'améliorer l'analyse de sentiment (Hadi et al., 2015), les téléchargements des vidéos à YouTube dépassent 72 heures chaque minute (Labrinidis and Jagadish, 2012). Les organisations américaines de soins de santé ont généré 150 Exaoctets de données sur les soins de santé en 2013 (Panahiazar et al., 2014), les capteurs placés à travers des milliards d'appareils collectent des données de différents types (*Gartner* estime que 6.4 milliard d'objets connectés utilisés en 2016).

Un autre exemple, un Airbus A380 possède environ de 25 000 capteurs qui génèrent une énorme quantité de données au cours d'un seul vol. Ce défi oblige les chercheurs et les entreprises à trouver de moyens et outils puissants et efficace pour pouvoir stocker ces immenses données dans des infrastructures suffisantes (en tenant compte de réplication de données pour assurer la disponibilité et la fiabilité) et les manipuler efficacement (temps de traitement raisonnable). La figure 2 montre l'accroissement des données entre 2005-2020 (Jeong and Ghani, 2014).

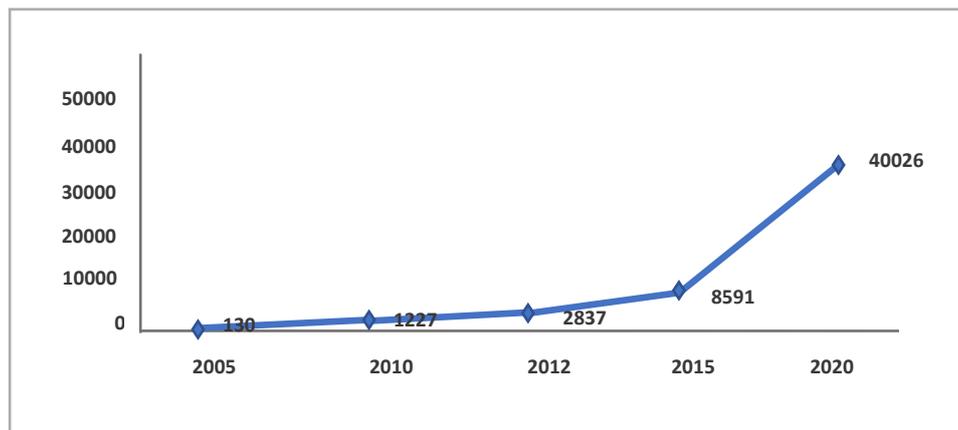


Figure 2 Croissance des données par an (exaocets) (Jeong and Ghani, 2014).

I.2.2 Variété (Données sous plusieurs formes)

Les différentes sources généralement hétérogènes génèrent des différents types de format de données, ces données sont classées sous trois formats, structurés (les données des bases relationnelles), semi structurés (web logs, email, flux de media sociaux, etc.) et non structurés (photo, audio, vidéo, capteurs, etc.) (Jeong and Ghani, 2014), ce type de données est le plus générés par rapport aux données structurées qui a conduit au développement des nouvelles techniques capables de l'analyser efficacement.

Les Big Data se composent avec cette variété des données qui sont en plus locales ou distants, public ou privé, complet ou non complets...etc. (Oussous et al., 2017), donc les données de Big Data sont rarement présentées sous un format standard et elles ne sont pas toujours prêtes pour le traitement sans passé à une étape de standardisation ou d'intégration (Abbes and Gargouri, 2018).

I.2.3 Vitesse (données en mouvement)

Cela signifie la rapidité de génération de données en temps réel à partir des milliards des machines connectées, *Gartner* (2015) a prévu que 6,4 milliards d'appareils connectés peuvent être utilisé dans le monde entier en 2016 et que ce nombre atteindra 20,8 milliards d'ici 2020 (Lee, 2017). Les données produites par des diverses sources changent et évoluent continuellement ce qui posent un défi majeur pour la recherche, le traitement, le partage et l'analyse du contenu en temps réel (Chen et al., 2014) par des périphériques et systèmes traditionnels qui ne sont pas capables à gérer ces données. Au début, les

entreprises analysaient les données avec de systèmes de traitement par lots selon la nature lente et coûteuse du traitement des données. À mesure que la vitesse de génération et de traitement des données augmentait, le traitement en temps réel devenu une norme pour les applications informatiques, dans la réalité, certaines activités sont très importantes et nécessitent des réponses immédiates, c'est pourquoi un traitement rapide maximise la valeur de l'information et l'efficacité de l'application (Lee, 2017).

Pour les systèmes critiques tels que la détection de la fraude, les flux de données doivent être analysés et exploités en temps réel afin de découvrir les actes illicites et les empêcher (par exemple, scruter 5 millions d'événements commerciaux créés chaque jour pour identifier la fraude potentielle) (Oussous et al., 2017). Ceci, donne une motivation et impose une nécessité urgente pour les chercheurs dans le domaine de Big Data pour construire l'infrastructure et la base de compétences nécessaires pour réagir rapidement à cet défi et soit plus agile pour réagir en temps opportun (Rao et al., 2015).

I.2.4 Valeur (la valeur des données extraites)

Cette caractéristique est introduite par Oracle comme une dimension supplémentaire au 3Vs du Big Data (Lee, 2017). Les données auront une grande valeur après le processus de traitement et analyse qui est considéré comme un objectif de la technologie Big Data (d'après IDC), pour cette raison les architectures de Big Data sont conçues pour extraire économiquement les données pertinentes à partir de très gros volumes d'une grande variété de données (Jeong and Ghani, 2014), en permettant la capture, la découverte et / ou l'analyse à haute , par conséquent l'importance apporté par le Big Data a permis les entreprises d'augmenter les revenus, réduire les coûts d'exploitation et mieux servir les clients, avec un coût d'investissement raisonnable (Saggi and Jain, 2018).

A titre d'exemple les Big Data peuvent améliorer l'efficacité des opérations gouvernementales, de sorte que les économies développées en Europe pourraient économiser plus de 100 milliards d'euros (ce qui exclut l'effet de réduction des fraudes, des erreurs et des différences fiscales. La figure 3 montre le volume de Big Data et l'impact sur le plan financière (Chen et al., 2014).

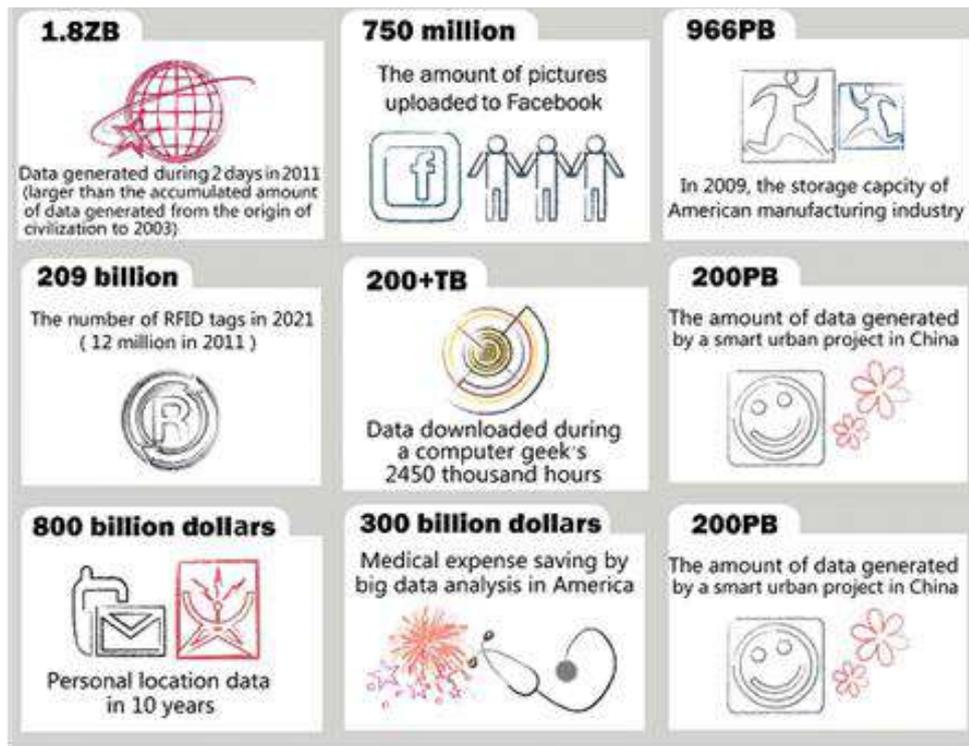


Figure 3 Big Data en constante évolution (Chen et al., 2014).

I.2.5 Véracité (la qualité ou la fiabilité des données)

Est suggéré par (IBM et Microsoft) comme une autre dimension permettant de mesurer la fiabilité des données proviennent de sources différentes (Yaqoob et al., 2016), donc la véracité désigne le degré de confiance accordé à un leader aux données traités et analysés pour prendre une décision, autrement dit c'est la conformité à la vérité ou au fait, la précision et la certitude ajouté aux données du Big Data (Emani et al., 2015). L'incertitude et le manque de fiabilité surviennent en raison de l'incomplétude, de l'imprécision, de la latence, de l'incohérence et de la subjectivité des données.

Les gestionnaires ne font pas confiance aux données lorsque les problèmes de véracité sont courants. En raison de la véracité, les résultats dérivés de Big Data ne peuvent pas être prouvés ; mais ils peuvent être assignés une probabilité désigne le degré de confiance donné à un décideur (Hadi et al., 2015). Des outils et des techniques statistiques ont été développés pour remédier ce manque de fiabilité des données avec des niveaux de confiance ou des intervalles de confiance spécifiés (Lee, 2017).

Autres aspects du Big Data sont cités dans la littérature et jugés aussi important comme la variabilité des données que signifie le changement du sens selon le contexte, la visualisation qui se réfère à la façon de représenter les données traitées d'une manière facilement lisible et interprétable, l'aspect de la sécurité et la confidentialité de données privés et stratégiques sont aussi un objet de recherche très exigeant.

Pour un traitement efficace et fiable des Big Data, les chercheurs doivent évaluer le critère de valeur par rapport au volume, capturé et traité la variété et à la véracité des données alors qu'elles sont encore en mouvement (vitesse). (Yaqoob et al., 2016) recommande de scientifiques qu'ils doivent attaquer conjointement tous les critères et fonctionnalités du Big Data sans se pencher vers des critères au détriment de l'autres, sinon les Big Data n'aboutit plus les ambitions attendues

Avec une autre perspective et pour une vision plus claire, on peut catégoriser les caractéristiques du Big Data selon les classes suivantes : sources de données, format de contenu, stockage de données, stade (étape) de données et traitement de données (Özköse et al., 2015).

- ✓ Sources de données : Web et réseau social, machines, capteurs, transactions et IoT, etc.
- ✓ Format de contenu : structuré, semi-structuré et non structuré.
- ✓ Stockage de données : Document orienté, orienté colonne, graphe basé sur une clé-valeur, etc.
- ✓ Stade de données : nettoyage, normalisation et transformation.
- ✓ Traitement de Données : Lot (patch) et Temps Réel (streaming).

I.3 Historique et développement du Big Data

Bien que l'émergence de Big Data c'est apparu que ces dernières années, mais le processus de collecte et de stockage des données se revient aux années 1950 avec l'utilisation de premiers ordinateurs commerciaux vendus. De cette période jusqu'aux années 1990, le progrès des données était lent en raison du coût élevé des ordinateurs et les supports de stockage, en plus les sources de génération des données étaient limitées. Les données dans cette période étaient structurées puisqu'elles étaient destinées pour servir des systèmes d'information opérationnels. Néanmoins, il y avait des technologies qui ont montrés une capacité considérable pour répondre aux besoins de traitement et stockage progressif notamment dans les années 1980 quand les bases de données

parallèles sont apparues, les architectures des systèmes sont basées sur les clusters dont chaque machine possède son propre processeur et disc de stockage (Chen et al., 2014). Au début des années 1990 le World Wide Web est apparu et conduit à un développement explosif des données qui ont impliqué une assistance efficace de traitement et d'analyse de ces masses croissantes des données (Lee, 2017). Ce processus a évolué en trois étapes principales :

I.3.1 Première génération : Big Data 1.0 (1994-2004)

Cette étape a connu l'avènement de commerce électronique, les grandes entreprises ont été les acteurs principaux du contenu web dont le développement des techniques d'exploration puissantes pour explorer et analyser les activités en ligne des utilisateurs est exigés. Pour relever le défi de la gestion de Big Data, Google a créé des modèles de programmation GFS et MapReduce pour assurer la gestion et l'analyse des données à l'échelle Internet (Chen et al., 2014). L'exploration de contenu Web était divisée en trois types différents : l'utilisation, la structure Web et de contenu Web. Qui correspond respectivement : l'application de techniques d'exploration de données pour découvrir les habitudes d'utilisation des internautes en ligne, le processus d'analyse de la structure d'un site Web ou d'une page Web et extraire des informations utiles du contenu des pages Web.

Pour explorer le contenu des pages web (texte, image, audio, vidéo. etc.) des techniques telles que la recherche d'information et le traitement de langage naturel ont été introduit pour extraire les informations souhaitées (Lee, 2017). Les techniques d'extraction de contenu Web étaient limitées pendant l'ère du Big Data 1.0 et nécessite une amélioration pour faire face au volume explosive de données.

I.3.2 Deuxième génération : Big Data 2.0 (2005-2014)

Big Data 2.0 est engendré grâce au Web 2.0 et les media sociaux, ce qui permet aux internautes d'interagir avec des sites web et partager leurs contenus. La majorité des grandes entreprises (EMC, Oracle, IBM, Microsoft, Google, Amazon, et Facebook, etc.) ont lancé leurs propres projets de Big Data. IBM depuis 2005 a investi plus de 16 milliards de dollars sur projets liées au Big Data (Chen et al., 2014). Dans cette période les réseaux sociaux deviennent très populaires pour le grand public et la majorité des entreprises, cette interaction permanente et intéressante a obligé les principaux acteurs dans ce domaine de trouver des outils puissants capable de stocker le volume

exponentiellement généré et d'analyser l'activité et le comportement des internautes afin de répondre aux besoins des clients et cibler les catégories intéressantes.

Contrairement à l'analyse du Web qui est utilisé généralement pour des données structurées, l'analyse de réseaux sociaux utilisent des données non structurées et proche au langage naturel qui se concentre sur la compréhension des sentiments et l'opinion des abonnés, ces outils d'analyse ont utilisé le cloud basé service avec un coût flexible. Pour l'analyse de sentiment et l'interprétation des opinions la méthode basée sur le lexique et la machine Learning sont largement utilisés, ce processus sert à mesurer la structure des réseaux sociaux, les relations, les nœuds et autres propriétés pour modéliser le développement et la dynamité de ces réseaux comme Facebook, Twitter et LinkedIn, ces derniers fournissent un point d'accès central et apportent la structure dans le processus de partage d'informations personnelles en ligne (Lee, 2017).

I.3.3 Troisième génération : Big Data 3.0 (2015-...)

Big Data 3.0 englobe les données de deux générations précédentes. Les applications d'internet des objets (IoT) c'est les contributeurs essentiels avec l'énorme quantité des données générées sous forme d'images, audio et vidéo. Ces données ont des caractéristiques différentes de celles du Big Data général en raison des différents types de données recueillies, tel que le bruit, et haute redondance (Chen et al., 2014). L'environnement technologique créé par l'IoT est alimenté par les données générées depuis les dispositifs et les capteurs ayant des identifiants uniques qui ont une capacité de partage et collaboration sur les réseaux du web d'une manière autonome. Le progrès massif de l'IoT ces dernières années et l'émergence dans plusieurs domaines a donné l'avantage aux appareils connectés et les capteurs pour se marquer devant les réseaux sociaux et le commerce électronique comme des premières sources de Big Data. *GE* installe des capteurs lisant les données déportées en temps réel depuis des applications employées pour le secteur de l'aviation et la santé.

La plupart des applications IoT envoient les données collectées par des capteurs installés sur place et l'analyse de ces données est effectuée en diffusion continue (streaming) (Wira et al., 2016). Contrairement à l'analyse par lot (batch) des données stockées utilisé par les réseaux sociaux. L'analyse en streaming nécessite un traitement en temps réel pour découvrir les modèles d'intérêt en cours de la génération et collecte des données ainsi pour prédire les futurs événements qui peuvent se produire. Par exemple,

dans le domaine de la santé, l'analyse en continu des données en provenance des capteurs peut surveiller et interpréter les changements physiologiques et comportementaux des patients et alerter dans le temps opportun les soignants sur les besoins médicaux urgents des patients (Lee, 2017).

Le processus ou le cycle de vie des données dans le Big Data est généralement le suivant (Özköse et al., 2015):

1. Gestion des données qui contient les étapes (acquisition et enregistrement, extraction, nettoyage et annotation, l'intégration et l'agrégation et représentation).
2. Analyse de données : (Modélisation et analyse, Interprétation).

I.4 Domaines de Big Data

Dans les dernières années, le Big Data a connu une vaste utilisation dans plusieurs domaines (Chen et al., 2012), à cause de l'explosion du volume de données générées et la variété des domaines utilisant cette nouvelle technologie. L'usage de Big Data dans ces secteurs devient une impérative pour répondre aux besoins excessifs de clients et de marché de données. Des nombreux domaines sont émergés dans ce nouveau paradigme (figure 4), nous citons les suivants (Özköse et al., 2015):

- ✓ Domaine médical.
- ✓ Commerce électronique.
- ✓ IoT.
- ✓ Réseaux sociaux et les services en ligne.
- ✓ Industrie automatisée et Haute technologie.
- ✓ Secteur des télécommunications.
- ✓ Éducation et recherche.
- ✓ Secteur des voyages et des transports.
- ✓ Services financiers.
- ✓ Services publics.
- ✓ Défense et sécurité.
- ✓ E-gouvernement et politique.
- ✓ Météo.



Figure 4 Les différentes sources de Big Data (Bansal and Kagemann, 2015).

I.5 Analyse de Big Data

Analyse de Big Data est défini par l'utilisation de techniques analytiques avancées sur les Big Data. Actuellement, les outils et les techniques disponibles peuvent assurer la collecte et l'analyse des données volumineuses. Pour effectuer l'analyse des conditions préalables sont nécessaires. Les outils et les capacités de stockage pour manipuler et gérer les gros volumes de données, en raison de sa taille, les Big Data offrent de grands échantillons statistiques et améliorent les résultats des expériences (Emani et al., 2015). Enfin, les entreprises et les gouvernements ont clairement valorisé les avantages pour investir dans les secteurs du Big Data. En cette raison, de nombreuses techniques sont utilisées pour l'analyse des Big Data :

I.5.1 Analyse du texte

Elle est utilisée pour la modélisation de sujets, la réponse aux questions et la recherche d'informations à partir des E-mail, blogs (Bello-orgaz et al., 2016), forums directs, cette méthode d'analyse implique l'analyse statistique et la machine Learning pour extraire des informations significatives (Tsai et al., 2015) et (Özköse et al., 2015) qui offre à la

machine une capacité d'évolution des comportements basé sur des données empiriques (Yaqoob et al., 2016).

I.5.2 Analyse audio

Les données audio ont un format non structuré, l'objectif de l'analyse audio est de tirer les informations désirées à partir ces masses de données non structurées (Saggi and Jain, 2018). Cette tâche est réalisée à l'aide des outils de reconnaissance vocale automatique, qui est généralement utilisé dans les centres d'écoute pour analyser et comprendre les besoins des clients afin de fournir un service adéquat aux ambitions des clientèles et augmenter la rentabilité de l'entreprises (Lee, 2017).

I.5.3 Analyse vidéo

Le format des vidéos ne représente pas le seul défi pour le traitement et l'analyse de ce type de données, la taille aussi des données vidéos qui ne cesse pas à se redoubler surtout avec la haute qualité (HD) et la résolution offerte par les cameras modernes engendre une difficulté majeure pour s'occuper et de trouver les outils efficaces qui prendra en charge le processus d'analyse d'une façon similaire aux autres types des données. Les sites de partage des vidéos sont les principaux contributeurs responsables à l'inondation de web. Prenons YouTube est alimenté par des centaines des heures vidéo chaque minute (Lee, 2017).

Des diverses techniques développées pour le traitement en temps réel ainsi que les vidéos préenregistrées utilisés généralement pour la recherche et l'indexation automatique de contenu multimédia, qui est effectué pour faciliter la recherche et la récupération des vidéos par l'association des techniques d'analyse audio et texte participant à la tâche d'indexation vidéo. Parmi les domaines d'application d'analyse vidéo on trouve le marketing et la gestion des opérations (contrôle de criminalité, frontière. etc.) (Özköse et al., 2015).

I.5.4 Analyse des réseaux sociaux

L'analyse des réseaux sociaux (SNA) est apparue avec la naissance du Web 2.0, il s'agit d'analyser les données structurés et non structurés à la fois et son objectif est de voir les relations sociales dans la théorie des réseaux sociaux (Yaqoob et al., 2016). Plusieurs catégories des media sociaux sont existés tels que les réseaux sociaux (Facebook, LinkedIn), Blogs (BlogSpot, WordPress), Microblogs (Twitter, Tumblr), Partage de

médias (Instagram, YouTube), Wiki (Wikipedia, Wikihow), etc. (Özköse et al., 2015). L'analyse des réseaux sociaux implique plusieurs disciplines, comme la psychologie, la sociologie, l'anthropologie, l'informatique, les mathématiques et l'économie. Les utilisateurs des réseaux sociaux avec la génération de leurs données qui sont sous forme des sentiments, avis, photos, vidéos, etc. et l'interaction avec autres entités web qui sont des personnes, organisations et produits sont les sources essentielles des données qui sont structurés et non structurés (Quboa and Mehandjiev, 2017).

Les techniques connues dans ce domaine sont : détection de communauté, analyse d'influence sociale, spécification de prévision de lien. Le marketing est la principale application de l'analyse des médias sociaux car il bénéficie de la généralisée et croissante des médias sociaux par les consommateurs du monde entier (Lee, 2017).

I.5.5 Analyse prédictive

Elle est basée principalement sur les méthodes statistiques, leur objectif est de prévoir les futures acte et résultats en se basant sur l'historique et les données actuelles. L'analyse prédictive cherche à découvrir les modèles et trouver les relations entre les données (Lee, 2017). Le moyen mobile, la régression linéaire et l'apprentissage automatique (réseaux de neurones) sont les principales techniques utilisées dans l'analyse prédictive. Elle est employée dans plusieurs secteurs : la commerce électronique, prévision météo, propagation d'épidémie, centre nucléaire, campagne électorale, etc. (Tsai et al., 2015), (Özköse et al., 2015).

I.6 Technologies de Big Data

L'immense quantité des données stockée et générée exponentiellement ainsi ces caractéristiques qui se diffèrent largement aux bases de données traditionnelles exigent des nouvelles technologies informatiques capable d'acquérir, stocker, manipuler et analyser les Big Data avec un coût abordable et un temps de traitement très raisonnable, les entreprises sont désormais davantage capable à gérer l'énorme volume de données auparavant traitées à l'aide de superordinateurs coûteux avec des outils beaucoup moins chers (Oussous et al., 2017). Les nouvelles techniques de l'informatique distribuée deviennent très dominantes pour surmonter les difficultés engendrées par le gros volume et la distribution des données.

L'importance du Big Data devient primordiale pour les grandes entreprises comme Yahoo!, Google, Facebook, Twitter et YouTube. L'usage des nouvelles technologies est inévitable pour gérer et analyser ces données dans un temps quasi réel qui apparaît comme un facteur crucial pour bénéficier de données provenant de l'utilisateur. Les solutions résultantes ont affecté le marché de gestion de données. Plusieurs solutions sont développées comme NoSQL data base, *R* et Hadoop. Le projet Apache Hadoop est un célèbre outil (créé par Doug Cutting en 2009), Hadoop est une open source écrit en Java qui se compose par deux modules principaux : System de Fichiers Distribués Hadoop Distributed File System (HDFS) et MapReduce (Shvachko et al., 2010).

Les innovations Hadoop, MapReduce et Big Table sont devenues un nouveau paradigme pour la gestion de données (Emani et al., 2015). Ces technologies offrent aux entreprises la possibilité de résoudre l'un des problèmes les plus fondamentaux dans les Big Data, à savoir la capacité de traitement des quantités massives de données de manière efficace, rentable et rapidement à moindre coût, indépendamment de sa structure (Hadi et al., 2015).

I.6.1 Hadoop

Parmi les technologies prometteuses du Big Data, on trouve l'Apache Hadoop qui est un Framework libre (open source) développé en Java, inspiré essentiellement par publication de MapReduce, Google FS et Big Table de Google, il est initialement développé par Yahoo!. **Doug Cutting** a créé Hadoop en 2009 qui a fait partie de leur projet de la fondation logicielle Apache. L'objectif principal du Hadoop est de remédier le déficit enregistré par les outils traditionnels de traitement et d'analyse de données massives. Hadoop offre des nombreux privilèges au Big Data grâce à ces composants fondamentaux et son écosystème qui possède une multitude de logiciels annexés. La solution Hadoop se caractérise par l'idée de parallélisation distribuée de traitement des données entre les nœuds de calcul qui accélère l'exécution des tâches et diminue la latence à travers des serveurs beaucoup moins coûteux, il peut également offrir la possibilité de traitement de tous les types de données (structuré, semi structuré, non structuré) (Kim et al., 2014). Yahoo! et autres entreprises optent le projet Hadoop pour qu'il soit la base de ses architectures informatiques qui gère leurs activités vitales.

Hadoop assure l'exécution des applications traitées par MapReduce sur un grand nombre de clusters qui atteindra les milliers et des pétaoctets de données, dans un laps de

temps très court (Hadi et al., 2015). Par exemples pour interroger des téraoctets de données par Hadoop ça nécessite que quelques seconds par contre un SIEM classique prend plus de 20 minutes (Oussous et al., 2017). Les grandes compagnies productrices des Big Data ne craignent plus du stockage et de la fiabilité de leurs données puisque HDFS de Hadoop a pris en charge la complexité de la tâche de l'hébergement de données, que ce soit sa taille et offre l'avantage pour l'évolutivité. Tant que les pannes sont fréquentes pour une raison ou une autre surtout sur le réseau web, Hadoop a trouvé la solution pour tous ces composants par la réplication de données qui garantit un degré de fiabilité et tolérance aux pannes quasi parfait et permettre les organisations à valoriser les données qui ont été considérées depuis peu de temps comme inutiles.

Autre avantage fourni par Hadoop par rapport aux technologies classique, l'exécution de données distantes ne sont pas copies en mémoire mais elles sont exécutées où elles sont stockées ce qui soulage énormément le réseau et les serveurs d'une charge de communication significative. En plus de HDFS et MapReduce qui constituent le noyau de Hadoop, autres logiciels enrichissent son écosystème comme exemple : Apache Pig, Apache Hive, Apache HBase, Apache Phoenix, Apache Spark, Apache ZooKeeper, Cloudera Impala, Apache Flume, Apache Sqoop, Apache oozie, Apache Storm, etc. (Shvachko et al., 2010).

HDFS et MapReduce représentent les deux sous-composants principaux qui offriront la puissance du Hadoop, néanmoins, les utilisateurs ont le choix d'ajouter autres modules sur la plateforme selon les besoins et en fonction de leurs exigences demandées par l'application en question (capacité, performance, évolutivité, fiabilité, etc.) (Oussous et al., 2017). Puisque chaque projet correspond à une fonctionnalité bien précise, et chaque module est associé à sa propre communauté de développeurs selon le cycle de développement de chaque application.

Pour avoir une idée plus élargie sur l'écosystème de Hadoop, on va expliquer en bref ses composants ainsi on les classe selon le rôle de chaque ensemble (stockage, traitement, gestion) (Chen et al., 2012).

Points forts de Hadoop

- ✓ Capacité de stockage et de traitement de données massive de différents types qui ne cesse pas d'agrandir exponentiellement venant généralement de media sociaux et IoT.

- ✓ Puissance de calcul grâce à l'architecture distribuée de calcul de Hadoop qui offre un traitement rapide des données volumineuses. Plus les nœuds de calcul sont nombreux, la puissance de traitement s'accroît.
- ✓ Tolérance aux pannes, avec la réplication de données et les multitudes des nœuds engagés pour le traitement le système bascule automatiquement en cas de panne ou indisponibilité des nœuds.
- ✓ Flexibilité, la possibilité de stocker sans limitation de données structurés ou non structurés ensuite l'application de différents types de traitement.
- ✓ Coût bas de stockage et de traitement garanti par le Framework open-source.
- ✓ Scalabilité de système par l'ajout des nœuds de stockage et de traitement facilement sans affecter le système.

I.6.2 HDFS

C'est un système de fichiers distribués conçu pour enregistrer une grande quantité de données à travers une multitude des nœuds réparties apparaît à l'utilisateur comme un seul disque dur, il fournit un stockage de données fiable et évolutif. Son rôle est pour couvrir une grande échelle de serveurs. La portabilité sur plusieurs plate-formes matérielles et logicielles hétérogènes est l'avantage principal de l'HDFS, qui réduit énormément la congestion de réseau et accroît la performance du système en mettant la fonctionnalité de calcul à proximité de l'endroit de stockage (Oussous et al., 2017). Par exemple, un seul cluster peut employer 4500 serveurs pour stocker 200 pétaoctets extensible qui contient environ un milliard de fichiers et de blocs de données.

L'architecture HDFS est de type maître-esclave respectivement représenté par (NameNode, DataNode), dont le nœud de noms est responsable pour la gestion de l'espace de noms, la hiérarchie de fichiers et les métadonnées de chaque fichier (figure 5) (Chen et al., 2012). Il localise les blocs de données enregistrés et dispatché dans le cluster. Le nœud de noms a une deuxième instance responsable de la gestion, la mise à jour de système de fichiers et assure le fonctionnement du cluster en cas d'une panne du NameNode original. L'aspect distribué et évolutif de l'HDFS assure une grande fiabilité et tolérance aux pannes et permet de fonctionner parfaitement avec une grande variété d'applications d'accéder aux données simultanément, en coordination avec YARN, la

combinaison des ressources de stockage accroître la linéarité avec un coût très acceptable sur tous les niveaux de stockage (Emani et al., 2015).

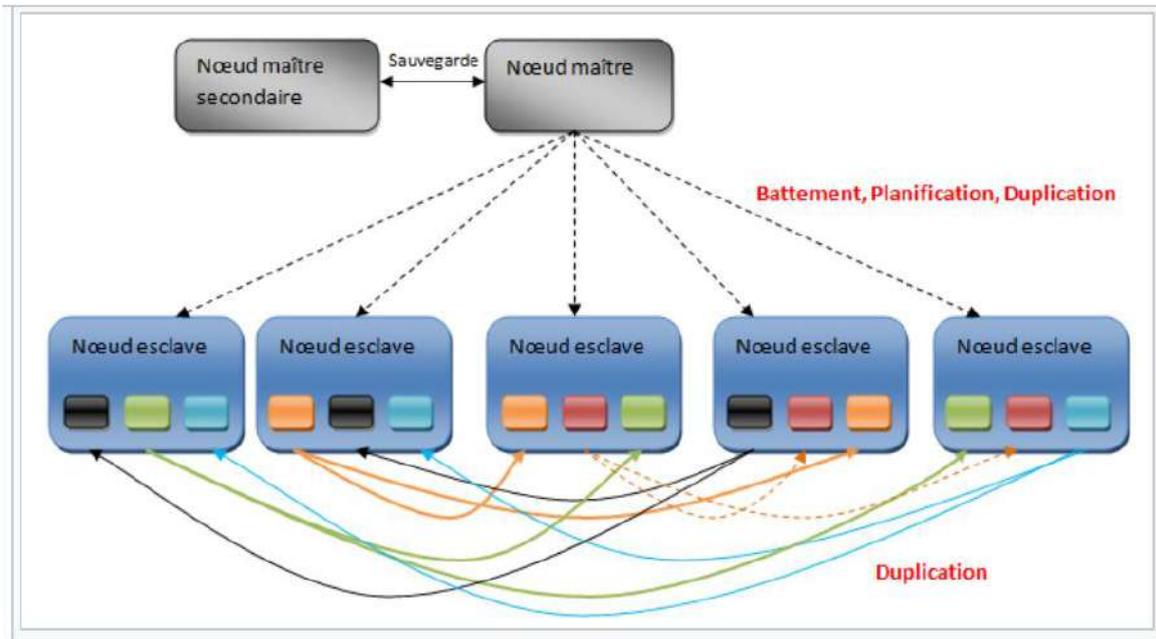


Figure 5 Architecture HDFS de Hadoop (Chen et al., 2012).

I.6.3 MapReduce

Est un moteur de traitement de données qui représente la deuxième composante principale de Hadoop, l'idée de MapReduce est introduite par des ingénieurs de Google en 2004 pour la première fois (Chen et al., 2012) pour résoudre le problème de création d'index de recherche Web. Elle constitue le module de traitement de données (structurées ou non structurées) stockées dans le système de fichiers distribués de Hadoop. L'efficacité et la rapidité apporté par le MapReduce se figure dans l'aspect de traitement parallèle des lots de données potentiellement très volumineuses, typiquement supérieures en taille à des téraoctets à travers les nœuds de cluster.

MapReduce se repose sur une architecture de type Maître-esclave où le nœud maître dirige tous les nœuds esclaves qui les déteints (Emani et al., 2015). Le processus de traitement se divise en deux fonctions, Map et Reduce, la phase Map sert à analyser le problème et le découper en sous problèmes qui sont envoyés à des autres nœuds (même opération peut se faire récursivement), les sous problèmes sont ensuite traités par les différents nœuds de cluster à l'aide d'une fonction Map qui a un couple (clé, valeur). Dans la phase Reduce, les nœuds les plus bas renvoient leurs résultats au nœud parent. Celui-ci calcule un résultat partiel avec la fonction Reduce et le fait remonté à son tour au

son nœud parent jusqu'à la fin de processus. Le nœud d'origine établit le résultat final au problème en question par la fonction : $\text{Reduce}(\text{clé}, \text{liste}(\text{valeur})) \rightarrow \text{valeur}$ (Oussous et al., 2017), la figure 6 montre ce processus. MapReduce réduit considérablement le trafic de données à travers le réseau par le déplacement de processus de calcul vers les données sur HDFS qui offre une rapidité de traitement de Hadoop. Par exemple un problème qui prend quelques jours de traitement peut être résolu en quelques heures ou minutes. Néanmoins, MapReduce a des inconvénients tels que la difficulté de traitement des données en streaming et des nombreux algorithmes ne se traduisent pas facilement dans ses modèles.

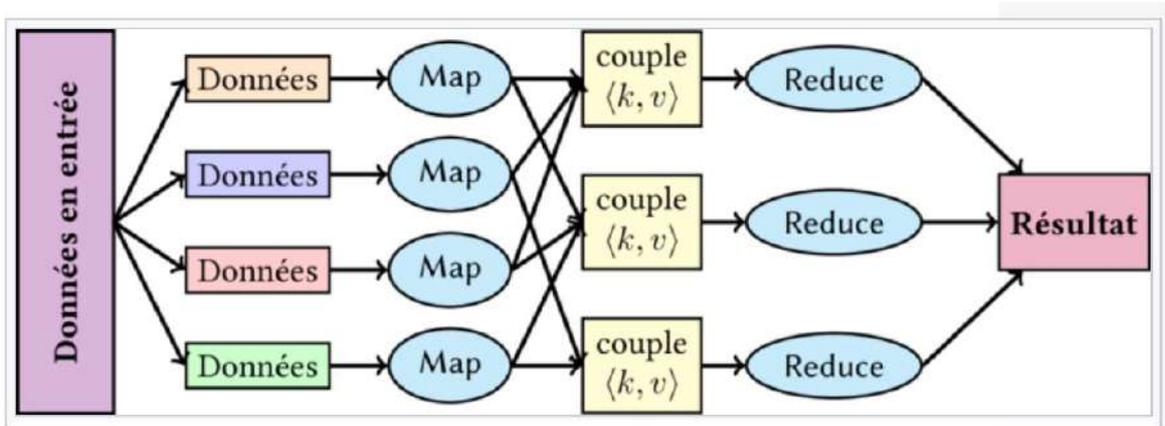


Figure 6 Schéma de fonctionnement de MapReduce (Oussous et al., 2017).

Hadoop a développé d'autres moteurs de traitement de données qui peuvent désormais fonctionner avec MapReduce par des différentes manières pour remédier les inconvénients rencontrés. YARN Hadoop est un exemple promoteur qui a apporté une efficacité et souplesse dans le paradigme de traitement dans l'écosystème Hadoop. Il permet de traiter les données en lot et en streaming, il offre une évolutivité, un parallélisme amélioré et une gestion avancée des ressources par rapport à MapReduce. L'architecture Hadoop a été modifiée pour intégrer YARN Resource Manager.

Autour de HDFS et MapReduce il y a des dizaines de projets qui ne peuvent être présentés en détail ici (Emani et al., 2015). Chaque module correspond à une fonctionnalité bien précise destinée à sa propre communauté de développeurs pour accomplir des tâches spécifiques. Dans la figure 7 les projets sont classés selon leurs capacités (stockage, traitement, gestion) (Chen et al., 2012). Plusieurs sociétés telles que Cloudera, Hortonworks et MapR proposent des distributions de Hadoop qui rassemblent un

certain nombre de ces projets. Des versions gratuites et professionnelles sont disponibles pour les praticiens et les professionnels.

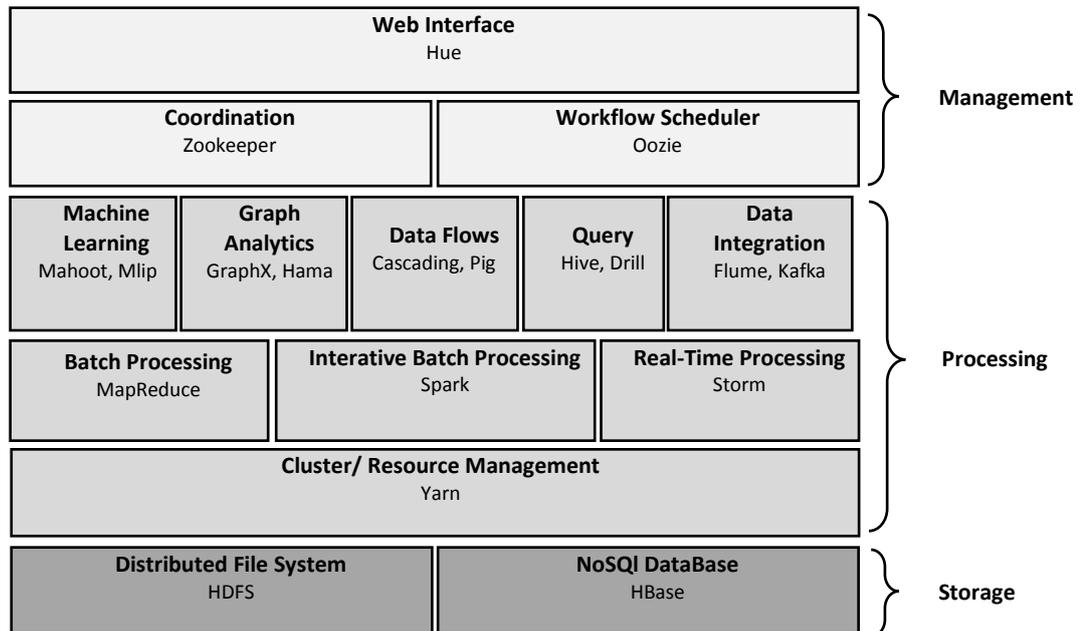


Figure 7 Hadoop Ecosystem (Chen et al., 2014).

I.7 Les systèmes multi-agents (SMA)

L'intelligence artificielle distribuée (IAD) est apparue comme une solution pour remédier les insuffisances constatées dans l'intelligence artificielle dans des nombreux domaines complexes. Cette notion s'intéresse à l'étude des connaissances et des raisonnements nécessaires pour la coordination d'actes entre les différents modules de traitements (Bourekache, 2014).

Les systèmes multi-agents (SMA) sont apparus comme un sous-domaine de l'IAD pour résoudre les contraintes que l'IA a loupé à résoudre. Les avantages d'utilisation des SMA sont liés à leur capacité d'aborder les problèmes complexes d'une manière distribuée et de proposer des solutions modulaires et robustes. Actuellement, l'emploi des SMA a pris une place de plus en plus importante en informatique surtout les disciplines qui s'attache aux comportements collectifs produits par les interactions de plusieurs entités autonomes et flexibles appelées agents (Hamida, 2014).

I.7.1 Le concept d'agent

Dans la littérature, il existe plusieurs définitions au concept d'agent. Les chercheurs se base généralement pour définir un agent plutôt par ses propriétés.

Selon Ferber " l'agent est une entité physique ou virtuelle qui est capable d'agir dans son environnement, pouvant communiquer avec d'autres agents, il est doté d'autonomie et possède des ressources propres, son comportement est la conséquence de ses objectifs, de sa perception, de ses représentations, de ses compétences et des communications qu'elle peut avoir avec les autres agents, il possède des compétences et offre des services" (Bourekache, 2014).

En effet, il existe plusieurs types d'agents à savoir : des agents réactifs, cognitifs et hybrides. Ces derniers peuvent être soit stationnaires ou mobiles. Les agents stationnaires exécutent leurs tâches au niveau du site qui les a créés. Tandis que les agents mobiles peuvent se déplacer au cours de l'exécution avec leur code et leurs propres données, ainsi qu'avec leur état d'exécution pour accéder aux données ou aux ressources.

Agent réactif

Ils se caractérisent par des comportements simples qui se résument à des réactions aux stimuli de l'environnement. Ces derniers sont capables d'actionner sur de groupes complexes et coordonnées. Ce type sont de petite taille et en grand nombre au sein de l'environnement. Ils ne demandent pas intelligents (Hamida, 2014).

Agent cognitif

Les agents cognitifs disposent d'une mémoire et possèdent une connaissance propre comprenant une représentation (partielle) de leur environnement, des autres agents, ainsi que de leur savoir-faire. Ce qui leur permet de gérer leurs interactions avec l'environnement et les autres agents. Ils se distinguent par leur faculté d'anticipation, la capacité de raisonner, l'analyse, la prévision, etc. Un point crucial dans la conception d'agents cognitifs est la planification de leurs actions.

Agent hybride

La possibilité de combiner les deux types d'agents précédents donne une architecture hybride. En cette architecture un agent est composé de modules qui gèrent

indépendamment la partie réflexe (réactive) et la partie réfléchie (cognitive) du comportement de l'agent.

I.7.2 Caractéristiques du Système multi-agents

Le passage du comportement individuel aux comportements collectifs est considéré comme un enrichissement de l'intelligence artificielle, d'où émergent de nouvelles propriétés et de nouveaux comportements.

Les SMA peuvent être introduits dans un système, un ensemble d'agents qui sont capables d'interagir et sont dotés de connaissances, d'intentions et de capacité d'évolution différente. Donc l'intérêt de l'agent apparaît au sein d'un univers d'agents dans lequel les agents communiquent pour résoudre ces problèmes complexes. Dans un SMA, l'agent a un comportement individuel ou collectif et il a ses objectifs à atteindre et les ressources et les compétences qu'il dispose. La puissance d'un SMA provient de comportement collectif des agents qui se traduit à partir de leurs interactions : coopération, négociation et coordination (Bourekache, 2014).

Les SMA sont à l'intersection de plusieurs domaines scientifiques: les systèmes distribués, les interfaces homme-machine, les bases de données et les bases de connaissances distribuées, les systèmes pour la compréhension du langage naturel, les protocoles de communication et les réseaux de télécommunication, la programmation orientée agents et le génie logiciel, la robotique cognitive et la coopération entre robots,... etc. (Coneglian et al, 2016).

Les perceptions permettent les agents d'acquérir des informations sur l'évolution de l'environnement, et le modifier par les actions. Les agents interagissent entre eux directement ou indirectement (Hamida, 2014).

Selon Ferber un SMA est un système composé des éléments suivants:

- ✓ E : un environnement, c'est-à-dire un espace disposant généralement d'une métrique.
- ✓ O: un ensemble d'objets : on peut associer une position E pour chaque objet à un moment donné.
- ✓ A : un ensemble d'agents : qui sont des objets particuliers.
- ✓ R : un ensemble de relations : qui unissent des objets et des agents entre eux.

- ✓ Op : un ensemble d'opérations : elles permettent aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O.

Dans un système ouvert, une organisation n'a pas forcément un objectif global et les agents peuvent être en concurrence, interaction, négociation, coopération, coordination ou en communication :

- ✓ L'Interaction : considérée comme la caractéristique principale d'un système multi-agents.
- ✓ La coopération : est la forme générale d'interaction.
- ✓ La négociation : est le processus d'améliorer les accords en réduisant les inconsistances et l'incertitude sur des points de vue communs ou des plans d'action.
- ✓ La coordination : est définie comme l'articulation des actions individuelles accomplies par chacun des agents de manière à ce que l'ensemble aboutisse à un tout cohérent et performant.
- ✓ La communication : est à la base des interactions et de l'organisation des agents.

En communiquant, les agents sont capables de coopérer, de négocier, de coordonner leurs actions ou de réaliser des tâches en commun (Hamida, 2014).

Afin de faciliter la communication et l'interopérabilité entre les agents, plusieurs langages de communication ont été développés à titre d'exemple :

- ✓ KQML (pour Knowledge Query and Manipulation Language).
- ✓ FIPA-ACL (FIPA Agent Communication Language) qui est une extension du langage KQML.

I.7.3 La plateforme de développement des SMA

Il existe plusieurs environnements de développement permettant de créer des applications SMA. On présente la plateforme la plus utilisée JADE (Bourekache, 2014).

Java Agent Development Environment (JADE)

JADE " Java Agent Development Environment" est une plateforme implémentée en Java qui permet l'implémentation des systèmes multi-agents et des applications conformes aux standards FIPA, notamment pour permettre l'interopérabilité entre agents. JADE permet

la création des agents et l'envoi des messages entre eux et autorise l'utilisation de n'importe quelle plate-forme et des outils associés, en particulier le fonctionnement dans un environnement ouvert et dynamique (Hamida, 2014).

JADE propose également des classes de bases pour des types de comportements spécifiques (CyclicBehaviour, ParallelBehaviour, ReceiverBehaviour, SenderBehaviour, SequentialBehaviour...). JADE utilise le langage de communication FIPA-ACL (Agent

Communication Language) et a implémenté le protocole Contract-Net qui a été normalisé par le FIPA.

I.7.4 Les raisons du choix du paradigme d'agent

L'importance de la technologie agent est très évidente et pertinente en matière de développement logiciel lorsque des critères comme la distance, la coopération entre entités distinctes ou l'intégration de logiciels existants sont à prendre en compte. Plusieurs Domaines comme les villes intelligentes peuvent bénéficier à la technologie agent (Anvari et al., 2017). En effet, ces domaines profitent des caractéristiques des systèmes Multi-Agents (SMA) afin de partager de la connaissance et assurer la coopération d'un ensemble d'agents et la coordination de leurs actions dans un environnement pour effectuer un but commun.

La caractéristique d'un système ouvert est provoquée par le changement dynamique de structure de ces systèmes qui sont souvent hétérogènes. En effet, les villes intelligentes se présentent comme une grande source d'informations hétérogènes et distribuée ce qui nous a encouragés à utiliser les SMA (Harmouch et al., 2019).

I.8 Conclusion

L'émergence de Big Data dans les différents domaines affiche clairement leur importance, ce qui conduit la plupart des entreprises à profiter de l'immense capacité et avantages fournis par cette nouvelle technologie. Dans ce chapitre nous avons vu un état de l'art sur la notion du Big Data, les caractéristiques, historique et les technologies utilisées pour la manipuler, ainsi un aperçu sur le système multi agent. Le chapitre suivant traite une notion très étroite avec le Big Data et devient une solution adéquate pour surmonter les défis générés par ce dernier. Le Web sémantique et les ontologies vont être expliqués et discuter leurs différents type avec le processus de construction.

Chapitre II

Le Web Sémantique et les Ontologies

II.1 Introduction

De nos jours, le Web devient une source d'information très riche et complexe par des milliards des pages Web, malheureusement la majorité d'entre elles ne sont pas compréhensible et exploitable que par l'homme, puisqu'elles ont une multitude de format non lisible par la machine qui la rendre très difficile à comprendre ou traiter avec des logiciels classiques. Cette réalité engendre qu'une grande partie de contenu Web reste inexploitable et par conséquence inutile. Pour remédier cette insuffisance et faire profiter le maximum de contenu Web, les chercheurs ont travaillé pour développer le Web sémantique où les ontologies qui visent à décrire la structure des informations (aspect syntaxique) et la signification de ces données (aspect sémantique) et intégrer une couche de connaissances aux systèmes qui lui offre la possibilité de traitement et d'exploitation les informations manipulées.

Les ontologies sont apparues dans les années 1990 pour la représentation des connaissances notamment dans le domaine de l'ingénierie de connaissance et l'intelligence artificielle pour aider la machine à exploiter ces connaissances. La technologie du Web sémantique a pour but de renforcer l'aspect représentatif des données dans le Web qui rendre la recherche, l'échange et l'exploitation de contenu assez facile (Bukhari et al., 2018). Selon *Goble*, ce processus est faisable par l'étiquetage (tagging) ou le balisage (marking up), et il est possible d'annoter le contenu du Web avec des informations supplémentaires appelées "métadonnées" qui peuvent être traitées par des agents logiciels (Bourekache et al., 2016).

Dans ce chapitre, nous allons décrire le Web sémantique et les ontologies. L'architecture du Web et les composants de l'ontologie l'importance et les types d'ontologie est discuté par la suite. Les critères pour construire une ontologie est donnée. Ensuite, il est évident d'évoquer les formalismes et langage de représentation d'ontologies avec les éditeurs qui assure leur manipulation.

II.2 Web sémantique

Le Web classique a montré une grande limitation pour l'exploitation de contenu ce qui mène à l'apparition du web sémantique qui est considéré comme un développement qualitatif et promoteur du Web standard. Selon *Tim Berners-Lee* (inventeur du Web et directeur du W3C), il a décrit le Web sémantique par cette expression : « le Web sémantique fournit un modèle qui permet aux données d'être partagées et réutilisées entre

plusieurs applications, entreprises et groupes d'utilisateurs ». Au contraire du Web classique qui se basent essentiellement sur la structure du document et les liens entre les pages dont l'extraction et l'exploitation d'informations par la machine est impossible, le Web sémantique offre une représentation sémantique de contenu qui rend l'information lisible et exploitable par les agents logiciels et donne l'avantage à extraire les connaissances cachées dans ces informations (Berners-Lee et al., 2001). Donc le rôle du Web sémantique ne limite pas à l'affichage et la façon de présentation des données mais aussi pour l'automatisation, l'intégration et la réutilisation des données entre diverses applications (Bourekache et al., 2016). On peut noter que le Web sémantique a réussi de joindre un sens bien défini aux données ce qui a facilité à la machine la compréhension et le traitement des informations généralement écrit en langage naturel, cette amélioration soulage largement l'utilisateur des tâches qui ont été précédemment de leurs prérogatives, et ajoute une souplesse dans la relation entre les ordinateurs et les personnes qui sont désormais capable de travailler et coopérer efficacement.

On note clairement qu'après l'apparition de la notion de Linked data par **Tim Berners Lee**, le Web sémantique a pris une nouvelle direction orientée vers le Web de données qui combine les technologies du Web sémantique avec les fondamentaux principes du Web existant afin de construire un réseau d'informations structurées, disponible en ligne et réutilisable par des nombreuses applications et domaines citons : le Big Data, le cloud, le web services intelligents, l'Open data, etc. autres chercheurs prévoient que le Web de demain sera doté d'une forme d'intelligence globale et collective plus que le Web d'aujourd'hui, il s'agit du Web 3.0.

« The Semantic Web will enable machines to comprehend semantic documents and data, not human speech and writings » Tim Berners Lee (Berners-Lee et al., 2001).

II.3 Architecture du Web sémantique

Le Web sémantique contient des nombreuses technologies organisées en couches interdépendantes, tels que les métadonnées, les ontologies, la logique et l'inférence. Il se repose sur les avantages fournis par plusieurs outils notamment les réseaux sémantiques et profite de la représentation des conceptualismes déjà existé, ce qui a conduit à l'amélioration de l'interopérabilité syntaxique et sémantique (Vandecasteele, 2012). Selon les besoins envisagés par une telle application, Il peut se comporter comme une base de données distribuée ou parfois il peut jouer un rôle semblable à une base de

connaissances collaborative surtout dans l'aspect de communication et l'interopérabilité. Les différentes couches fournies des fonctionnalités complémentaires et collaboratives. L'URI (Uniform Resource Identifier) il permet l'identification unique des ressources ainsi que les relations entre ces ressources (Bourekache et al., 2016), XML (Extensible Markup Language) est un composant fondamental qui fournit une syntaxe élémentaire de la structure de contenu dans le documents Web par un schéma, il assure l'interopérabilité syntaxique entre les différents systèmes d'information hétérogènes d'une façon automatisé (Ding et al., 2007), XMLS (XML Schéma) est un langage de définition qui utilise le syntaxe XML, son objectif est comment décrire la structure de données des documents XML par un vocabulaire et une structure hiérarchique bien défini.

L'extensibilité est aussi assuré par le modèle de données graphique RDF (Resource Description Framework) qui est déjà une norme fondamentale du Web sémantique, il n'est pas qu'un simple langage de syntaxe à balise mais il est considéré comme un type de réseau sémantique, l'usage du RDFS (RDF Schema) a permet d'annoter les ressources web et de donner un sens aux informations transmises qu'elles deviennent interprétable par des agents informatiques. A ce niveau, il apparaît l'aspect de l'interopérabilité sémantique dans le web. Néanmoins, la sémantique offerte reste un peut limiter. C'est pourquoi la nécessité à une autre couche d'ontologie et logique devient imminente. OWL (Ontology Web Language) a relie la déclaration de connaissances à l'inférence logique qui se base essentiellement sur la logique de description, il possède un vocabulaire très riche qui lui permet d'avoir une bonne expressivité.

La couche preuve dispose des langages logiques qui mettent en œuvre des outils de raisonnement permettent de tester la cohérence des informations, classées et d'évaluer les règles utilisées. Ils permettent également d'inférer des nouvelles informations à partir celles qui sont décrites à l'aide des moteurs d'inférence. Le langage SWRL (Semantic Web Rule Language) est un bon exemple de ces travaux (Ding et al., 2007). La couche confiance se positionne au haut du modèle, elle utilise des différents types de connaissances pour assurer un certain degré de fiabilité et garantir l'origine des informations par l'usage de signature numérique afin de détecter les changements ou la modification de documents. Ces couches sont complétées par une autre couche de cryptographie (Bourekache et al., 2016). La figure 8 représente l'architecture proposée par *Tim Berners - Lee* (Ding et al., 2007).

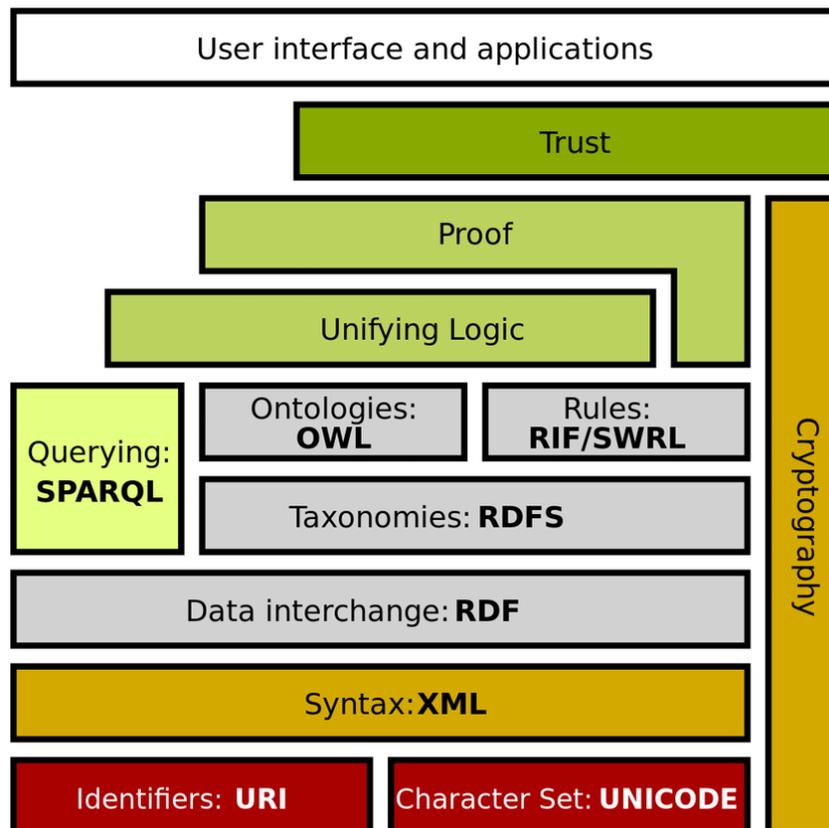


Figure 8 Architecture du Web sémantique proposé per Tim Berners-Lee (Chahdi, 2017).

Les métadonnées

Les différentes ressources dans le Web sémantique sont partagées et exploités grâce à la représentation structurelle et sémantique de contenu. Pour permettre d'exploiter ce contenu il est évident de décrire les ressources par des informations additionnelles bien structurées appelées « les métadonnées » qui sont des données sur les données. Elles sont existées sous plusieurs niveaux. Des informations qui décrivent la syntaxe, autres pour décrire la structure et le contexte sémantique. **Sheth** a donné un modèle qui englobe les différents types de métadonnées (syntaxique, structurelle et sémantique) (Bourekache et al., 2016).

II.4 L'ontologie

L'origine du terme « ontologie » revient à un mot grec composé « ontos » et « logos » que signifie « étant » et « discours », l'ontologie dans la philosophie est une branche de la métaphysique qui étudie la nature de l'être et de l'existence. **Aristote** a décrit l'ontologie comme « la science de l'être en tant qu'être ». Cependant en informatique et en science de l'information l'ontologie est un outil utilisé pour facilite la réutilisation des

connaissances par l'usage d'ontologie explicite (Ding et al., 2007). Elle constitue un modèle de données représentatif pour un ensemble structuré des termes et des concepts qui représentent la sémantique et le sens des informations dans un domaine donné, l'objectif principal de l'ontologie est de modéliser un ensemble de connaissances dans un domaine dont les concepts sont organisés dans un graphe duquel les relations entre ces concepts soient des relations sémantique ou relation de subsomption.

Les ontologies sont introduites dans des nombreux domaines comme le Web sémantique, l'intelligence artificielle, le génie logiciel, l'informatique biomédicale et les systèmes d'information où elles ont montré un grand succès pour faciliter le partage et la réutilisation des connaissances (Berners-Lee et al., 2001). Les éléments décrits par les ontologies sont (les individus, les classes, les attributs, les relations, les événements. etc.).

Plusieurs définitions données pour décrire l'ontologie mais la plus connue c'est de **Gruber** "une spécification explicite d'une conceptualisation". Ensuite cette définition est améliorée par **Borst** en affirmant que " Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée" (Bourekache et al., 2016), (Guarino et al., 2009), avec les significations des termes clés suivantes :

Formelle : pour que la machine puisse interpréter la sémantique des informations exploitées, l'ontologie se développe sous une forme lisible et compréhensible ce qui exclut le langage naturel.

Explicite : tous les concepts et les contraintes de leurs utilisations dans une ontologie sont définis d'une façon claire, précise et explicite.

Conceptualisation : c'est l'établissement d'un modèle abstrait d'un phénomène ou un certain aspect du monde réel par l'identification des concepts appropriés de ce phénomène.

Partagée : c'est un aspect exigé pour que l'ontologie avoir un consensus sur la connaissance entre la communauté d'utilisateurs, et elle n'est pas destiné pour des individus restreint. Pour simplifier le sens de l'ontologie on peut dire que la valeur l'ontologie pour données est comme la valeur de la grammaire pour langage.

II.4.1 Composants d'ontologie

L'ontologie est vue comme une structuration des concepts d'un domaine réunie pour exprimer les connaissances qui en dispose, En outre un vocabulaire partagé destiné pour définir la sémantique et la signification des termes et les relations entre ces termes dans ce domaine ayant un objectif de représenter les objets de monde sous un modèle compréhensible. À partir de ce point de vue, il est évident qu'une connaissance est formalisée dans une ontologie par les principaux composants suivants : concepts (ou classes), relations (ou propriétés), fonctions, axiomes (ou règles) et instances (ou individus) (Bourekache et al., 2016).

Les concepts : appelés aussi termes ou classe de l'ontologie, qui représente un objet, une notion, un ensemble ou une classe d'entités dans un domaine, les concepts constituent une abstraction d'une partie de la réalité et ils sont définis selon les objectifs envisagés par l'application et organisés sous forme d'une structure hiérarchique liés par des propriétés conceptuelles, ils sont conçus pour représenter les connaissances du domaine. Généralement les catégories dans une ontologie incluent les concepts. Il existe deux types de concepts : les concepts primitifs (canoniques) et les concepts définis (non canoniques) (Vandecasteele, 2012).

Les relations : constituent un composant principal de l'ontologie et décrivent les interactions entre les concepts dans le domaine analysé, elles montrent la structuration des concepts ou des classes, ce qui permet de représenter la sémantique de l'ontologie par la liaison des instances de concepts. Les relations se divisent en deux grandes catégories : les relations taxonomiques qui représentent les concepts dans une structure arborescentes et les relations associatives qui sont employé pour établir les liens entre les concepts à travers les structures arborescentes. On peut trouver d'autres types des relations comme les relations associatives, nominales, fonctionnelles ou causatives. Une relation se caractérise par un terme et une signature, il existe une diversité des propriétés liant deux relations, on cite les propriétés algébriques, la cardinalité, l'incompatibilité, l'inverse et l'exclusivité.

Les axiomes : dans une ontologie ont le rôle de modéliser les assertions et les conditions admis comme vrais pour un domaine, la construction des valeurs des classes ou des instances est faite par les axiomes ainsi que les propriétés des relations qui sont aussi considérés comme un sort d'axiome. Dans une ontologie, les axiomes comprennent des règles plus générales et peuvent vérifier la consistance par des axiomes formels. Les

objectifs attendus par l'inclusion des axiomes dans l'ontologie sont pour définir la signification des composants d'ontologie, définir les restrictions sur les valeurs des attributs, vérifier la validité d'une information spécifique et l'inférence de nouvelles informations

Instances : dans une ontologie les individus sont représentés par des instances qui correspondent à des instances concrètes de la classe originale. Elles sont considérées comme une définition extensionnelle d'une ontologie qui conduit à former une base de connaissance. Les objets sont responsables pour transporter les connaissances au sujet d'un domaine (Vandecasteele, 2012).

II.5 Raisons de l'utilisation des ontologies

Les ontologies offrent un cadre unificateur et fournissent des primitives facilitant la communication entre les individus, entre les individus et les systèmes et entre les systèmes. Elles peuvent être utilisées pour éliminer l'ambiguïté des termes comme elles offrent un potentiel d'inférence. Les ontologies sont utilisées dans des divers domaines pour (Laallam et al., 2014):

- ✓ Partager l'information et les connaissances, faciliter la compréhension commune de la structure de l'information entre les systèmes ainsi entre les personnes et la machine.
- ✓ Standardiser le vocabulaire à travers un domaine et avoir une compréhension commune.
- ✓ Faciliter l'accès aux des ressources hétérogène.
- ✓ Améliorer les processus de recherche d'informations (Coneglian et al., 2016).
- ✓ Permettre la réutilisation du savoir sur un domaine (définir une seul fois mais utilise dans plusieurs applications du même domaine)
- ✓ Expliciter ce qui est considéré comme implicite sur un domaine (définir les classes, les relations et les instances)
- ✓ Distinguer le savoir sur un domaine du savoir opérationnel.
- ✓ Analyser le savoir sur un domaine (le sens d'association entre les objets)
- ✓ L'exécution et au traitement de requêtes exprimées en langue naturelle

II.6 Les types d'ontologies

Dans la littérature, il existe une diversité des critères utilisés pour la classification des ontologies, chacune prend en considération la particularité d'utilisation visée. Qui conduit à l'apparition des modèles plus ou moins formels, générique ou bien spécifique selon un paradigme destiné à un domaine d'utilisation précis. Dans ce contexte on distingue sept types d'ontologie groupé en deux catégories (Figure 9), le premier ensemble contient trois types proposés par *Van Heijst* classé selon les critères de la richesse et le deuxième ensemble qui englobe quatre types classé selon les objectifs de la modélisation. Ces types des structures utilisées pour construire l'ontologie (Van Heijst et al., 1997):

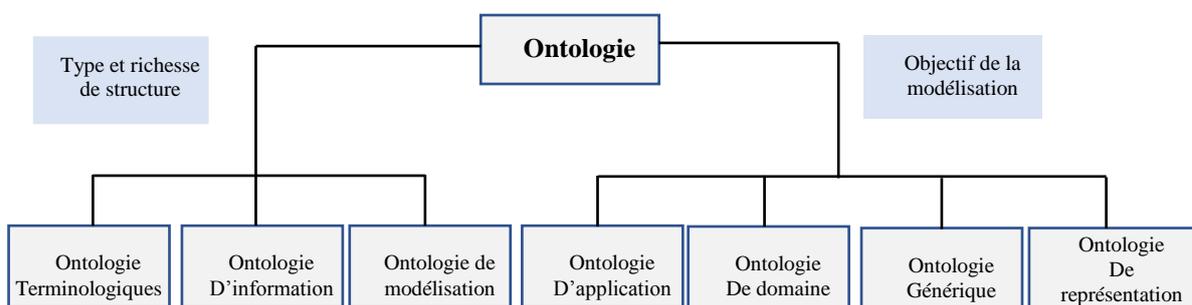


Figure 9 Classification des ontologies (Vandecasteele, 2012).

II.6.1 Les ontologies terminologiques

Dans un domaine de connaissance donné, les termes du vocabulaire sont utilisés d'une manière spécifique pour clarifier le sens du terme dans ce contexte (Guarino, 1997).

II.6.2 Les ontologies d'information

Généralement utilisées dans les bases de données afin de permettre le stockage des données et d'informations suite une spécification de la structure et le schéma de la base de données (Guarino, 1997).

II.6.3 Les ontologies de la connaissance

Pour ce type, l'ontologie est conçue selon le but de l'application envisagée, le choix d'une structure interne très riche est très recommandé pour faciliter le partage de connaissances entre les différents agents logiciels.

Le deuxième ensemble donné par *Guarino* et *Van Heijst* dont les critères de classification se basent sur les objectifs de la conceptualisation (Van Heijst et al., 1997), ce qui donne quatre catégories d'ontologie (figure 10) (Guarino, 1997).

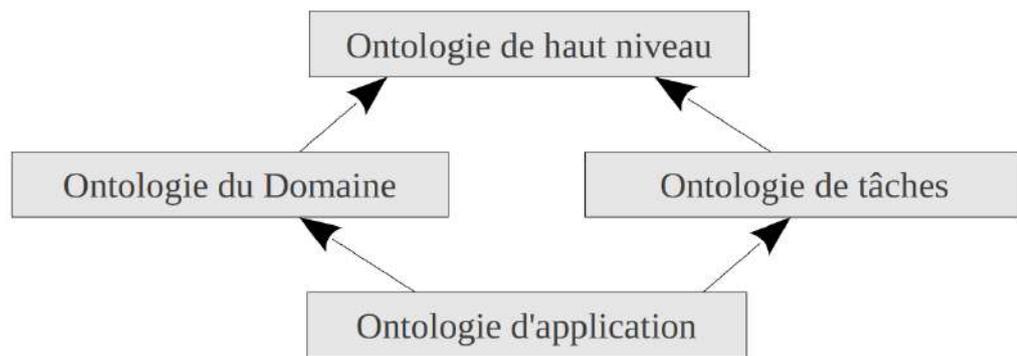


Figure 10 Typologie d'ontologie selon le niveau de conceptualisation

II.6.4 Les ontologies d'applications

Les concepts utilisés dans cette ontologie est très spécifique à un domaine particulier et ils ne sont pas réutilisable pour une autre application, les concepts sont destinés à jouer un rôle bien déterminé par les entités du domaine au cours de l'exécution de certaines activités. De plus la modélisation des connaissances d'un domaine nécessite une définition complète des concepts et leurs relations. Dans ce type d'ontologie le vocabulaire utilisés est spécialisé pour une tâche donnée afin de servir un domaine particulier tout en décrivant le processus d'exécution (Vandecasteele, 2012).

II.6.5 Les ontologies de domaine

Une spécification de conceptualisation des connaissances pour le domaine donné est très recommandée, ce qui permet la réutilisation de l'ontologie dans plusieurs applications du même domaine puisque les concepts et les relations entre ces concepts utilisés offrent une bonne couverture des vocabulaires, activités et théories de ce domaine (Djilani, 2017). *Mzoguchi* dans ses travaux explique que l'ontologie de domaine représente et caractérise la connaissance ou la tâche décrite dans ce domaine (Gómez-pérez et al., 2002).

II.6.6 Les ontologies génériques (haut niveau)

Ce type d'ontologie utilisé pour représenter les connaissances communes ou consensuelles, elle est considérée comme un noyau ontologique qui peut être réutilisé dans des autres domaines grâce à la modélisation des connaissances plus au moins abstraite, elles sont conçues pour réduire les incohérences des termes définis en aval de la hiérarchie. De plus, les ontologies de haut niveau utilisent des connaissances générales et inclut un vocabulaire lié aux choses, évènement, espace, temps, causalité, fonction,

comportement, etc. qui donne l'avantage pour l'employer dans des domaines différents (Vandecasteele, 2012).

II.6.7 Les ontologies de représentation

L'idée de base utilisé pour la modélisation est de représenter et formaliser les connaissances d'une façon assez explicite sous un paradigme donné, dont la représentation des entités du monde réel est sans priori. Les concepts de ce type d'ontologie est aussi valable pour l'utiliser dans d'autres ontologies comme les ontologies génériques ou de domaines car la conceptualisation de formalisme de représentation de connaissances est très explicite et modélise les primitifs utilisés dans les connaissances représentées (Gómez-pérez et al., 2002).

Une autre classification des ontologies est ajoutée à celle de *Guarino* qui se basent sur l'objet de conceptualisation par *Uschold* qui a porté sur le niveau de formalisation de représentation, qui a donné quatre niveau classé selon le degré de formalisation qui sont respectivement : Ontologies informelles (exprimées en langue naturelle), Ontologies semi-informelles (la sémantique du langage est plus structurée et limitée), Ontologies semi-formelles (exprimée formellement dans un langage artificiel) et Ontologies formelles (exprimée dans un langage artificiel disposant d'une sémantique formelle permettant d'effectuer des vérifications).

II.7 Construction d'ontologies

Le processus de développement d'ontologies qui sont d'ailleurs destiné à être utilisé dans des systèmes d'information doit suivre des procédures similaires à ceux utilisés en génie logiciel, du fait qu'une ontologie est destiné pour supporter des manipulation automatique par des agents logiciels qui doivent connaître les étapes à suivre pour effectuer les tâches appropriées à sa mission, soit pour appeler une procédure inscrit dans son code ou bien effectuer une correspondance entre la description des tâches et les procédures. Cette description est réalisée à l'aide d'un langage formel qui se caractérise par une représentation ontologique, qui part d'une définition des primitives de représentation du domaine et définir le sens et la sémantique des concepts qui sont utilisés par la suite pour la représentation de connaissances. Pour qu'une ontologie soit durable et réponde aux objectifs envisagés, il est évidant de suivre et respecter les règles de construction d'ontologie. En commençant par la définition du domaine d'application, assurer la réutilisation d'ontologies existantes pour faire profiter de ce qui déjà disponible et

exploiter le maximum des langages standards. Des nombreuses méthodologies proposées pour la construction d'ontologies mais aucun consensus est adopté pour se convenir à une meilleure pratique.

Néanmoins, tous les processus suivis partagent une certaine étape primordiale qui sont : la spécification sémantique, la conceptualisation et la formalisation (Vandecasteele, 2012). Autres auteurs ajoutent à la phase de formalisation par une phase d'ontologisation et une phase d'opérationnalisation. Le processus de construction d'ontologies comprend des passages aller-retour entre les phases de conceptualisation et la formalisation. Donc le parcours de création n'est pas linéaire puisqu'il a un aspect itératif et incrémental (Guarino, 1997). Parmi les méthodes les plus connues on cite *On-To-Knowledge* et *Methontologie*, ces deux méthodologies couvrent les étapes essentielles du cycle de vie d'ontologies. Afin d'obtenir une ontologie plus cohérente, il faut respecter quelques recommandations et principes généraux qui sont utilisés pour guider la modélisation lors de la construction d'une ontologie à savoir (Vandecasteele, 2012):

La clarté et l'objectivité : La définition des concepts doit être claire et précise loin de toute ambiguïté et confusion, et transmettre le sens attendu du terme définit. Néanmoins, la définition doit être complète et objective, indépendante complètement du contexte. Les entités ontologiques sont accompagnées par des définitions en langage naturel et des exemples qui permettent d'avoir un sens précis et unique (Gruber, 1995).

La complétude et perfection : L'expression des axiomes logiques est fortement recommandée pour avoir le maximum des conditions nécessaires et suffisantes pour que les axiomes accomplissent leurs rôles parfaitement. La cohérence dans une ontologie, les différents composants ne doivent jamais se contredire les uns avec les autres, y compris les connaissances inférées avec les concepts primitifs (ou les définitions exprimées en langage naturel). Donc, une consistance logique est obligatoire entre tous les composants de l'ontologie (Gómez-pérez et al., 2002).

L'extensibilité : Le concepteur d'une ontologie doit prévoir l'ajout des nouveaux concepts même après la mise en œuvre de l'ontologie sans avoir le risque de contradiction ou d'entraîner une révision des concepts existants auparavant. Cette anticipation protège les fondations de l'ontologie (Gómez-pérez et al., 2002).

L'engagement ontologique minimal : Parmi les objectifs de l'ontologie est de définir un vocabulaire assez expressif pour décrire un domaine donné d'une façon exhaustif. **Gruber** affirme qu'un engagement ontologique peut être minimisé en spécifiant la théorie la plus faible couvrant un domaine, elle définit que les termes nécessaires pour partager les connaissances consistantes avec cette théorie (Gruber, 1995). Les spécialisations du monde réel sont réalisées pendant la phase de construction.

La déformation d'encodage minimale : Parfois la spécification influe sur la conceptualisation lorsqu'une déformation d'encodage se produite, cette déformation doit être évité autant que possible (Gómez-pérez et al., 2002).

Distinction ontologique : Une séparation des classes qui représentent des différents critères d'identité. Pour éviter le chevauchement des individus ou des propriétés entre les classes.

Autres critères sont utilisés selon les exigences du domaine d'ontologie comme la modularité minimale, la subdivision des taxonomies, la distance sémantique minimale, la normalisation des termes et la différenciation sémantique.

II.8 Formalismes et langages de représentation d'ontologies

Pour représenter une connaissance dans un tel système informatique comme pour les individus, un formalisme de représentation et un langage spécifique est indispensable qui sont généralement différents selon les formes utilisées pour la représentation ainsi la puissance descriptive qui est fourni par le langage utilisé. Le choix de formalisme et le langage pour représenter une ontologie est très critique afin d'obtenir une bonne expressivité. Cela se fait selon les exigences et la spécificité de l'application envisagé et le domaine concerné.

II.8.1 Formalismes de représentation d'ontologie

Parmi les principaux objectifs de l'intelligence artificielle est la modélisation adéquate du domaine étudié et de concevoir des systèmes capables à modéliser les connaissances sous une forme facile à manipuler, si on tient compte que chaque connaissance dans la réalité est généralement basée sur un schéma cognitif spécifique pour chaque individu, l'implémentation de ces connaissances sous une forme pratique est efficace et très imminente. De plus, la sélection des structures et des mécanismes employés pour représenter ces connaissances doit être l'objet d'une étude tentative pour avoir un bon

formalisme de représentation manipulable par les systèmes et les individus. Ces formalismes se diffèrent selon le degré d'expressivité, de la rigueur et de la sémantique offerts (Vandecasteele, 2012).

Dans la littérature, il existe deux familles de formalismes, la première est apparue avant les années 1980 qu'il s'agit des formalismes basés sur la logique (descriptive, de premier ordre, etc.) et les représentations non logiques (les réseaux sémantiques, les frames, etc.). La deuxième famille se base principalement sur l'utilisation commune de différents formalismes à la fois (exemple les schémas avec la logique), qui a conduit à l'apparition des nouvelles générations utilisés pour la description d'ontologies telle que la logique descriptive.

Le choix des formalismes de représentation se fait selon l'importance affichée dans le domaine de représentation et l'héritage conceptuel. Les principaux formalismes utilisés sont les réseaux sémantiques, les graphes conceptuels, les frames et la logique de description. Ces différents formalismes possèdent une sémantique formelle puissante qui permet d'avoir une formalisation des connaissances satisfaisante, néanmoins ils n'ont pas le même niveau d'expressivité ni la même capacité de raisonnement chaque formalisme est conçu pour répondre à des besoins spécifiques exigés par le domaine d'application et les objectifs envisagés, le formalisme de représentation le plus utilisé pour concevoir des ontologies est la logique de description qui est considéré comme une référence pour les nouveaux langages du web sémantique grâce à sa forte expressivité et la capacité d'inférence qui la détient (Vandecasteele, 2012).

II.8.2 Langages de représentation d'ontologies

L'objectif principal d'un langage de représentation d'ontologie est d'exprimer une interprétation assez satisfaisante du monde à l'aide d'une sémantique formelle et un format syntaxique précis. Les premiers langages traditionnels sont apparus dans les années 1990 basant essentiellement sur le domaine de l'intelligence artificielle. Des années plus tard, l'apparition du Web a exigé une démarche de standardisation pour dissimuler les différents paradigmes utilisés. Ce travail est piloté par un groupe OntoWeb du W3C, les travaux concluent une multitude des langages standardisés améliorés progressivement qui ont introduit une nouvelle manière de manipulation de l'information. Les langages issus du Web sémantique deviennent les plus utilisés pour la construction d'ontologies même dans des domaines hors du monde web (Vandecasteele, 2012).

Les langages traditionnels

Ces langages d'ontologie appelés aussi langages traditionnels, qui sont apparus au début des années 1990, ils se basent essentiellement sur la représentation logique et non logique ou parfois les deux à la fois, il existe une diversité des Langages de ce genre dont les plus connus sont : Ontolingua, Cycl, Loom et Flogic (Vandecasteele, 2012).

- ✓ **Ontolingua** Est à l'origine une interlingua destiné pour représenter et partager l'ontologie développée par KSL (Knowledge Systems Lab) (Mizoguchi, 2004). Il est conçu par l'ajout des fonctionnalités de représentation et de traduction du langage KIF (Knowledge Interchange Format) (Ghafour, 2004), pour représenter la connaissance il utilise une syntaxe LISP. Les schémas et la logique de premier ordre sont utilisés pour la modélisation des connaissances. Ontolingua possède une bibliothèque d'ontologie modulaire et réutilisable. Il a une expressivité suffisante pour représenter les connaissances déclaratives. Il peut traduire de et vers des langages logiques de description tels que Loom, Epikit, etc. Il est utilisé par Ontolingua Server (Gómez-pérez et al., 2002). Actuellement, il n'est plus actif à cause de l'avènement des langages de la famille XML.
- ✓ **Cycl** Est un langage de représentation de connaissance qui se caractérise par l'expressivité et la déclaration, il est similaire à la logique de premier ordre (Vandecasteele, 2012). le langage *CycL*, qui est basé sur le calcul des prédicats et dont la syntaxe est similaire à celle du langage Lisp. *Cycl* possède un moteur d'inférence qui peut effectuer une déduction logique général. Le développement de l'ontologie *Cyc* est fait grâce au *Cycl* (Fensel, 2001). Toutefois, *Cycorp* travail à fournir au système *Cyc* la possibilité de communiquer avec l'utilisateur en langage naturel (Mizoguchi, 2004), et à simplifier la démarche de représentation de la connaissance via de l'apprentissage automatique.
- ✓ **Loom** Développé en même temps qu'*Ontolingua* (Corcho et al., 2003), ce langage est considéré comme un langage de haut niveau qui partage une des caractéristiques avec les deux langages précédents de faite qu'il est basé sur *Loom* est basé sur les DL et les règles de production et fournit des classifications automatiques des concepts, il fait partie de la famille de KL-One. Il possède une expressivité et déclaration explicite, et un moteur d'inférence capable de vérifier automatiquement l'ontologie ainsi sa consistance (Gómez-pérez et al., 2002).

- ✓ ***FLogic*** Le langage de schéma logique (frame logic) comme indique son abréviation, est un langage qui combine les schémas et la logique de premier ordre (Corcho et al., 2003), donc il détient les avantages de la modélisation conceptuelle et les langages orientés objets et orientés cadres. Il peut supporter d'autres caractéristiques par rapport aux autres langages tels que la possibilité d'héritage, l'encapsulation, inclusion des objets simple et complexe. etc. le système de déduction de *FLogic* se base sur le calcul de prédicat et l'héritage structurel et comportemental (Vandecasteele, 2012), *OnoBroker* est utilisé comme un moteur d'inférence. Le projet *NeOn* et protégé utilise *FLogic* comme un environnement de développement d'ontologie.

A travers des années, des travaux intéressants dans le domaine de l'intelligence artificielle sont consacrés pour avoir une capacité de raisonnement acceptable pour les différents langages de représentation traditionnels développés. Mais après l'apparition et le développement croissant de Web des nouveaux besoins sont exigés qui n'ont pas remplis par les langages traditionnels. Cela conduit forcément à la création de nouveaux langages capable à aboutir les objectifs attendus dans le nouveau contexte.

Les langages issus du Web Sémantique

Après l'avènement du Web sémantique au cours des dernières années, les langages traditionnels ne se concordent plus aux nouveautés imposées par les applications du Web sémantique car elles s'intéressent beaucoup plus à la syntaxe et la structure de données dans une page Web. Il était donc très évident d'accompagner cette évolution remarquable par la création des langages d'ontologie qui répondent aux exigences de cette nouvelle technologie qui ne cesse pas à se développer jour après jour. Du moment que les ambitions des utilisateurs et des développeurs s'accroître progressivement selon les besoins et la concurrence exigée par le marché du web, les ontologies devient bien placé pour jouer un rôle crucial dans le Web sémantique de faite qu'elles offrent une spécification formelle des vocabulaires pour décrire le contenu du Web. Les ontologies ont apporté un changement majeur dans le contenu du Web déjà existé et conduites à l'apparition de troisième génération du Web. Ce qui permet d'accéder et de comprendre les informations d'une manière assez facile que ce soit par un humain ou bien par une machine. Ces atouts sont désormais garantis par la puissance d'expressivité qui caractérise les langages d'ontologie.

Dans la littérature plusieurs langages ont été développés afin de décrire les ontologies. Avec l'absence d'une contrainte sémantique sur la description des informations dans le langage XML qui se repose sur une représentation hiérarchique et logique (Vandecasteele, 2012), c'est pourquoi les langages RDF et RDFS (des langages orientés-Web créés par W3C) sont intégrés pour annoter les ressources du Web et permettent d'avoir une première représentation sémantique, d'autres langages qui se réfèrent à la logique de description sont ajoutés pour remédier l'insuffisance d'expressivité offerte par les langages cités, DAMEL+OIL, WOL et autres représentent une nouvelle couche ontologique qui devient une recommandation du W3C. L'un des objectifs attendus de ces langages est de représenter une information sémantiquement dans un Web capable d'échanger des données hétérogènes dans un environnement aussi hétérogène (Gómez-pérez et al., 2002).

- ✓ ***Ressource Description Framework (RDF)*** Est une norme fondamentale du Web sémantique et le premier langage standardisé par W3C (Vandecasteele, 2012) utilisé pour la description des ressources des Web et la représentation des métadonnées, Il fournit également des mécanismes permettent de représenter explicitement les services, les processus et les modèles d'affaires, en plus il assure la reconnaissance d'informations non explicites. Il est considéré comme un langage simple de syntaxe à balise qui est représenté par un graphe simple dont les nœuds sont des ressources ou littéraux et les arcs sont les relations binaires entre ces nœuds. En outre, RDF est considéré comme un type de réseau sémantique qui utilise la syntaxe XML très similaire au model relationnel (Ding et al., 2007). L'objectif de Création de langage RDF est pour représenter l'information d'une manière la plus souple et la plus générique possible, RDF s'appuie sur un modèle fixe composé d'assertions (déclaration) (Berners-Lee et al., 2001). Ce modèle de données est conçu sous forme des objets (ressources) et des relations qui associent ces objets sous forme de triplets, dont chaque triplet est représenté par (sujet, prédicat, objet), dans lequel la ressource est représenté par le sujet et l'objet, le prédicat représente le type de propriété et l'objet représente une donnée ou une autre ressource (Ding et al., 2007), Les primitives de modélisation offertes par RDF sont très basiques mais il offre une grande liberté dans la création d'extensions personnalisées (Gómez-pérez et al., 2002). Dans un graphe RDF un sujet peut être une référence de différentes notions. Dans ce cas, et pour éviter toute ambiguïté chaque ressource RDF est identifiées à l'aide d'un Universal Ressource Identifier (URI) qui assure une identification unique pour n'importe quelle

ressource utilisée. Afin que le graphe RDF soit facile à comprendre par la machine, la nécessité de codage des informations sous un format facilement maniable. Il s'agit d'une démarche de sérialisation qui a plusieurs syntaxe (N-Triples, N3, etc.), le format le plus utilisé dans le processus sérialisation est le RDF/XML (Vandecasteele, 2012).

- ✓ **RDFS (RDF Schéma)** C'est une extension sémantique du langage RDF, publié par W3C et devient une recommandation en 2004. RDFS est considéré comme un méta-modèle et un langage fondamental du Web sémantique, les connaissances décrites par ce modèle sont organisées d'une manière centrée sur le concept dont l'utilisateur peut représenter le monde à différents niveaux d'abstraction (Ding et al., 2007), RDFS utilise une sémantique formelle pour spécifier les ressources utilisées dans un triplet RDF (Vandecasteele, 2012). Grâce à sa forte expressivité de vocabulaire, il permet de décrire et de structurer les classes d'objets (groupes de ressources similaires) et les propriétés des ressources (relations entre ces ressources) par les primitives (rdfs: Class, rdfs : Property, rdfs : subPropertyOf, etc.).

Il donne aussi la possibilité d'organiser les ressources hiérarchiquement par des liens de subsomption (hiérarchie de taxinomie) qu'il les détient (rdfs:subClassOf, rdfs:subPropertyOf), ces liens de spécialisation permettant les classes et les propriétés d'hériter des caractéristiques définies auparavant dans les classes et les propriétés parentes (Mizoguchi, 2004). En outre, il associe un sens aux propriétés attribués aux ressources et spécifier des contraintes sur les valeurs d'une propriété (rdfs:domain, rdfs: range). L'instanciation des classes et des propriétés des ressources est faisable pour un domaine spécifique qui rend les relations moins dépendantes sur les concepts (figure 11).

Les avantages offerts par RDFS lui qualifient d'être utilisé comme format de représentation dans de nombreux outils et projets comme Amaya, Protégé, Mozilla, SilRI, etc. (Gómez-pérez et al., 2002). Malgré l'insuffisance d'expressivité pour décrire les situations complexes (l'impossibilité de spécifier l'unicité, la disjonction des classes, la restriction de cardinalité, etc.) le langage RDFS dispose des éléments fondamentaux pour définir les ontologies qui sont intégrés par la suite dans un langage d'ontologie plus expressif comme OWL qui est construit au-dessus du RDF (Bourekache et al., 2016).

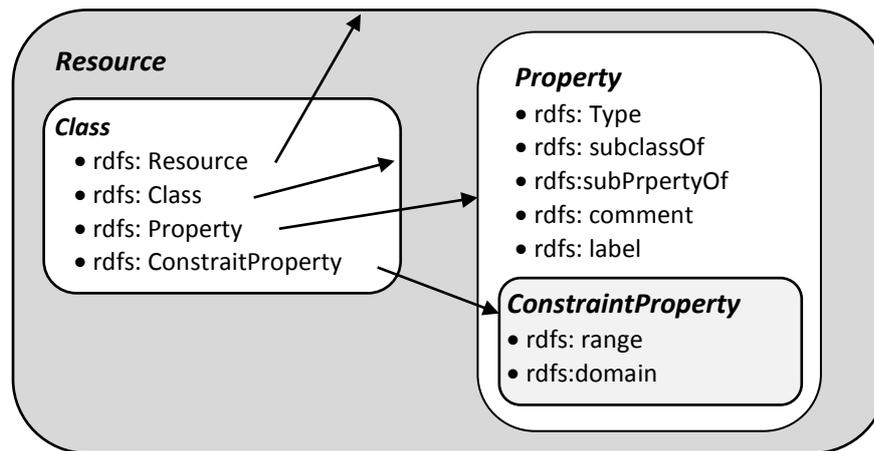


Figure 11 Classes et propriétés définies par le RDFS (Vandecasteele, 2012).

- ✓ **DAML+OIL** Est une fusion de deux langages DAML (Darpa Agent Markup Language) et OIL (Ontology Interchange Language) afin de bénéficier des avantages des deux langues qui est par la suite destiné à l'utilisation d'ontologie sur le Web. Bien que RDFS n'assure pas une expressivité suffisante aux exigences du nouveau Web (Vandecasteele, 2012), un groupe des chercheurs européens sont appelés à développer le langage OIL. Ce langage utilise une syntaxe RDF et permet d'exprimer une sémantique par l'usage du modèle de frames (Saettler et al., 2017). En plus, il détient des nouvelles primitives permettant de définir des classes à l'aide de mécanismes ensemblistes qui se base sur la logique de description (intersection, union, complément des classes), et il peut contraindre la cardinalité et restreindre la portée. La conception du DAML est basée sur une syntaxe RDFS pour garantir l'interopérabilité sémantique et offre des primitives de représentation des frames permettant l'expression d'ontologie (Gómez-pérez et al., 2002).

L'intégration d'OIL avec DAML offre un mécanisme de raisonnement et d'inférence similaire à celle de la logique de description qui peut exprimer la sémantique formelle, notamment les calculs de liens de subsumption (Corcho et al., 2003). En outre, DAML+OIL permet de définir un ensemble de constructeurs et d'axiomes permettant de créer une ontologie et révéler les incohérences (Ghafour, 2004). Cependant, DAML+OIL souffre également de certaines restrictions de fait qu'il considère uniquement les relation unaire et binaire, ne permet pas la composition et la fermeture transitive, la déclaration des variables, etc. malgré la considération du DAML+OIL comme le premier langage qui fournit des mécanismes d'inférence sur les concepts

d'ontologies mais il n'a pas été normalisé, ce qui a encouragé le W3C à recommander un nouveau langage standardisé plus puissant capable d'assurer une représentation des ontologies dans le Web sémantique et qui possède une expressivité suffisante pour pallier les insuffisances enregistrées par les autres langages (Bourekache et al., 2016).

- ✓ **OWL (Ontology Web Language)** La couche Ontologies et Logique offre une définition de langages ontologiques comme DAMEL, OIL et OWL. OWL est un standard défini par W3C depuis 2004 pour la création d'ontologies (Ghafour, 2004), OWL couvre la plupart des caractéristiques du Langage DAML+OIL et renomme la majorité de ses primitives qui se base essentiellement sur la logique de description, il possède un vocabulaire très riche qui lui permet d'avoir une bonne expressivité par la description des propriétés, les classes, les relations entre les classes, la cardinalité, l'égalité, le typage des propriétés, les caractéristiques de propriétés, leurs transitivités, etc. (Mizoguchi, 2004). OWL fournit une représentation explicite du sens pour les termes d'un vocabulaire et rendre les ressources des Web accessible aux processus automatisés par la structuration standard et compréhensible et l'ajout des méta-informations à ces ressources. Il est de ce fait possède toute la sémantique nécessaire pour la description des connaissances et permet ainsi une modélisation plus fine contrairement à XML, RDF et RDFS (Vandecasteele, 2012). De plus, OWL permet de recueillir les informations à partir de différentes ressources distribuées et maitre en relation les ontologies et d'importer des informations provenant d'autres ontologies (Ghafour, 2004).

Le W3C propose trois sous-langages OWL pour faciliter et simplifier l'utilisation du OWL, ces variantes échangent la complexité computationnelle et possèdent un niveau d'expressivité croissante selon les besoins de représentation ou d'inférence : OWL-Lite, OWL DL et OWL Full. Chacun de ces sous langages étant lui-même une extension de son prédécesseur (Vandecasteele, 2012):

- ✓ **OWL-Lite** C'est le sous langage OWL le plus simple, utilisé pour des besoins essentiellement d'une hiérarchie de classification et des contraintes simples (Vandecasteele, 2012). Il contient des constructeurs RDF (*rdfs:subClassOf*, *rdfs:Property*, etc.) et éventuellement quelques vocabulaire de OWL comme une nouvelle classe (*owl:Class*), notions d'équivalence (*owl:sameIndividualAs*) et d'exprimer des contraintes simples (*owl:minCardinality* et *owl:maxCardinality*) avec des entiers égaux à 0 et 1 (Ghafour, 2004). L'avantage majeur offert par OWL-

Lite est la simplicité pour les utilisateurs ainsi les développeurs avec un processus de raisonnement efficace basé sur la logique de description adapté particulièrement pour les cas de migration de thésaurus et taxonomies (Ding et al., 2007).

- ✓ **OWL-DL** Qui signifie OWL Description Logics, il est basé sur la logique de description et se concentre sur la sémantique formelle commune, une expressivité supérieure à celle de OWL-lite est assuré (la conjonction, la disjonction et la négation) (Vandecasteele, 2012) tout en gardant une complétude des raisonnements (toutes les conclusions sont garanties d'être calculables) et une décidabilité des calculs (les calculs achèveront en un temps fini). Il possède deux mécanismes d'inférence importants : la subsomption et la cohérence (Ding et al., 2007). Toutefois, il étend les contraintes de cardinalité qui permet d'utiliser tous les entiers positifs. Il peut capturer des connaissances sur un domaine dont les instances sont regroupées en classes avec une relation binaire comme il peut affirmer que deux classes n'ont aucune instance commune (*owl:disjointWith*). OWL-DL utilise le vocabulaire d'ontologies OWL avec certaines restrictions (Ghafour, 2004).
- ✓ **OWL-Full** Destiné pour les utilisateurs qui demandant une expressivité maximale et une liberté syntaxique de RDF mais sans garantir la décidabilité (Vandecasteele, 2012). OWL-Full permet l'ontologie d'améliorer la signification du vocabulaire prédéfini (RDF ou OWL). Il se diffère à OWL-DL par la conjonction de l'espace de classe et l'espace d'instance, le vocabulaire de l'OWL est utilisé sans aucune restriction (Ding et al., 2007). Dédier principalement aux applications qui ont besoins à un haut niveau d'expressivité et ne demandent pas une garantie de complétude des calculs. L'un des inconvénients de OWL-Full est l'absence d'une implémentation complète de ce sous langage, ce qui donne un support de raisonnement moins prévisible (Ghafour, 2004).
- ✓ **OWL2** Est un langage d'ontologie pour le Web sémantique avec une signification formellement définie. Développé par le groupe de travail W3C, WOL 2 est considéré comme une révision et extension du OWL afin d'améliorer l'expressivité parfois limité du OWL et simplifier la syntaxe du langage OWL qui s'apparait un peu compliquer. Autre raison qui a encouragé à cet amendement est l'impossibilité de la notation des axiomes. Donc, l'addition des nouveaux axiomes (ObjectHasSelf, ObjectExactCardinality, etc.) a renforcé les capacités originales d'OWL. Les besoins réclamés par les utilisateurs ont pour but de profiter au maximum du mécanisme de

raisonnement efficace. OWL 2 regroupe des sous-ensembles (OWL 2 EL, OWL 2 QL, OWL 2 RL) dont chacun est destiné pour une application spécifique. Parmi les primitives fournis par OWL 2 on cite les classes, les propriétés, les individus et les valeurs de données qui sont stockées en tant que documents Web sémantiques, le vocabulaire du OWL 2 est écrit en RDF comme il est possible d'utiliser d'autres syntaxes plus lisible (Manchester Syntax, Turtle) qui sont utilisés dans plusieurs éditeurs d'ontologie (Vandecasteele, 2012). La figure 12 montre l'évolution historique qui a conduit à l'apparition de l'OWL.

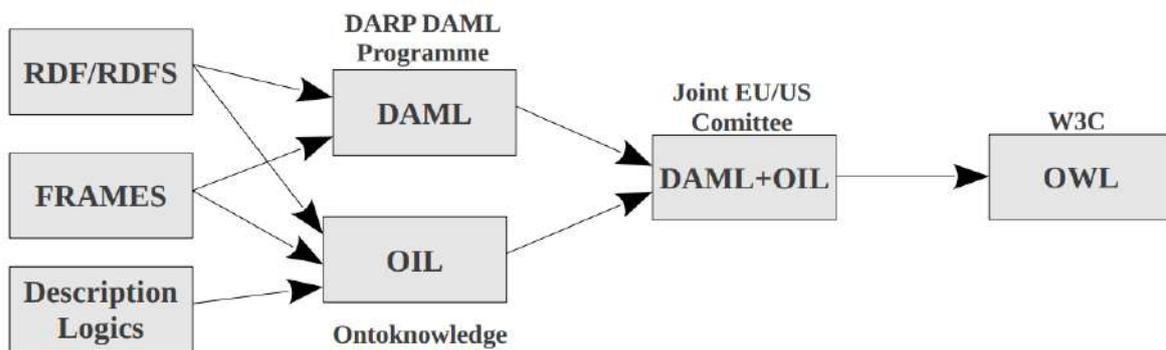


Figure 12 Les familles de langages qui ont conduit à l'OWL (Vandecasteele, 2012).

SWRL (Semantic Web Rule Language)

Au-dessus de la couche ontologie dans l'architecture du Web sémantique s'ajoute un niveau supplémentaire consacré pour effectuer des règles spécifiques à des situations particulières. Ce niveau de preuve dispose des langages logiques qui mettent en œuvre des outils de raisonnement permettant à la fois de tester la cohérence des informations, les classiers et d'évaluer les règles utilisées. Ils permettent également d'inférer de nouvelles informations à partir de celles décrites à l'aide de moteurs d'inférence. Une règle dans le contexte ontologie est rédigé sous forme d'une instruction conditionnelle « si la condition est vérifiée alors une conséquence doit être exécutée » donc une nouvelle connaissance est ainsi ajoutée (Vandecasteele, 2012). Plusieurs approches proposées dans ce contexte qui sont dans le vois de standardisation (Jena Rules, OWL 2 DL Axioms, etc.).

Le langage SWRL (Semantic Web Rule Language) est un bon exemple de ces travaux qui combine entre l'expressivité, la standardisation et la décidabilité, comme il est possible de l'intégrer dans les bibliothèques de programmation (API) existantes. Bien que

SWRL soit en cours de soumission dans le W3C pour qu'il soit un standard homologué, des nombreux moteurs d'inférence de différentes applications commence à supporter SWRL comme Protégé, Pellet, *Bossam*, *Hoolet*, *KAON2*, *RacerPro* et *R2ML*. L'objectif du SWRL est de renforcer les potentialités du langage OWL DL avec l'addition des nouvelles règles tout en respectant la compatibilité de la syntaxe sémantique existante. Le raisonnement effectué est basé sur les logiques de description et les règles de Horn avec la structure antécédent (s) \rightarrow conséquence (s) (Horrocks et al., 2004), les antécédents et les conséquences sont des atomes exprimés par des concepts OWL (classes, propriétés, individus, etc.). SWRL est vue comme une combinaison de OWL DL and OWL Lite qui sont des sous langage OWL et RuleML (Rule Markup Language) (Ding et al., 2007).

II.8.3 Éditeur d'ontologies

Formaliser une connaissance est une tâche très critique qui nécessite un soin attentif pour permettre la manipulation facile et l'intégration sûre des connaissances représentés au sein d'un système informatique. C'est pourquoi, au cours des dernières années des nombreux environnements et interfaces spécifiques de construction d'ontologies sont développés pour faciliter la création d'ontologies (collaboratives ou indépendantes) et de fournir un support pour le processus de conception (Vandecasteele, 2012). Cependant, certains d'entre eux sont encore maintenus et plus aboutis dans le demain du Web sémantique. Des nombreux critères sont tenus en comptes lors du choix d'un outil comme l'architecture logicielle et évolution de l'outil, interopérabilité avec d'autres outils et langages de développement d'ontologies, représentation des connaissances, services d'inférence attachés à l'outil, la réutilisabilité, etc. (Alatrish, 2013). L'éditeur qui permet de gagner du temps lors du développement d'une ontologie aide beaucoup les ingénieurs à se concentrer sur la sémantique sans qu'ils trop penser à l'organisation syntaxique (Ding et al., 2007).

- ✓ **Protégé** Parmi les outils les plus reconnus dans le cadre de la création et du développement d'ontologies, Protégé est au premier plan développé au milieu des années 1990 par l'Université de Stanford. Développé en Java et possède une architecture modulaire et flexible conçu pour intégrer des nombreux langages (RDF, RDFS, OWL, DAML+OIL, etc.) et plugins (Alatrish, 2013). Les versions précédentes avaient deux modes de modélisation, Protégé-frames dont l'ontologie se compose par des classes organisées hiérarchiquement et Protégé-OWL qui supporte le format OWL et permet d'éditer, de visualiser, ajouter les classes et de vérifier les contraintes, et

même inférer des nouveaux faits (Vandecasteele, 2012). Il représente une plate-forme extensible qu'il peut supporter et gérer des contenus de différents formats (Mizoguchi, 2004). Protégé-OWL est par la suite amélioré vers la version 4 qui se base sur l'API Open Source OWL API supportant la possibilité de charger et fusionner (semi-automatique) une diversité d'ontologies sur le même endroit de travail à l'aide de plugin Prompt (Corcho et al., 2003). Protégé est adopté par une large communauté des développeurs ainsi des experts de domaine grâce aux fonctionnalités offertes et la riche et souple interface graphique (GUI) qu'il possède (Ding et al., 2007). Les applications développées avec Protégé peuvent servir à résoudre des problèmes et pour la prise de décisions dans un domaine spécifique.

- ✓ **OntoStudio** Est un environnement commercial de modélisation et de développement professionnel d'ontologies. Développé par l'entreprise *Ontoprise* basé sur IBM *Eclipse* (Vandecasteele, 2012), destiné pour la création et la maintenance d'ontologies de différents formats par l'utilisation des moyens graphiques, ainsi des outils de modélisation d'ontologie et les règles permettant d'intégrer les ressources de données hétérogènes. Comme Il est possible d'ajouter d'autres modules auto-développés permettant d'enrichir ses capacités et l'adapter selon les besoins. Il se distingue par ses capacités étendues dans la modélisation d'ontologies intuitives.

Les fonctions pour créer des classes et des instances, définir des propriétés (attributs et relations), affectation des valeurs aux propriétés au niveau de l'instance sont aussi offerts. *OntoStudio* peut supporter les normes W3C comme RDF, RDFS, OWL, SPARQL et F-Logic pour traiter en logique des règles, De plus, les ontologies peuvent être exportées vers bases de données relationnelles via JDBC. En outre, *OntoStudio* peut s'interfacer avec des bases de données, des systèmes de fichiers, des applications et des services Web grâce aux nombreux connecteurs qu'il détient. Il est basé sur l'architecture client / serveur dont plusieurs clients peuvent utiliser le serveur *OntoBroker* à l'aide d'*OntoStudio* pour créer et améliorer simultanément des ontologies collaboratives (Alatrish, 2013). Une version *NeON Toolkit* open source est disponible.

- ✓ **OntoEdit** Est un environnement d'ingénierie d'ontologie flexible et extensible basé sur une architecture modulaire, il partage plusieurs aspects avec les éditeurs précédents. Destiné pour éditer, parcourir et maintenir des ontologies, ainsi d'importer et exporter des ontologies de différents formats (XML, RDFS, DAMEL,

etc.) grâce aux riches fonctionnalités qu'elles possèdent. Il se base sur *FLogic* pour assurer l'inférence en utilisant *OntoBroker* (Corcho et al., 2003). Afin de développer une ontologie *OntoEdit* possède sa propre méthodologie *On-To-Knowledge* originaire de Common KADS, deux outils sont utilisés dans la phase de capture d'ontologie (*OntoKick* et *Mind2Onto*) (Mizoguchi, 2004). En outre, il offre d'une manipulation facile à l'aide d'une interface visuelle permettant l'identification libre des relations entre les concepts. Néanmoins, il a une limitation lors de la conversion des cartes en une organisation formelle pour générer une ontologie. *OntoEdit* utilise une architecture client / serveur qui permet de générer des ontologies collaboratives, il est disponible sous deux versions : *OntoEdit Free* et *OntoEdit Professional*. Dernièrement une version récente *KAON (Karlsruhe Ontology)* est développée pour remplacer *OntoEdit*.

D'autres outils sont disponibles pour le développement d'ontologies réparties entre des environnements open source ou avec licence, à usage multiples ou uniques. En outre, il existe certains outils nécessitent un apprentissage et connaissance approfondie d'une langue particulière. Par contre, il existe une diversité des outils conçus à être exploités aisément et qui détiennent des interfaces graphiques. Plusieurs éditeurs sont manquants et d'autres restent en compétition, citons comme exemple : *WebOnto*, *Ontolingua Server*, *Ontosaurus*, *WebODE*, *TopBraid Composer*, *NeOn Toolkit*, *Apollo*, *Swoop*, *OntoGraf*, etc. Tableau 1 montre une comparaison de quelques éditeurs d'ontologie. Pour une lecture plus détaillée voir (Corcho et al., 2003), (Alatrish, 2013) et (Mizoguchi, 2004).

Type	Coopération	Ontologie Library	Expressivité	Consistance Check	Graphique notation	Extensibilité	Disponibilité
Protégé	Non	Oui	Oui	Oui	Oui	Oui	Open source
OntoStudio	Oui	Oui	Oui	Oui	Oui	Oui	Software license
OntoEdit	Oui	Oui	Oui	Oui	Oui	Oui	Software license
WebOnto	Oui	Oui	Oui	Oui	Oui	Non	Software license
Ontolingua Server	Oui	Oui	Non	Non	Non	Non	Open source
Ontosaurus	Oui	Oui	Oui	Oui	Non	Non	Open source
WebODE	Oui	Non	Non	Oui	Oui	Non	Open source

Tableau 1 Comparaison de différent éditeurs d` Ontologies

II.9 Conclusion

Le web sémantique devient une technologie efficace pour augmenter l'aspect représentatif des données dans le web. Il facilite l'exploitation, la recherche et l'échange de contenu en représentant ce contenu sémantiquement. Dans ce chapitre, nous avons présenté une vue globale sur le Web sémantique et les ontologies. Nous avons commencé par décrire l'architecture du Web et les composants de l'ontologie l'importance et les types d'ontologie. Ainsi que les critères pour construire une ontologie. Nous avons présenté les formalismes et langage de représentation qui est inévitables pour construire l'ontologie, les éditeurs pour manipuler les ontologies sont aussi évoqués.

Nous proposons donc, dans le chapitre suivant, une étude concernant l'intégration de la sémantique dans le Big Data et présentons les avantages et les défis qui peuvent rencontrer ce processus.

Chapitre III

Intégration de la Sémantique dans le Big Data

III.1 Introduction

Cette partie abordera l'objectif principal de cette thèse. Par conséquent, il est nécessaire de voir d'abord le sens de la notion d'intégration, on va exposer l'importance de l'ontologie pour le Big Data. Il est important de citer les différentes approches d'intégration d'ontologies pour qu'on puisse comprendre les étapes nécessaires et l'objectif de chaque approche. Par la suite, on va montrer comment l'intégration d'ontologie peut relever les défis du Big Data tout en les montrant les capacités offertes aux Big Data par l'intégration de l'ontologie. Une diversité des travaux avec une brève discussion est donnée à la fin de ce chapitre.

III.2 Des systèmes d'intégration de données vers l'intégration sémantique de données

Depuis l'avènement des bases de données au cours des années 1960, les chercheurs ont travaillé pour combiner les différentes sources de données hétérogènes en vue de fournir une interface de requête standard pour les interroger et assurer le partage de leurs données. Cette combinaison est effectuée selon la localisation des données : une intégration virtuelle de données par l'architecture de médiateur, et une intégration matérialisée de données dont les données sont stockées dans un entrepôt de données. Cependant, le web actuel et la prolifération massive du Big Data génère un nouveau défi dans l'hétérogénéité des sources. Cette hétérogénéité s'apparaît dans deux aspects : hétérogénéité des schémas et hétérogénéité des données.

Pour surmonter cette hétérogénéité des schémas et des données, les travaux se concentrent sur les informations présentes dans les schémas et les données par des technique de comparaison syntaxique comme les mesures de similarité, des heuristiques, etc. jusqu'à ce stade, le problème de l'hétérogénéité n'est pas encore résolu, donc une explicitation de la sémantique de données intégrées est essentielle (Pinto et al., 1999). Autrement dit, c'est l'intégration sémantique des données qui peut réduire l'impact de l'hétérogénéité des sources et rendre les données plus cohérentes et exploitables c'est-à-dire les requêtes peuvent être exprimées en termes d'une seule ontologie et aussi interrogées via une seule interface de requêtes (Pinto et al., 1999).

III.3 Nécessité de l'ontologie dans le Big Data

Aujourd'hui, le Big Data héberge de données massives générées à partir des millions de personnes et des systèmes stockés dans une myriade de sources (Saggi and Jain, 2018).

Mais elles restent toujours souffrir d'un manque dans l'aspect sémantique. L'impact apporté par le Big Data bouleverse l'industrie, la science et la société elle-même et conduit à réfléchir sérieusement pour adopter des nouvelles pratiques plus efficaces sur le plan commercial et les politiques gouvernementales, notamment dans les secteurs vitaux comme l'énergie, la santé, le transport, etc. (Koutsomitropoulos and Kalou, 2017). Cependant, les données sont souvent stockées sous un format incomplet, non structuré et hétérogène, ce qui rend une grande partie de données inaccessible aux utilisateurs. Cette situation fastidieuse oblige la communauté des chercheurs de trouver de technologies et d'outils permettant la recherche, l'exploitation et le partage de ces données afin de les rendre plus compréhensives et prêtes pour les décideurs (De Giacomo et al., 2018). L'intégration d'ontologies dans les Big Data s'avère la solution adéquate pour surmonter ce défi et enrichir les données du Big Data par la sémantique. En plus, assure une interopérabilité sémantique entre les systèmes utilisant le Big Data. Pour cela, les outils existants basés sur ETL ne sont plus suffisants pour envisager l'intégration de nombreuses sources de données distribuées, l'exploration de potentiel des ontologies et du Web sémantique peuvent réaliser les promesses du Big Data (Ostrowski et al., 2016).

III.3.1 L'intégration de la sémantique est-il un défi ?

Le stockage et le traitement des données dans le Big Data ne sont plus un souci pour les chercheurs, mais l'enjeu majeur qui les préoccupe est comment doter ces différentes données avec une sémantique permettant l'analyse et l'extraction des informations pertinentes. Il s'agit donc, d'une démarche vise à annoter et organiser les données et leurs métadonnées d'une façon où les concepts ayant un sens similaire non contradictoire (Obrst et al., 2014). Cela permettra les moteurs de recherche et les outils d'analyse de trouver et extraire les informations d'une manière sûre et efficace. En outre, Les ontologies offrent un moyen contribuant à faciliter la compréhension des informations échangées entre les systèmes interopérables (sémantiquement) à travers une perspective de standardisation de la représentation des concepts et de leurs relations (Sayah, Kazar and Ghenabzia, 2020).

III.3.2 Intégration d'ontologie au cours du processus de construction

Plusieurs méthodes proposées pour la construction d'ontologies dont chacune possède sa propre démarche. Néanmoins, la majorité partagent des étapes essentielles permettant d'aboutir l'objectif d'intégration des données provenant de diverses sources hétérogènes dans un schéma ou un modèle qui fournit une sémantique suffisante pour effectuer des

requêtes intelligentes et concevoir des applications plus performantes. Cet objectif reste un débat ouvert dans le domaine de Big Data et le Web sémantique. Dans ce contexte, le processus de construction souvent composé par les étapes suivantes (Pinto and Martins, 2001):

1. Spécification : identifier l'objectif et déterminer l'utilisation future de l'ontologie.
2. Conceptualisation : Structuration de la connaissance du domaine dans un modèle conceptuel.
3. Formalisation : transformation du modèle conceptuel en modèle formel.
4. Intégration : réutilisation d'ontologies existantes s'il est possible pour accélérer la démarche de développement.
5. Implémentation : construction d'un modèle opératoire utilisable par un ordinateur et validation.
6. Maintenance : mise à jour de l'ontologie en cas de besoin.

Bien que l'étape d'intégration soit primordiale pour construire une ontologie où l'importance des activités d'intégration dans le processus de construction est critique, la question qui se pose est comment l'intégration devrait-elle être réalisée ?

III.3.3 Que signifie l'intégration d'ontologies

Les ontologies sont considérées comme un cadre formel utilisé pour doter un ensemble de données d'une sémantique qui permet non seulement de décrire et représenter ces données, mais les rendre aussi compréhensives et partageable entre des différents systèmes et fournit un potentiel de communication et interopérabilité entre eux. À cet égard, la sémantique se concentre sur l'organisation et l'action de l'information qui est vu comme un intermédiaire entre des sources de données hétérogènes, ce qui peut entrer en conflit non seulement avec la structure mais aussi avec le contexte ou la valeur de données.

Pour cela, il est évident d'abord de comprendre la signification de l'intégration dans le domaine d'ontologie. L'intégration de données concerne l'unification de données partageant certaines sémantiques communes mais provenant de sources non apparentées. Il s'agit de combiner des données de manière à ce qu'une vue uniforme soit disponible pour les utilisateurs (Abbes and Gargouri, 2018). Par ailleurs, ce mot réfère à trois significations (Ghafour, 2004):

- ✓ **Intégration** Intégration d'ontologies lors de la construction d'une nouvelle ontologie en réutilisant d'autres ontologies existantes répondant aux exigences appropriées par l'ontologie envisagée. Parfois on construit une ontologie complète par l'assemblage d'autres ontologies, ces ontologies doivent satisfaire la spécification ou l'adaptation, etc. à la nouvelle ontologie.
- ✓ **Fusion** Sert à construire une nouvelle ontologie par la fusion (merging) de différentes ontologies existantes dans une seule ontologie qui les unifie qui exploite leurs capacités par la fusion des idées, des concepts, des axiomes, etc. Autrement dit, mêler les connaissances des différentes ontologies dans une seule vise le même sujet afin d'unifier les concepts, les terminologies, les définitions, etc. (Pinto et al., 1999).
- ✓ **Utilisation** l'idée de base est d'utiliser une ou plusieurs ontologies partagées entre différentes applications dans une application destinée pour spécifier ou implémenter un système à base de connaissance (KBS), généralement partage les mêmes types de connaissances utilisées dans les différentes applications, mais chacune est dédiée pour des besoins spécifiques (Pinto et al., 1999).

III.3.4 Processus d'intégration d'ontologies

Les ontologies sont introduites dans les systèmes d'intégration matérialisés et virtuels afin de résoudre les conflits sémantique et syntaxique. Ce conflit est géré par trois approches (Guido and Paiano, 2010):

1. Manuelle à l'aide d'un expert humain (le système *Tsimmis*).
2. Semi-automatique qui se base sur les ontologies linguistique (le système *Momis*, *WordNet*).
3. Automatique par l'incorporation des ontologies conceptuelles (le projet *Buster*, *Picxel* ou *SHOE*).

L'utilisation des ontologies est étendue aux tâches de conception (modélisation conceptuelle, multidimensionnelle, processus ETL, etc.) qui a offert une gestion efficace et automatique des conflits rencontrés (Djilani, 2017). Le choix des ontologies candidates pour le processus d'intégration se fait que ce soit en trouvant des ontologies disponibles, ou bien choisir parmi les ontologies disponibles celles qui sont éligibles (Pinto and Martins, 2001).

Le processus d'intégration d'ontologie comprend des différentes étapes :

1. Identifier la possibilité d'intégration, les modules nécessaires pour construire la future ontologie, identifie les hypothèses et les engagements ontologiques, identifient les connaissances à représenter dans chaque module.
2. Identifiez les ontologies candidates soit en : recherchant des ontologies disponibles, soit en choisissant parmi les ontologies disponibles celles qui sont éligibles.
3. obtenir les ontologies candidates sous une forme appropriée qui comprend leurs représentations et toute la documentation disponible.
4. Etude et analyse des ontologies candidates à travers deux activités : l'évaluation technique des ontologies candidate par des experts du domaine à l'aide des critères spécialisés et l'évaluation des ontologies candidates par les ontologues avec de critères spécialisés orientés vers l'intégration.
5. Choisir les ontologies sources dont les fonctionnalités de développement sont liées à la manière dont l'ontologie a été construite (Horrocks et al., 2016).
6. Appliquer les opérations d'intégration par l'une des opérations : réutilisation, adaptation, spécialisation et la généralisation.
7. Analyser l'ontologie résultante après l'intégration des connaissances afin d'évaluer et analyser l'ontologie qui en résulte.

III.3.5 Comment peut-on améliorer l'intégration

Pour une intégration efficace des données provenant de différentes sources hétérogènes vers des modèles de données significatifs permettant d'effectuer des requêtes plus intelligentes afin de faciliter le développement des applications. Cela, reste un défi majeur dans le domaine de Big Data. Pour rapprocher à cette ambition, un expert humain chargé de l'identification du domaine et l'analyse manuelle des données est obligatoire. Malheureusement, ce processus prend beaucoup de temps (Bansal and Kagemann, 2015). Donc, la nécessité à des algorithmes capable de générer des modèles de données sémantique d'une façon automatique est devient primordial. Ainsi, des suggestions peuvent améliorer l'intégration, à savoir:

- ✓ Identifier les ontologies existantes par un processus en cours d'intégration de données.
- ✓ Réutiliser une ontologie existante avec des classes, propriétés et relations pertinentes ou bien concevoir une nouvelle ontologie.
- ✓ Effectuer l'alignement des ontologies existantes par des règles conçus pour des bases de données.
- ✓ Produire des règles d'alignement entre le modèle de données sémantique concerné et les ontologies existantes tels que *FOAF*, *DBPedia* et *Wordnet*.

III.4 Approches de construction des ontologies

Différentes approches développées pour la mise en place d'une ontologie en vue d'explicitier la sémantique de l'information source. Chaque architecture affecte profondément le formalisme de représentation de l'ontologie. Alors, le choix d'une alternative est adapté aux besoins de l'application développée. Dans le domaine d'intégration de données, trois variantes possibles à savoir : l'approche mono-ontologie, l'approche multi-ontologies et l'approche hybride (Wache et al., 2001).

III.4.1 Approche mono-ontologie (single-ontology approach)

Cette approche consiste à employer une seule ontologie répartie entre les différents systèmes à base de connaissances, l'utilisation d'un vocabulaire partagé est destinée pour la spécification de la sémantique (Laallam et al., 2014). L'ontologie globale relie toutes les sources d'informations dont chaque source possède un modèle indépendant établit en vue de relier ces objets au modèle globale (figure 13) (Ghafour, 2004). L'approche *SIMS* et *PicseI* sont des exemples de ce type. L'inconvénient de cette structure d'intégration est la nécessité de redéfinir l'ontologie en cas d'ajout d'une nouvelle source, ainsi l'absence d'une autonomie schématique des sources (Djilani, 2017).

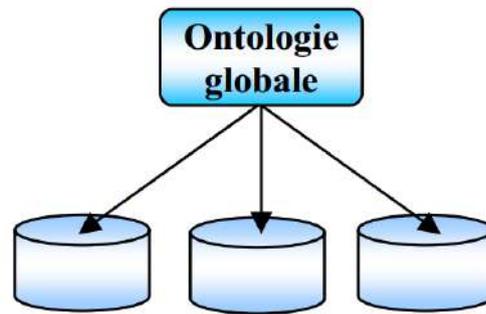


Figure 13 Approche mono-ontologie.

III.4.2 Approche multi-ontologies

Parfois la modification des sources peut affecter l'ontologie globale dans un système interopérable, l'approche multi-ontologies sert à doter chaque source d'informations par une ontologie locale et fournit une liberté de définition pour cette source sans tenu compte aux autres ontologies (figure 14). Cette approche est utilisée dans le cas où il est impossible de trouver une ontologie consensuelle causé par la différence sémantique flagrante entre les systèmes. Cependant, l'absence d'un vocabulaire commun s'affiche comme une contrainte majeure limitant la communication entre les sources. Pour surmonter cette limitation, un mapping inter-ontologie est mis en place afin d'effectuer une correspondance entre les différents termes d'ontologies (Ding et al., 2007). Malheureusement ce mapping est difficile à concrétiser à cause de l'hétérogénéité sémantique qui peut se rencontrer et le degré de complexité de mapping : pour N sources, la complexité est de $N(N-1)/2$ (Horrocks et al., 2016). L'approche OBSERVER est utilisée dans ce contexte (Wache et al., 2001).

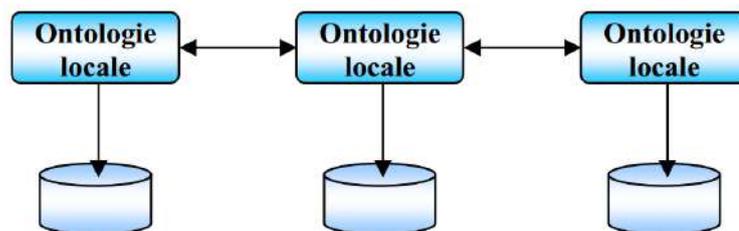


Figure 14 Approche multi-ontologies.

III.4.3 Approche hybride

C'est une combinaison de deux approches précédentes qui utilise les ontologies locales et partagés dont la sémantique de chaque source est définie par une ontologie locale (figure 14), un vocabulaire commun est construit afin de faciliter le mapping et rendre les différentes sources d'ontologies comparable (Laallam et al., 2014). L'alignement entre les ontologies locales et l'ontologie partagée peut se faire a priori ou a posteriori (Djilani, 2017). Ce vocabulaire est composé par des termes et primitives de domaine d'une façon où ces primitives construisent des termes complexes en les combinant avec des opérateurs, ce qui rend les termes comparable et partageable entre les ontologies sans aucun conflit (Wache et al., 2001).

Ce vocabulaire partagé est représenté sous forme d'ontologie, le projet *COIN* représente le contexte d'information par un vecteur de valeurs d'attributs, *METACOTA* annote chaque source par des labels. Les sources d'ontologies est vue comme un raffinement de l'ontologie générale dans *BUSTER*. L'apport ajouté par cette approche est la possibilité d'ajouter des nouvelles sources sans affecter ou modifier le vocabulaire commun, ainsi elle assure l'évolution de l'ontologie (Ghafour, 2004).

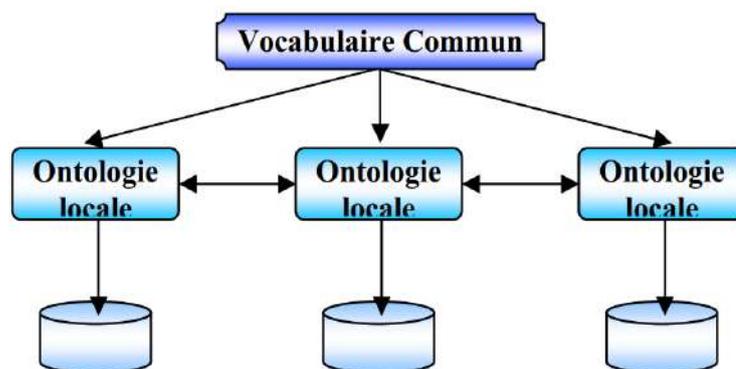


Figure 15 Approche Hybride.

Le tableau 2 illustre une comparaison entre les trois approches d'intégration vues précédemment :

Approche d'intégration d'ontologie	Nombre d'ontologies utilisés	Vocabulaire utilisé	Modèle d'ontologie	Faisabilité d'implémentation	Avantages	Inconvénients	Exemples
Approche mono-ontologie	Une seule pour toutes les sources (globales)	Partagé	Un seul modèle Reliant toutes les sources	Mis en œuvre Difficile	- Interopérabilité entre les différentes sources - Possibilité de mapping entre les sources	- Absence d'autonomie des sources - Extensibilité nécessite la redéfinition de l'ontologie	SIMS Picstel
Approche multi-ontologie	Ontologie locale pour Chaque source	Approprié pour chaque source et (normalisé avec les autres sources)	Indépendant pour chaque source + schéma inter-mapping	Mise en œuvre facile (Sauf le processus de normalisation)	- Autonomie de définition pour chaque source - Extensibilité de source	- Difficulté de mapping entre les sources - Manque de vocabulaire commun	OBSERVER
Approche hybride	Ontologies locales + globale	Appropriée + partagé global	Modèle hybride	Mis en œuvre relativement facile	- Possibilité de mapping - Extensibilité des sources - Communication entre les sources	- Difficulté de réutilisé les ontologies locales	COIN BUSTER MECOTA

Tableau 2 Une comparaison entre les trois approches d'intégration.

III.5 Rôle des ontologies dans le Big Data

Les ontologies jouent un rôle majeur dans le Web sémantique (Ding et al., 2007) qui sont davantage appliquées pour faciliter le traitement et la compréhension des données du Big Data, ainsi pour réduire les hétérogénéités syntaxiques et sémantiques (Djilani, 2017). L'intégration sémantique peut automatiser la communication entre les différents systèmes informatiques, ce processus aborde les problèmes engendrés par les aspects du Big Data tel que la variété car tout logiciel qui utilise des Big Data doit s'assurer qu'il n'y a pas de conflit sémantique. Ainsi, la variété peut être atténuée par l'annotation des données et l'utilisation de métadonnées partagées. De plus, la normalisation des termes joue un rôle important qui facilite leurs réutilisations.

L'utilisation des ontologies pour l'analyse de données montre une efficacité élevée dans la gestion des données. Actuellement, la plupart des projets de Big Data traitent les données de manière ponctuelle plutôt que systématique. Il existe des ontologies pour décrire des données, comme *PROV-O* ontologie. Développement ontologies standard pour des modèles de processus, tels que la boucle *OODA* et les modèles de fusion *JDL / DFIG* (Obrst et al., 2014). Les techniques d'analyse du Big Data fournissent une certaine sémantique qui facilitera considérablement la diffusion des résultats, qui vont être corrélé et lié à des schémas sémantiques plus vastes.

III.5.1 Le mapping comme solution pour l'intégration d'ontologies

La réutilisation d'ontologie nécessite généralement l'intégration de plusieurs ontologies, qui s'apparait aussi difficile qu'un développement d'une nouvelle ontologie. L'idée clé est la création de modules intégrables qui fusionnent la sémantique des composants réutilisés. Le rôle de mapping qui consiste à relier les ontologies au contenu réel d'une source d'information est très important dans la réutilisation lorsqu'il existe plusieurs ontologies potentiellement réutilisées (Wache et al., 2001). Selon *Noy* « le mapping d'ontologies est un processus qui spécifie une convergence sémantique entre différentes ontologies afin d'en extraire » (Pinto et al., 1999). Les correspondances entre certaines entités L'intégration vise à diminuer le problème de variété dans le Big Data, où les ontologies peuvent aider par l'annotation des données et des métadonnées.

Dans des nombreuses applications Big Data, les terminologies utilisées peuvent avoir une interprétation différente. Pour cela, les ontologies possèdent le potentiel pour atténuer grandement ce problème, en fournissant un modèle standard, indépendant de la représentation de données particulières et terminologies par le processus de mapping qui joue un rôle important pour atténuer l'hétérogénéité entre les différentes ontologies en les alignant à travers d'une correspondance sémantique entre les entités de ces ontologies afin d'assurer l'interopérabilité sémantique (Hui et al., 2018). Le Web sémantique, la communication entre agents, la composition des services Web, etc. utilisent ce principe (Djilani, 2017).

III.5.2 Comment l'intégration d'ontologie peut relever les défis du Big Data ?

L'évolution massive dans les sources inondant le web par une immense variété de données tels que les ordinateurs connectés, les appareils mobiles, les réseaux sociaux, les capteurs, etc. dans plusieurs domaines comme l'IoT, la santé, l'environnement, le transport, la

gestion des catastrophes, etc. les enjeux de modélisation et de gestion de ces systèmes sont organisés autour de 5Vs de Big Data (volume, variété, vélocité, véracité et valeur). L'intégration de la sémantique dans cet environnement peut adresser les défis du Big Data (Thirunarayan and Sheth, 2013), (Ostrowski et al., 2016):

- ✓ Pour relever le défi du volume, le changement de niveau d'abstraction des données de bas niveau en abstraction de plus haut niveau rendant les informations plus significatives et adaptées à la prise de décision. Cette tâche est réalisée à l'aide d'une perception sémantique, qui vise à intégrer la sémantique pour effectuer l'inférence perceptuelle afin d'extraire des informations pertinentes et exploitables (Sayah, Kazar and Ghenabzia, 2020).
- ✓ Pour relever le défi de Variété, l'utilisation de métadonnées sémantiques et des modèles sémantique assurant une annotation des données pour décrire et intégrer les données. Cela, peut faciliter l'interopérabilité des systèmes et surmonter l'hétérogénéité sémantique et syntaxique des données, surtout si l'annotation de données est générée automatiquement (Koutsomitropoulos and Kalou, 2017).
- ✓ Pour gérer le défi de la vitesse, les chercheurs travaillent pour utiliser la possibilité d'une sémantique contenue afin de créer des modèles spécifiques capables de connaître les nouveaux concepts et les entités rencontrés. Néanmoins, des algorithmes online sont très recommandés pour l'analyse et le filtrage des données en temps réel (Bermúdez-Edo et al., 2016).
- ✓ Pour gérer la Véracité, l'exploration des modèles sémantiques de confiance et les méthodes garantissant la fiabilité par la vérification des données multimodales en utilisant des contraintes sémantiques (Storey and Song, 2017). De faite que les informations exploitables provenant de sources multiples nécessitent l'abstraction et l'intégration de données hétérogènes et parfois conflictuelles et non fiables. L'intégration sémantique peut résoudre cette fragilité.
- ✓ La valeur des données du Big Data est parfaitement évoluée par l'implication de l'aspect sémantique qui a enrichi les modèles sémantiques qui devient plus expressifs et complets. En outre, les bases de connaissances sont davantage renforcées et participent efficacement dans l'intelligence décisionnelle et aident énormément à la prise de décision.

III.6 Comparaison des travaux basés sur l'intégration d'ontologies dans le Big Data

Dans cette partie, nous citons quelques travaux de différents domaines d'application tels que : healthcare, l'industrie, la gestion de données d'entreprises, les réseaux sociaux, IoT, sécurité et défense (tableau 3). Ces travaux partagent l'idée de l'intégration d'ontologies dans le Big Data. Toutefois, chaque auteur a utilisé des outils Big Data et techniques ontologiques spécifiques au domaine exploré selon les besoins et les objectifs sollicités. Le tableau suivant montre une diversité des travaux dont certains critères sont présentés afin de donner une idée générale décrivant le sujet abordé, les outils Big Data utilisés avec les techniques d'ontologie choisies, également une description de l'approche et les étapes suivies pour développer chaque proposition sont présentés. Il est très évident de noter les avantages apportés par ces travaux ainsi les inconvénients et les contraintes rencontrés.

Tableau 3 Comparaison des travaux basés sur l'intégration de la sémantique dans le Big Data.

Sujet d'article auteur, domaine	Description de travail	Outils Big Data	Ontologies utilisé	Autre outils	Approche utilisée	Avantages Apportés	Inconvénients rencontrés
(Mezghani et al., 2015) Healthcare	Développement d'une architecture enrichissant le model NIST Big Data d'une sémantique générique afin de comprendre intelligemment les données collectées à travers des dispositifs portables qui vont assister les médecins à superviser l'évolution de la santé du patient et mettre à jour le patient de son état de santé. (Pour les diabétiques)	-NIST Big Data -Cloud environment -Data processing algorithms -MapReduce	WH_Ontology -RDF -SPARQL	-Semantic MediaWiki (SMW) -KaaS - Apache Jena Fuseki	Application KaaS pour intégrer les données provenant de multiples dispositifs portables par la description d'une couche de connaissances sémantiques et l'utilisation de SMW avec d'autres extensions pour annoter les caractéristiques des dispositifs portables et visualiser le profil du patient.	-offrir une solution évolutive pour stocker le grand volume de données sur les soins de santé générées par plusieurs sources, -soutenir le partage et l'intégration des données pour une meilleure prise de décision préventive. -extraire des informations précieuses à partir de données hétérogènes.	- la sécurité et la confidentialité lors de la transmission de données restent un défi. - difficulté pour l'extensibilité de KaaS portable pour des autres scénarios
(Williams et al., 2015) Industrie	Développement d'une infrastructure Big Data par l'utilisation de la sémantique pour créer un modèle de domaine qui offre un accès aux données unifiées et permettre de tirer plus de valeurs des données et effectuer l'analyse afin d'améliorer la conception d'ingénierie de turbines à gaz.	-Hive -HDFS	-SPARQL -RDF triple store -URI -Semantic triple store Virtuoso	-SPARQL graph - SQL-like - Python-based analytics	la couche de stockage contient le modèle sémantique et les données instanciées pour effectuer le test et le mappage de variables, l'interface utilisateur basée web permet de créer des requêtes exécutés sur la base sémantique et la base de séries temporelles. L'utilisateur peut effectuer des analyses sur les données de séries temporelles	- Le système offre une importante économie de productivité et de dépenses. - l'architecture est extensible horizontalement pour le stockage de données de séries temporelles -le système offre une rapidité de traitement et analyse et permet de tirer plus de valeur des données sur des années afin d'améliorer les conceptions techniques.	- cette architecture est limitée pour un type de données. - Le système ne supporte pas le stockage à des autres plateformes NoSQL, HBase ou Apache Cassandra qui permettant l'extensibilité. -Le système n'explore pas une sémantique évolutive

<p>(Abbes and Gargouri, 2018)</p> <p>Gestion de données</p>	<p>Illustration d'une approche pour construire une ontologie modulaire intégrant le Big Data par l'exploitation d'une base de données NOSQL (MongoDB) afin de composer une ontologie globale à partir d'encapsuler les sources de données vers la base MongoDB et la génération des ontologies locales (scénario proposé pour deux entreprises)</p>	<p>-MongoDB NOSQL database</p>	<p>- OWL - OWL -DL</p>	<p>- Talend - MOOM - M2Onto - DBRef (MongoDB API) - DBList</p>	<p>il commence par intégration au niveau du schéma en normalisant les données des sources dans les schémas utilisés (MongoDB) pour représenter les données ensuite le mappage de données MongoDB à un module ontologie OWL afin de les fusionner vers un module d'ontologie global</p>	<p>- l'intégration de données offre une vue intégrée qui facilite l'accès et la réutilisation l'information. -approche est applicable dans plusieurs domaines. -La possibilité d'intégrer des nouvelles sources qui sont émerger dans l'ontologie globale. (Evolution horizontale)</p>	<p>-difficulté de mise à jour de l'ontologie globale. -Pas d'évolution verticale pour les sources de données. - lenteur pour gérer un grand nombre des sources (absence des outils Big Data ex hadoop).</p>
<p>(Hamdouni et al., 2018)</p> <p>Réseaux sociaux</p>	<p>Ce travail basé sur l'ontologie et les bases de données NoSQL combine analyse des sentiments (textuel et visuel) afin de tirer les émotions exprimées dans les données exploitées à partir des réseaux sociaux pour estimer la réaction du public envers un sujet spécifique (élections françaises 2017). Ce qui permet de classer les candidats (sondage) et détecter les régions où ils sont en tête.</p>	<p>-MongoDB -NoSQL database engine -Spring Data MongoDB</p>	<p>-SenticNet -SentiBank -JSON</p>	<p>-Spring Data MongoDB -Twitter Streaming API -Graph ExplorerAPI -spring boot</p>	<p>-l'extraction des données via les APIs de Facebook et Twitter basées sur des mots clés, stockage ces données dans MongoDB. -Implémentation des paradigmes sémantique et d'ontologies de domaines. -Analyse de données par le raisonnement sur les ontologies prédéfinies afin de produire les résultats des sentiments.</p>	<p>-la combinaison de web sémantique et les outils d'analyse de sentiments offrent des résultats exacts sur les émotions de public. -le système peut être utilisé dans des autres domaines pour évaluer les émotions et les sentiments qui peuvent être utilisés à diriger cette opinion. -le système offre un outil de prévision de comportement social.</p>	<p>-le système se limite à l'analyse de données spécifiques (texte, image) -le système incapable de faire de traitement en temps réel. -le système ne peut pas déterminer les différentes communautés formant les réseaux sociaux.</p>

<p>(Sezer et al., 2018)</p> <p>IoT</p>	<p>Proposition d'un cadre augmenté intégrant les technologies web sémantiques, les Big Data et IoT. Avec de nouvelles fonctionnalités sémantique et analyse de Big Data par un module d'apprentissage, ce système fourni un soutien efficace à tous les types de capteurs en vue de stocker et appliquer un raisonnement sur les données pour avoir des meilleurs résultats</p>	<ul style="list-style-type: none"> - NoSQL databases - Apache Spark 	<ul style="list-style-type: none"> - SSN ontology - RDF, RDFS - JSON-LD 	<ul style="list-style-type: none"> - RDFS reasoning rules - Jena rule syntax - machine learning methods 	<p>Acquisition de différents types données provenant des capteurs, application le processus ETL sur les données afin de les sémantiser par les outils RDF et SSN, puis appliquer les règles de raisonnement sur les données RDF pour servir les objectifs de systèmes, suivie par une phase d'apprentissage à l'aide de machine learning pour améliorer les résultats. Enfin, exécuter les actions</p>	<ul style="list-style-type: none"> - Ce cadre fourni des meilleures caractéristiques pour un réseau de capteurs grâce à l'utilisation de la sémantique et les outils Big Data. -Ce système offre un cadre évolutif, intelligent et interopérable supportant des différents domaines d'application. -l'usage d'apprentissage et le raisonnement peut améliorer les résultats des systèmes envisagés. 	<ul style="list-style-type: none"> -la difficulté pour choisir les outils et les méthodes adéquates pour implémenter un tel système. -l'adaptation de différentes technologies dans un domaine nécessite une stratégie puissante. -la difficulté de l'intégration de différents types de capteurs avec l'hétérogénéité des données -le défi de disponibilité et la sécurité des données.
<p>(Anne-Claire Boury-Brisset, 2013)</p> <p>Security (intelligence)</p>	<p>Proposition d'une plate-forme d'analyse et d'intégration de données de renseignement évolutif (MIDIS) par l'utilisation de technologies web sémantique et Big Data visant à améliorer l'intégration et l'analyse de données hétérogènes et fournir des informations opportunes et pertinentes aux services de sécurité et de renseignement grâce à la recherche intuitive et les mécanismes de découverte</p>	<ul style="list-style-type: none"> -HDFS -HBase 	<ul style="list-style-type: none"> -Semantic annotation -mapping entre les termes de l'ontologie -RDF -SPARQL 	<ul style="list-style-type: none"> -GATE Platform -SOA Technologies -plateforme cloudera 	<ul style="list-style-type: none"> -acquisition de données à partir de sources hétérogènes et les intégrées dans des segments de stockage unifié (HDFS, HBase) -Enrichissement sémantique basé sur l'ontologie (mapping et annotation) -Interrogation de données et analyse (recherche, filtrage, notification, alerte) - Interactions avec des modules de raisonnement externes 	<p>Objectifs :</p> <ul style="list-style-type: none"> - Fournir des informations opportunes et pertinentes à l'analyste grâce à des mécanismes intuitifs de recherche et de découverte. -Fournir un cadre facilitant l'intégration de données hétérogènes et la préparation de diverses données à l'analyse pour une meilleure connaissance des situations 	<ul style="list-style-type: none"> -l'enrichissement sémantique offre qu'une évolution horizontale d'intégration de données. -manque de module d'apprentissage peut diminuer l'efficacité de système. -la plate-forme n'exploite pas les données de réseaux sociaux qui sont des informations critiques et importantes pour ce domaine.

III.7 Discussion

Les travaux présentés ci-dessus couvrent des divers domaines d'actualité où la quantité et la variété de données est un aspect important caractérisant la manipulation et l'analyse de données traitées, ce défi est résolu à l'aide des moyens et des capacités offertes par les outils Big Data (*Hadoop, HDFS, MapReduce, Hbase, Hive, MongoDB, etc.*). La nécessité de remédier l'hétérogénéité de différents types de données générées à partir d'une multitude ressources exige l'emploi du paradigme d'ontologie, qui peut fournir des données intelligentes facile à partager et réutiliser de faite qu'elles sont standardisées et harmonisées. De plus, il fait ressortir la sémantique cachée dans l'immense base de données qui n'est pas directement disponible à partir des sources distribuées. À ce stade les données sémantisées peuvent être communiqué entre des individus bien qu'entre des systèmes, ce qui garantir l'interopérabilité et enrichir les fonctionnalités des applications par des nouveaux services. En outre, l'ajout des modules d'inférence fourni des systèmes intelligents capables d'extraire des nouvelles connaissances et avoir un certain niveau d'autonomie. La majorité de travaux illustrés introduisent des outils Big Data pour le traitement et le stockage de données, et utilisent une ontologie de domaine pour couvrir la sémantique de ces données.

Dans (Mezghani et al., 2015) les auteurs proposent une plate-forme pour l'intégration de données mobiles hétérogènes dans le domaine de la santé. Dans cette architecture, ils ont utilisé l'environnement cloud et MapReduce avec WH ontologie en plus de l'application *KaaS* pour intégrer les données provenant de multiples dispositifs portables par la description d'une couche de connaissances sémantiques et l'utilisation de SMW avec d'autres extensions pour annoter les caractéristiques des dispositifs portables et visualiser le profil du patient.

(Williams et al., 2015) traite le secteur de l'industrie, ils ont développé une infrastructure Big Data manipulé par *HDFS* et *Hive*, ils ont utilisé les technologies sémantiques (*SPARQL, RDF triple store, semantic triple store Virtuoso*) pour créer un modèle de domaine qui offre un accès aux données unifiées et permettre de tirer plus de valeurs des données et effectuer l'analyse afin d'améliorer la conception d'ingénierie de turbines à gaz.

(Abbes and Gargouri, 2018) propose une approche pour construire une ontologie modulaire intégrant le Big Data par l'exploitation d'une base de données NOSQL

(MongoDB) afin de composer une ontologie globale à partir d'encapsuler les sources de données vers la base MongoDB et la génération des ontologies locales (scénario proposé pour deux entreprises), OWL DL est utilisé avec d'autres outils.

(Hamdouni et al., 2018) touche un autre domaine (les réseaux sociaux) dont le travail est basé sur l'ontologies et les bases de données NoSQL (MongoDB), la combinaison de l'analyse des sentiments (textuel et visuel) a permis de tirer les émotions exprimées dans les données exploitées à partir des réseaux sociaux en vue d'estimer la réaction du public envers un sujet spécifique (élections françaises 2017) par l'implémentation des paradigmes sémantique et d'ontologies de domaines. Ce qui a permis de classer les candidats (sondage) et connaître les régions de forte représentativité pour chaque candidat.

Les auteurs (Sezer et al., 2018) ont développé un cadre augmenté intégrant les technologies web sémantiques, le Big Data et IoT. Avec de nouvelles fonctionnalités sémantique et analyse de Big Data par un module d'apprentissage, ce système fourni un soutien efficace à tous les types de capteurs en vue de stocker et appliquer un raisonnement sur les données pour avoir des meilleurs résultats. L'utilisation de base de données NoSQL, Apache Spark, SSN ontology, RDF, RDFS, RDFS reasoning rules, etc. ont clairement amélioré les performances du système.

Dans (Boury-Brisset, 2013) l'auteur conçoit une plate-forme d'analyse et d'intégration de données de renseignement évolutif (MIDIS) par l'utilisation de technologies web sémantique RDF (annotation, mapping) et Big Data (HDFS, Hbase) visant à améliorer l'intégration et l'analyse de données hétérogènes et fournir des informations opportunes et pertinentes aux services de sécurité et de renseignement grâce à la recherche intuitive et les mécanismes de découverte.

III.8 Conclusion

Dans ce chapitre nous avons abordé le concept d'intégration de la sémantique dans le Big Data tout en soulignant le rôle et l'importance de ce processus, les approches d'ontologies utilisées, cette partie est conclu par une comparaison et discussion de quelques travaux qui ont utilisé les techniques d'intégration dans des domaines d'actualité.

Chapitre IV

Modélisation de l'approche

« Un Système Intelligent de Gestion de l'Energie dans les Villes Intelligentes (ISEM-SC) basé sur les Big Data et l'Ontologie »

IV.1 Introduction

Pour valider notre contribution, ce chapitre est consacré pour l'approche proposé. Cette approche se focalise sur les villes intelligentes comme exemple contemporain. Nous proposons une solution intelligente pour faire face aux défis urbains dans la vie réelle des citoyens. La réduction de la consommation d'énergie est l'un des domaines les plus pertinents pour le développement et la durabilité des villes intelligentes (Ejaz et al.,2017). Dans ce contexte, nous proposons une architecture de diverses technologies coopérant ensemble pour fournir une optimisation énergétique efficace. Non seulement pour les maisons individuelles, mais aussi pour la ville entière. L'architecture proposée est essentiellement basée sur l'ontologie des maisons intelligents (Onto-SB), qui est un outil puissant pour représenter les connaissances du domaine et fournit un cadre structurel pour organiser les données sur les bâtiments intelligents (Degha et al., 2019). L'Onto-SB permet le raisonnement en représentant formellement la connaissance du domaine. En outre, nous utilisons le système multi-agents (MAS) pour offrir un rôle fiable, qui fournit une coopération efficace et l'autonomie nécessaire pour gérer les énormes données générées et échangées entre les différents acteurs du système (Ma et al., 2019).

L'utilisation des technologies Big Data offre un mécanisme fiable pour le stockage et le traitement des données. Dans cette partie, nous allons expliquer les différentes couches du système proposé, une description détaillée de l'environnement d'expérimentation sera donné avec tous les caractéristiques des appareils et les acteurs participant dans le scénario.

IV.2 Contexte de l'étude

Cette section vise à présenter les principaux aspects de ce travail et à aider le lecteur à situer l'étude dans le contexte de la recherche. En outre, elle fournit un bref aperçu résumant les principaux thèmes de recherche utilisés tels que les Big Data, la sémantique et les MAS.

IV.2.1 Intégration sémantique dans les Big Data pour les villes intelligentes

Une évolution colossale des sources de données dans les villes intelligentes génère une grande variété de données à travers les capteurs et les appareils installés. Les défis

rencontrés dans la modélisation et la gestion des systèmes dans ces domaines apparaissent avec les 5V de Big Data. Les nouvelles exigences en matière de Big Data en termes de data mining sont de plus en plus différentes des systèmes traditionnels d'entreposage et d'extraction de données. Les données hétérogènes et non structurées font appel à de nouvelles technologies pour assurer un traitement et une analyse efficaces des données. Pour permettre des modèles cachés et en tirer de la valeur, diverses méthodes sont utilisées, telles que les arbres de décision, les réseaux neuronaux et le SVM. D'autre part, les Big Data associées au cloud computing offrent des opportunités prometteuses en fournissant une haute disponibilité et une grande élasticité avec un faible coût, une faible durée d'exécution et permet le déploiement de nouvelles applications (Merizig *et al.*, 2019). En outre, il permet de partager les ressources, y compris des logiciels de traitement, de stockage, de mise en réseau et d'analyse (Yang *et al.*, 2017).

Cependant, les chercheurs dans le domaine des villes intelligentes étaient inquiets de savoir comment combiner les différentes sources de données hétérogènes tout en fournissant une interface de requête standard. En effet, l'intégration sémantique offre de nombreuses possibilités prometteuses et exploite la puissance des bases de connaissances sémantiques (Beneventano and Vincini, 2019). Elle réduit comparativement l'hétérogénéité des sources, offre une cohérence des données et garantit une interopérabilité fiable entre les différents systèmes (Abbes and Gargouri, 2018). En outre, il permet de partager et de réutiliser les connaissances en normalisant le vocabulaire, en fournissant des informations pertinentes pour la recherche et en exécutant des requêtes en langage naturel. En outre, il diminue le trafic d'informations à travers le réseau en cartographiant (mapping) l'ontologie dans les sources de données et offre une inférence perceptive (Calvanese, 2015). Les ontologies peuvent déployer des outils sémantiques de gestion des risques de sécurité.

Néanmoins, lorsque les ontologies se développent, de nombreux déficits apparaissent, en particulier le rapport entre la taille des instances créées et la mémoire de travail réservée, ce qui peut affecter la performance et la flexibilité du système (temps de fonctionnement). L'écosystème Big Data offre donc les outils nécessaires pour améliorer les systèmes basés sur les ontologies. Alors que les *HDFS*, *HBase* et *MongoDB* permettent de stocker un grand volume d'instances d'ontologies et de fournir un espace de travail suffisant quelle que soit leur taille (Bhadani and Jothimani, 2016). En outre, *MapReduce*, *Storm* et *Spark* peuvent assurer un cadre de traitement rapide. L'utilisation de bases de données *NoSQL* et de langages de requête tels que *SPARQL* permet de prendre en charge les structures complexes

des ontologies liées aux applications Big Data. En outre, il permet une gestion fiable et efficace des données (Sayah et al., 2018).

IV.3 Économie de l'énergie dans les villes intelligentes

Les scientifiques prévoient que d'ici 2050, environ 70 % de la population mondiale résidera dans des villes (Dritsa and Biloria, 2018). Par conséquent, les villes auront besoin de technologies intelligentes capables de résoudre les problèmes de durabilité liés au développement de la consommation d'énergie (Wang and Moriarty, 2019). Une ville intelligente est considérée comme un écosystème intégré doté de technologies puissantes, qui visent à rendre les villes plus durables. En fait, les villes intelligentes sont considérées comme l'une des applications Big Data les plus importantes et un domaine actif de l'IoT (Shafik et al., 2020). Les chercheurs ont consacré une partie de leurs travaux sur les villes intelligentes à la réduction de la consommation d'énergie (Soomro et al., 2019). En particulier avec la croissance de l'utilisation des appareils dans la vie quotidienne des gens (Butt et al., 2019). Sachant que la demande mondiale d'énergie ne cesse d'augmenter, il devient nécessaire de trouver de nouvelles solutions qui assurent des économies d'énergie et offrent une vie confortable aux habitants sans nuire à la protection de l'environnement (Samuel, 2016).

IV.3.1 Travaux connexes

La croissance exponentielle du volume de données à obliger les chercheurs de s'occuper sérieusement des nouveaux aspects des Big Data (5 V). Diverses technologies de pointe ont été développées pour faciliter la gestion, le stockage, l'analyse et le partage des données. Par ailleurs, la sémantique contribue à faciliter l'interopérabilité entre les différents systèmes (Ma et al., 2019). Ces questions caractérisent principalement les villes intelligentes. Dans le cadre de nos recherches, les travaux présentés ci-après s'intéressent aux sujets traitant des technologies intelligentes utilisant des ontologies, des Big Data et des MAS pour l'économie d'énergie dans les maisons intelligentes et les villes intelligentes en général, qui visent à assurer la durabilité et à améliorer la vie des citoyens.

Parmi les travaux axés sur les économies d'énergie, les suivants ont utilisé les MAS comme paradigme principal grâce à leurs caractéristiques telles que la communication, l'autonomie, la tolérance aux pannes et la grande flexibilité. (Anvari et al., 2017) ont proposé un cadre pratique de système de gestion de l'énergie (EMS) basé sur les SMA pour le suivi et le contrôle optimal des bâtiments et des systèmes de micro-réseau. Dans le même

thème, les travaux présentés dans (Kofinas et al., 2018) et (Harmouch et al., 2019) ont discuté d'un fonctionnement en temps réel du système de gestion de l'énergie, qui est basé sur le MAS et l'algorithme T-Cell à convergence rapide. Ces travaux visent à réduire le coût de fonctionnement du réseau et à maximiser la réponse en temps réel dans le réseau. (Soetedjo et al., 2019) ont présenté une plateforme d'essai matériel pour tester le système de gestion de l'énergie des bâtiments (BEMS) à l'aide de MAS. En effet, ils utilisent à la fois un algorithme génétique pour trouver la puissance optimale requise et le contrôleur à logique floue pour surveiller les appareils du bâtiment. Par ailleurs, (Lejdel and Kazar, 2018) ont proposé un MAS pour répartir les différentes tâches entre les agents lorsque chacun d'eux peut exécuter des algorithmes génétiques pour optimiser la consommation d'énergie en temps réel. Ensuite, ils développent un système SIG, qui permet de détecter la position des bâtiments et toutes les données s'y rapportant.

Dans le même but, mais avec des moyens différents, les auteurs (Kott and Kott, 2019) choisissent une approche ontologique de la consommation d'énergie. (Delgoshaei et al., 2018) ont décrit une approche pour surveiller les consommations d'énergie dans les maisons intelligentes en combinant des techniques d'apprentissage par machine avec la modélisation et le raisonnement sémantique. Un algorithme d'apprentissage supervisé avec regroupement de K-means est intégré pour identifier et prévoir la consommation d'électricité dans les maisons. Des travaux plus pertinents sont présentés dans (Degha et al., 2019) pour gérer et améliorer l'efficacité énergétique dans les maisons en tenant compte du comportement des résidents et de l'environnement du résidence. Leurs travaux sont basés sur la prise en compte du contexte dans le but de réduire la consommation d'énergie et de assurer le confort des habitants. L'ontologie intégrée offre un modèle générique permettant l'inférence logique et un algorithme de classification par exploration de données, qui permet d'obtenir les règles représentant les consommations d'énergie normales (Lork et al., 2019). Par ailleurs, (Saba et al., 2019) basant dans leurs travaux sur les mêmes paramètres, ainsi que sur le comportement et les activités des occupants a utilisé la LIB et la TRS pour la présentation des connaissances et le raisonnement intelligent pour réduire la consommation d'énergie dans les villes intelligentes.

En outre, la technologie Big Data est utilisée pour offrir une gestion et un stockage des données fiables, évolutifs et distribués. (Bokolo et al., 2019) ont développé une architecture en couches fournissant des données énergétiques qui vise à faciliter la prosommation d'énergie avec une source renouvelable. Les API ont été utilisées comme adaptateurs de

données pour les prosummateurs, les parties prenantes et le streaming en temps réel pour gérer l'énergie dans les villes intelligentes. De plus, dans (Grolinger et al., 2016) les auteurs ont utilisé l'apprentissage local avec les Big Data et le SVR de régression vectorielle de soutien pour augmenter la vitesse de formation et réaliser des scénarios de prévision énergétique.

En plus des technologies précédentes, l'IoT et d'autres techniques sont utilisées pour réduire la consommation d'énergie dans les villes intelligentes (Wala et al., 2020). (Mahapatra *et al.*, 2017) ont élaboré un cadre unifié pour l'IoT dans les maisons intelligentes afin de construire une ville intelligente verte et durable. Un algorithme de Q-learning basé sur les réseaux de neurones est appliqué à la fois pour réduire la demande et pour conserver l'énergie pendant les périodes de pointe tout en minimisant les inconvénients pour les résidents. (Papastamatiou *et al.*, 2017) ont présenté une méthodologie combinant l'efficacité énergétique et la gestion de l'énergie en utilisant des sources de données multidisciplinaires. L'approche proposée repose sur deux piliers : l'évaluation et l'optimisation de la consommation d'énergie dans les villes intelligentes pour des périodes de courte et de longue durée. Une autre perspective basée sur le cloud pour la gestion de l'énergie dans les villes intelligentes employant un algorithme méta-heuristique a été testée par (Butt *et al.*, 2019), une VM mise en œuvre pour exécuter rapidement les applications des utilisateurs. Cependant, les différents composants du système proposé participent à fournir les informations pertinentes utilisées pour prendre des décisions plus intelligentes et pour aider les consommateurs à réduire la consommation d'énergie dans toutes les villes intelligentes.

Dans les travaux cités ci-dessus, nous avons aperçu certaines lacunes :

- ✓ Certaines approches visent à gérer une seule maison et à fournir une solution individuelle.
- ✓ En cas de panne du système, les maisons perdent le contrôle de la consommation énergétique globale.
- ✓ L'application de certaines modifications du système dans les villes intelligentes entraîne un coût global élevé.
- ✓ Certains cadres ne permettent pas la collecte de données pour l'ensemble de la population.
- ✓ Dans plusieurs cas, nous ne pouvons pas prévoir la consommation globale, si nous ne pouvons pas connaître le comportement général.

- ✓ Absence de politique de consommation à long terme pour éviter d'augmenter la production d'énergie et la pollution de l'environnement.
- ✓ La majorité des approches ne garantissent pas l'extensibilité et la coopération entre les différentes composantes du système.

Pour surmonter ces inconvénients, nous proposons une nouvelle architecture qui vise à accomplir les systèmes précédents en utilisant une ontologie de construction qui offre un cadre structurel (Ma et al., 2019). Nous utilisons l'outil Big Data pour gérer un grand nombre de résidences avec de multiples systèmes et dispositifs, ainsi que pour gérer les interactions en temps réel pour le stockage d'une quantité considérable de données provenant de toutes les ressources des maisons. En outre, il peut assurer l'évolutivité, la disponibilité, la durabilité et réduire considérablement le coût du matériel installé dans chaque maison. Notre système utilise le système multi-agent (SMA) qui permet un haut niveau d'autonomie et de coopération, il réduit considérablement le flux de données à travers le réseau (Howell *et al.*, 2017).

Grâce à l'analyse des données offerte par les outils Big Data et les règles d'inférence, le système peut comprendre le comportement collectif des habitants, connaît les appareils consommateurs et les périodes de pointe. Il peut prédire la consommation à long terme et permet de prendre des décisions grâce à une véritable vue globale afin d'établir une politique générale pour réduire la consommation d'énergie dans les villes intelligentes (Bokolo et al., 2019).

IV.4 Proposition d'architecture multicouche

La section suivante décrit l'architecture globale du système et explique les différentes couches qui le composent, chaque couche contenant un ensemble de technologies qui répondent aux exigences du système.

IV.4.1 Objectif du système

L'objectif principal de l'architecture proposée est de permettre un contrôle efficace des appareils et dispositifs des bâtiments intelligents (climatisation, chauffage, éclairage, téléviseurs, portes, etc.) afin d'offrir une stratégie d'économie d'énergie appropriée, d'assurer le confort des habitants et de réduire la consommation énergétique globale dans les villes intelligentes.

IV.4.2 Architecture générale du système

Dans cette section, nous présentons une description des différentes couches constituant notre approche et les relations entre elles. Le Système Intelligent pour la Gestion de l'Energie dans les Villes Intelligentes (ISEM-SC) repose sur une architecture multicouche en utilisant des agents. Chaque couche contient des agents avec un comportement particulier et des rôles spécifiques. La connaissance est distribuée entre les agents cognitifs et nous choisissons un mode de communication point à point. De plus, ISEM-SC utilise une ontologie de construction intelligente (Degha *et al.*, 2019). D'autres couches sont ajoutées pour répondre aux exigences des villes intelligentes, telles que le nombre élevé de maisons desservis (vitesse de traitement et l'espace de stockage), l'interaction en temps réel, les services d'autonomie, etc. Elle assure également l'extensibilité et l'évolutivité en utilisant l'infrastructure Big Data, l'autonomie et l'interopérabilité avec le paradigme MAS, la flexibilité et la prédiction en intégrant l'ontologie et en employant des règles d'inférence (Lork *et al.*, 2019).

L'architecture ISEM-SC est divisée en quatre couches essentielles ; chaque couche contient plusieurs composants ou modules qui jouent un rôle complémentaire les uns avec les autres et qui échangent des données avec les modules adjacents pour atteindre les objectifs de l'ISEM-SC. (Le tableau 4 montre les couches et les technologies associées).

La Couche	Outils et Technologies
La couche utilisateur	Agents, capteurs, appareils.
La couche service (Backend agent)	Agents, kafka.
La couche stockage	MongDB, Jena, ontology, SWRL.
La couche Hadoop	Hadoop HDFS, MapReduce.

Tableau 4 Les couches de l'architecture et les technologies associées utilisées.

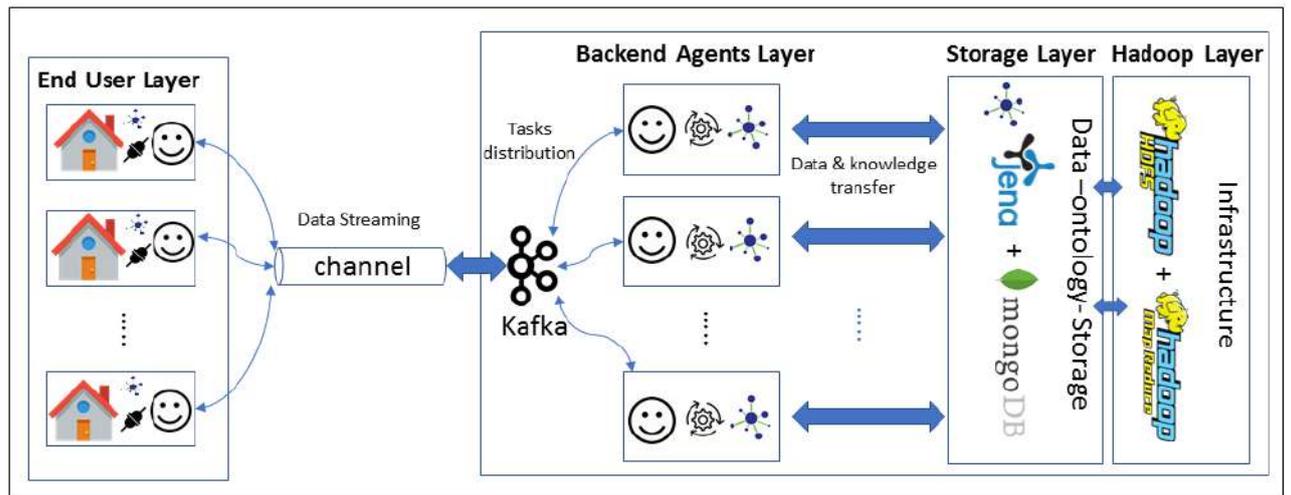


Figure 16 Une architecture multicouche pour ISEM-SC.

Les outils Big Data sont déployés pour traiter des milliers de gigaoctets de données qui sont générés par divers appareils installés dans les villes intelligentes (Raghavan et al., 2020). Ces outils offrent des solutions prometteuses pour gérer le volume, la vitesse et la variété des données énergétiques. De même, une approche en couches offre la robustesse, la capacité et l'évolutivité nécessaires au traitement et à l'analyse des données. De plus, elle permet de gérer à la fois des données énergétiques en temps réel et des données archivées. Ces outils thématiques deviennent une solution de talent pour soutenir une gestion distribuée de la consommation d'énergie dans les villes intelligentes et assurer la durabilité par la politique de bonne gouvernance énergétique (Bokolo *et al.*, 2019).

Dans ce qui suit, nous fournissons une description de chaque couche et une brève explication est également incluse sur la raison de chaque technologie sélectionnée et son rôle (comme le montre la figure 16).

La couche de l'utilisateur Collecte les données générées par différents capteurs et rassemblées chez un agent supervisé de chaque maison qui les envoie au Kafka pour qu'elles soient ensuite redirigées pour être traitées (Alvarez et al., 2019). En outre, il reçoit des commandes qui sont renvoyées par la couche fournisseur de services via Kafka pour activer et changer l'état des dispositifs afin de modifier les conditions environnementales et de réduire la consommation d'énergie.

La couche fournisseur de services Une fois que Kafka reçoit des données, il redirige chacune d'entre elles vers un agent choisi pour traitement par l'agent de service, qui importe les connaissances de la base de données (Mongo DB) et intègre l'ontologie du

bâtiment (Jena) par la communication avec la couche stockage. A ce stade, l'agent de service agent invoque les règles SWRL et applique des traitements sur les connaissances pour mettre à jour la base de données et envoyer des commandes et des conseils à la couche utilisateur final. À cette couche, Kafka a servi de plate-forme de distribution de flux d'événements pour la construction de pipelines de données en temps réel et d'une application de flux. Elle fonctionne comme un cluster sur plusieurs serveurs qui peuvent s'étendre sur plusieurs centres de données, elle peut gérer des trillions d'événements par jour (Le Noac'H *et al.*, 2017). Leur utilisation dans notre architecture consiste à créer des pipelines de données en temps réel qui transfèrent de manière fiable les données entre les couches (couche de l'utilisateur final et du fournisseur de services) et les convertissent en flux de données.

La couche de stockage Basé sur MongoDB et Jena qui sont utilisés pour sauvegarder des informations et des connaissances provenant de ressources environnementales telles que les appareils des maisons intelligentes, les données météorologiques, etc. (Chodorow *et al.*, 2019). Elle contient tout l'historique des données et fournit un cadre pour la construction de connaissances à partir de la transformation des données combinée à l'ontologie onto-SB. Cette couche comprend un moteur de règles d'inférence qui est utilisé pour offrir une gestion efficace des données énergétiques des villes intelligentes (Yang *et al.*, 2018).

La couche d'infrastructure Hadoop Dans l'architecture de la ville intelligente, d'énormes quantités de données sont collectées à partir de millions de capteurs déployés dans chaque maison (Jawhar *et al.*, 2018). Dans notre modèle, la couche Hadoop communique directement avec la couche de stockage via MongoDB et Jena pour effectuer un traitement rapide des données et permet une gestion efficace du stockage, de la disponibilité, des performances et de l'évolutivité, etc. (par HDFS et MapReduce).

IV.4.3 Diagramme de séquence AUML de l'architecture proposée

Ce diagramme résume la séquence des événements et les interactions entre les différents composants utilisés dans le système proposé (voir figure 17).

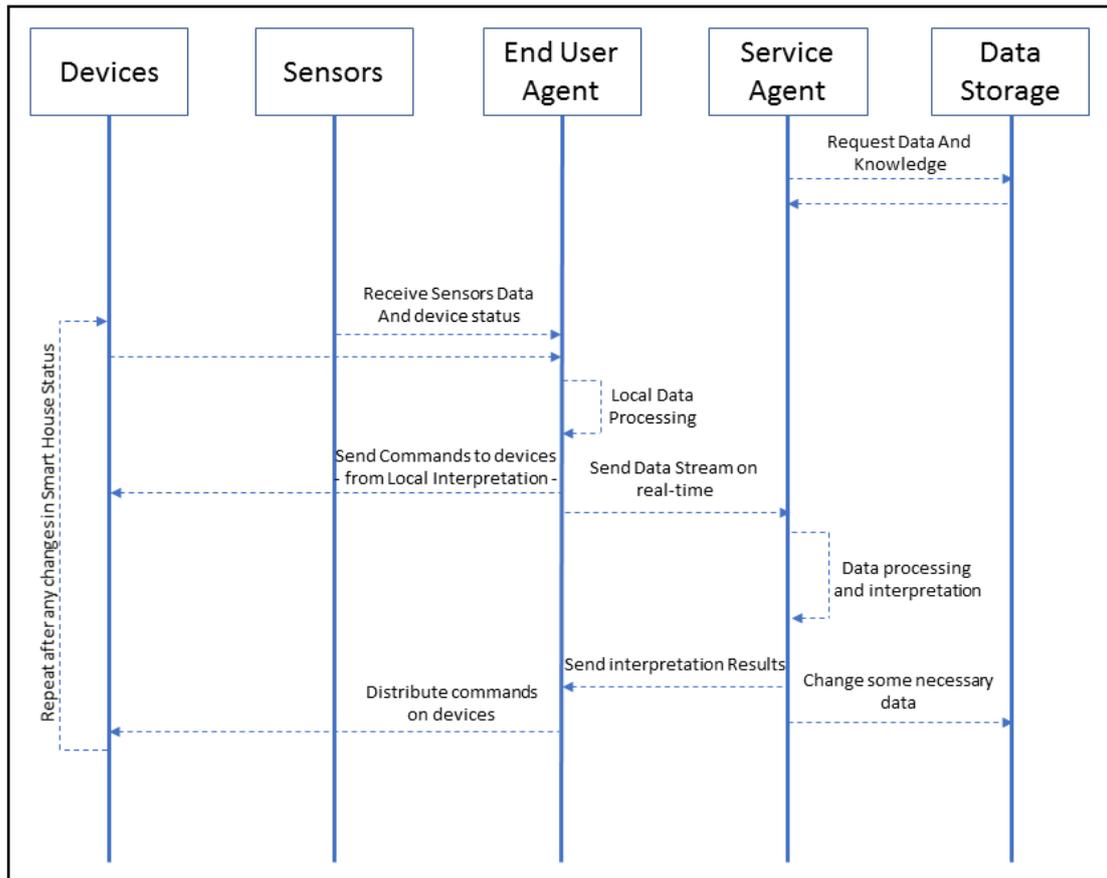


Figure 17 Diagramme de séquence AAML de l'architecture ISEM-SC.

IV.4.4 Fonctionnement du système

La maison représente un environnement dynamique où les paramètres et les états des appareils changent fréquemment. Une fois que les capteurs détectent une modification, ils envoient des données à l'agent utilisateur (maison) qui reçoit et collecte toutes les données provenant des appareils, des capteurs et de l'environnement (Estrada et al., 2019). Chaque agent invoque l'ontologie et exécute les règles de gestion des appareils locaux et envoie les données (paramètres d'environnement, circonstances du domicile, etc.) à l'agent de service pour traitement. Le grand nombre de tâches envoyées par les différents agents domestiques est redirigé via *Kafka* vers les agents de service finaux. Ceux-ci enregistrent les nouvelles informations de chaque base de données de maison intelligente et mettent à jour l'ontologie de la maison intelligente avec l'état en temps réel du bâtiment intelligent. Les agents de service invoquent le Reasoning Engine Module (*REM*) et génèrent des décisions et de nouvelles connaissances basées sur le contexte actuel. À cette étape, l'agent vérifie la base de données et l'ontologie stockées au niveau de la couche de stockage (*MongoDB* et *Jena*) afin de fournir des décisions adéquates pour le contexte actuel. L'outil Hadoop déployé

pour gérer le traitement parallèle et réaliser le stockage via (*HDFS* et *MapReduce*). Ensuite, les agents de service envoient des commandes et des décisions aux agents des utilisateurs finaux (maison) qui demandent aux infrastructures d'actionneurs et d'appareils de changer leur état pour réduire la consommation d'énergie.

En outre, les agents de service assurent un contrôle général sur l'ensemble de la ville intelligente en vérifiant la demande totale d'énergie et en le comparant avec l'énergie disponible produite par le fournisseur d'énergie. Deux cas se présentent : si l'énergie disponible est suffisante pour répondre à la demande, il n'y a pas d'intervention. Dans le cas contraire, l'agent de service envoie des instructions à tous les agents utilisateurs pour réduire l'énergie en modifiant le seuil de fonctionnement des appareils actifs. Dans ce cas, le système contribue grandement à la gestion des heures de pointe, assure une politique équitable pour tous les consommateurs et évite le crash énergétique. Ainsi, l'architecture proposée permet d'atteindre un objectif important en réduisant la consommation d'énergie, en diminuant la facture des consommateurs, en assurant la disponibilité de l'énergie et en économisant la production d'énergie sans affecter le confort des habitants (Sayah *et al.*, 2020).

IV.4.5 Ontologie de maisons intelligentes Onto-SB

La figure 18 montre notre modèle de connaissance ontologique appelé Onto-SB (Degha *et al.*, 2019) qui est destiné à servir l'architecture ISEM-SC. Il est proposé pour offrir un cadre structurel de données sur les maisons intelligentes. Il contient des définitions des concepts élémentaires utilisés et de leurs relations. L'Onto-SB est utilisé pour représenter les concepts et les relations employés dans l'ontologie. Onto-SB propose un cadre structurel pour l'organisation des données de la maison intelligente. Il contient une définition compréhensible de la plupart des concepts élémentaires utilisés dans les maisons intelligentes et de leurs relations. L'Onto-SB comprend plus de 200 concepts, représentant les résidents, les appareils, les lieux, l'environnement, les services, etc. En outre, SWRL est déployé pour effectuer des règles à différents fins lorsque les relations structurées entre les concepts et la représentation formelle des connaissances du domaine permettent de raisonner (Degha *et al.*, 2019).

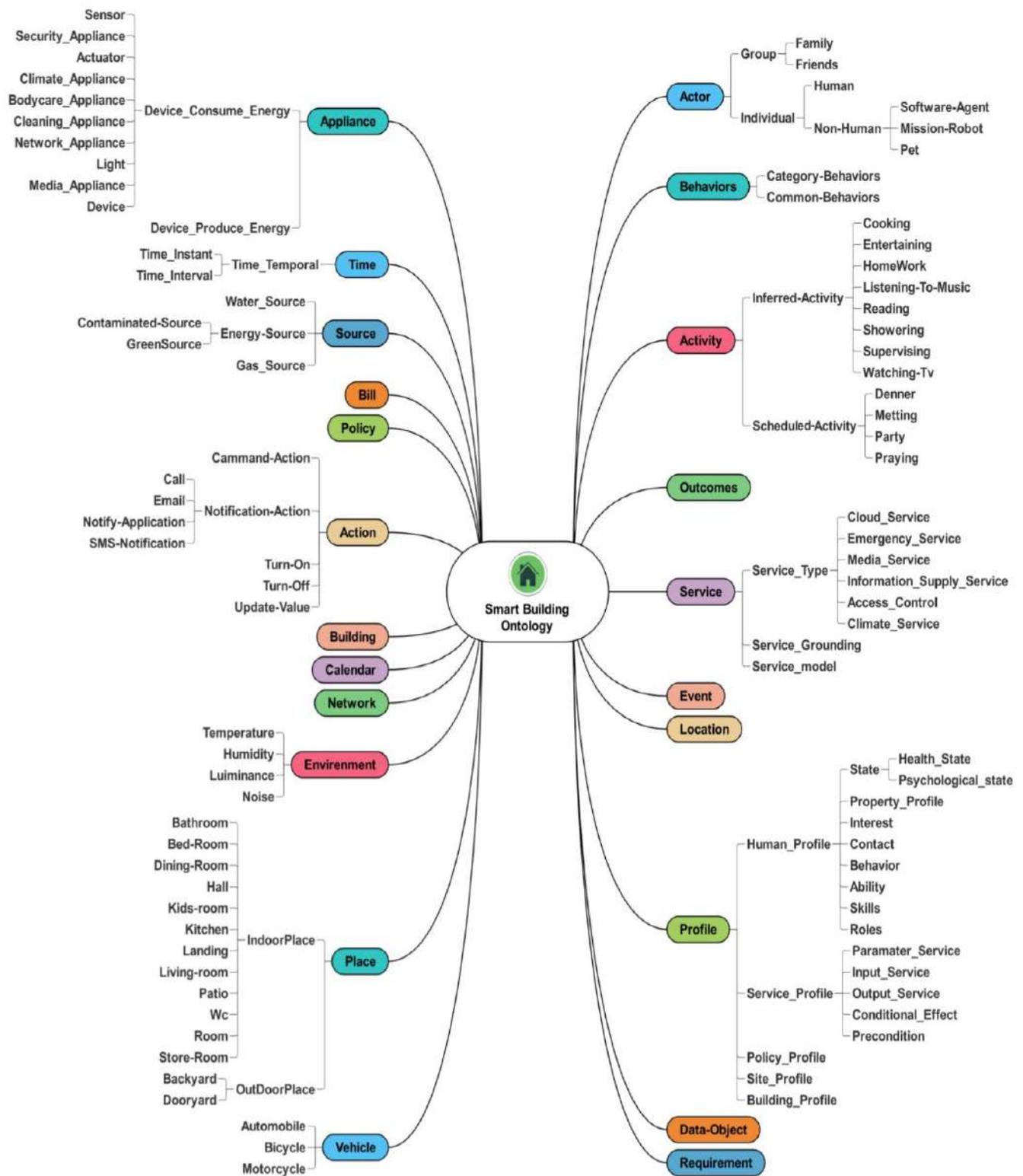


Figure 18 La taxonomie utilisée dans les concepts de l'Onto-SB (Degha et al., 2019).

Dans les sections suivantes, nous présentons brièvement certains concepts de l'Onto-SB avec leur description et quelques exemples. Pour chaque attribut, nous illustrons la relation entre les concepts (tableau 5).

<i>Concept</i>	<i>Description</i>	<i>Instances</i>	<i>Attribut</i>	<i>Relations</i>
Acteur	Représente les résidents de la maison	Mère, père, sœur, enfant, grand-père, etc..	OccupantName OccupantRole	OccupantHasActivity OccupantHasplace
Maison	Représente la maison	ID maison, emplacement, taille, profil, etc.	BuildingName BuildingSize	BuildingHasActor BuildingHasPlace
Appareil	Représente les appareils de la maison	Lumière, Lap top, TV, fer à repasser, climatiseur, etc.	ApplianceName AppliancePower	ApplianceHasAction ApplianceHasPlace
Place	Représente les différents lieux de la maison	Chambre, cuisine, WC, garage, salon, etc.	RoomSize RoomName	PlaceHasAppliance PlaceHasEvent
Activité	Représente les activités des habitants	Dormir, regarder la TV, cuisiner, manger, se réveiller, entrer, etc.	BehaviourName ActivityTime	ActivityHasPlace ActivityHasTime

Tableau 5 Concepts, instances, attributs et relations de la maison intelligente.

Cette section est consacrée à la présentation de l'environnement des bâtiments concernés par cette étude et met l'accent sur les éléments qui consomment de l'énergie (appareils et dispositifs). De même, nous mettons en évidence les paramètres qui influencent la consommation d'énergie (données climatiques, activités des habitants, etc.).

IV.4.6 Module moteur de raisonnement (REM)

Ce module intégré à la couche service est déployé pour générer des décisions en utilisant les connaissances disponibles sur les maisons intelligentes. Il a un rôle essentiel dans la mise en œuvre et l'efficacité du système proposé. REM basé sur des règles SWRL et représenté comme une logique conditionnelle, où chaque règle utilise une conjonction de clauses prédicatrices à une liste d'actions exécutables. Grâce à ces règles, le REM peut détecter le gaspillage d'énergie et fournir des mesures pour économiser l'énergie. Les décisions déduites sont utilisées pour mettre à jour la base de connaissances et envoyer des commandes aux agents des utilisateurs finaux qui agissent sur les actionneurs installés dans le bâtiment intelligent pour réduire la consommation d'énergie. La figure suivante montre quelques règles qui sont utilisées pour gérer et surveiller le gaspillage d'énergie dans le bâtiment intelligent (figure 19).

Rule 1 switch off the lights in a place (room, garage, kitchen, etc.) in case of the external lighting (sun) is suitable (in our case, greater than 22 Linux):
Lighting-Sensor (?j) ^ Appliance-Value (?j, ?val) ^ swrlb : greaterThan (? val, ?22) ^ Appliance-LocatedInPlace (?z, ?j) ^ Lighting-appliance (h) ^ Appliance-LocatedInPlace (?z, ?h) --> Appliance_State (?h, "off") ^Turn-off(?h)

Rule 2 stops lighting appliance fonction when it detects the absence of inhabitant in a given place:
Moving-Sensor (?x) ^ Appliance-State (?x, ?stat) ^ swrlb : equal (? Stat, ?on) ^ Appliance-Value (?x, ?val) ^ Appliance-Value (?x, ?0) ^ Appliance-LocatedInPlace (?x, ?z) ^ Appliance-LocatedInPlace (?l, ?val) ^ Place (?z) ^ Light (?l) ^ Appliance-State (?l, ?stat2) ^ swrlb : equal (? Stat2, "on") --> Appliance_State (?l, "off") ^Turn-off(?l)

Rule 3 stops cooling appliance fonction when it detects the absence of inhabitant in the room:
Moving-Sensor (?x) ^ Appliance-State (?x, ?stat) ^ swrlb : equal (? Stat, ?on) ^ Appliance-Value (?x, ?val) ^ Appliance-Value (?x, ?0) ^ Appliance-LocatedInPlace (?x, ?z) ^ Appliance-LocatedInPlace (?l, ?val) ^ Place (?z) ^ aircondition (?l) ^ Appliance-State (?l, ?stat2) ^ swrlb : equal (? Stat2, "on") --> Appliance_State (?l, "off") ^Turn-off(?l)

Rule 4 used to stop cooling appliance (air condition) if the ambient temperature in the room reach 26 C :
Temperature-Sensor (?t) ^ Appliance-Value (?t, ?val) ^ swrlb : lessThan (? val, ?26) ^ Appliance-LocatedInPlace (?z, ?t) ^ Cooling-appliance (h) ^ Appliance-LocatedInPlace (?z, ?h) --> Appliance_State (?h, "off") ^Turn-off(?h)

Rule 5 used to turn off the water pump if the user turns it on and leaves their place (in the bathroom, WC or kitchen):
Moving-Sensor (?x) ^ Appliance-State (?y, ?stat) ^ swrlb : equal (? Stat, ?on) ^ Appliance-Value (?y, ?val) ^ Appliance-Value (?y, ?0) ^ Appliance-LocatedInPlace (?y, ?v) ^ Appliance-LocatedInPlace (?l, ?val) ^ Place (?v) ^ pump (?l) ^ Appliance-State (?l, ?stat2) ^ swrlb : equal (? Stat2, "on") --> Appliance_State (?l, "off") ^Turn-off(?l)

Rule 6 turn off the air exchanger in the kitchen when the relative humidity (produced in cooking) less than 30 per cent, so the mother finish cooking:
Humidity-Sensor (?h) ^ Appliance-Value (?h, ?val) ^ swrlb : lessThan (? val, ?30) ^ Appliance-LocatedInPlace (?z, ?h) ^ Airexchanger-appliance (h) ^ Appliance-LocatedInPlace (?z, ?h) --> Appliance_State (?h, "off") ^Turn-off(?h)

The **Rule 7** shut off every selected useless devices when bedtime comes (in our case 10:30) except the exempted appliances such as refrigerator, camera, clock, etc.:

Time-Instant (?t) ^ swrlb : equal (?t, "10:30 PM") ^ applianceSleepMode (?x, ?m) ^ swrlb : equal (?m, "on") --> Appliance_State (?x, "off") ^Turn-off(?x)

Figure 19 Règles SWRL utilisées pour gérer l'énergie des appareils.

IV.5 Présentation de l'environnement

Notre étude a été réalisée dans la ville d'El Oued en Algérie. El Oued est située dans le sud-est de l'Algérie, à environ 600 km de la capitale Alger. Elle se caractérise par sa latitude : 33°, 22', 00" N et une longitude : 06°, 51', 00" E avec une altitude de 85 m au-dessus du niveau de la mer. Une famille de neuf membres occupe la maison (grand-père, grand-mère, mère, père et cinq enfants), la maison conçue dans l'architecture locale (dans notre modèle, maison avec 3 pièces, cuisine, salle de bain et garage).

El Oued se caractérise par un climat désertique hyper-aride et chaud, avec un été très long et brûlant et un hiver court et froid. La moyenne des températures maximales en été est de 46-48 C° en juillet, elle est considérée comme l'une des villes les plus chaudes d'Algérie ("Infoclimat", 2020). Ces caractéristiques engendrent une forte demande en énergie qui est utilisée pour la climatisation. Plus de 120 jours par an, le mercure dépasse les 40 C°. En hiver, la température devient modéré le jour mais froide la nuit, ce qui nécessite des sources de chauffage. La plupart des jours de l'année, le ciel est clair et le soleil est omniprésent, la durée moyenne d'insolation est d'environ 3 978 h par an, ce qui offre un environnement parfait pour la source d'énergie solaire. L'humidité relative est généralement faible toute l'année, avec une moyenne annuelle d'environ 26 %.

La maison expérimentale comprend une variété d'appareils électriques. Pour avoir une idée précise de l'énergie consommée dans la maison, nous présentons ci-après un exemple d'appareils électriques et de leurs puissances (voir tableau 6).

<i>Place</i>	<i>Appareils électriques</i>	<i>Nombre D'Appareils</i>	<i>Consommation W/h</i>
Cuisine	Lumière	2	25
	Réfrigérateur	1	200
	Lave-vaisselle	1	1300
	Four micro-onde	1	1150
	Mixeur électrique	1	300
	Machine à café	1	700
	aspirateur	1	800
	Ventilateur de plafond	1	70
	Échangeur d'air	1	370
Salon	Lumière	3	25
	Climatiseur (8000 BTU)	1	900
	TV LCD	1	150
	Démodulateur	1	25
Chambre 1	Lumière	2	25
	Climatiseur (6000 BTU)	1	700
	Le fer a repassé	1	850
	PC Portable	1	70
Chambre 2	Lumière	2	25
	Climatiseur (6000 BTU)	1	700
	L'horloge	1	5
Chambre à coucher	Lumière	3	25
	Climatiseur (6000 BTU)	1	700
	L'horloge	1	5
	Ordinateur de bureau	1	80
Salle de bain et WC	Lumière	4	25
	Machine à laver	1	350
	Rasoir électrique	1	15
	Pompe à eau	1	1200
Garage	Lumière	4	25
	Camera	1	25

Tableau 6 Illustration des appareils utilisés dans le modèle de maison intelligente.

Le tableau suivant illustre les différents capteurs utilisés dans les bâtiments intelligents et explique le rôle de chacun d'entre eux (Kertiou et al., 2018). Le nombre de chaque type de

capteur est également indiqué. Tous les capteurs sont configurés en fonction du rôle destiné aux différents capteurs. Chaque capteur génère périodiquement (1 sec, 5 sec, 1 min, etc.) des données et les envoie à l'agent local. Les mesures et les détections produites expriment les différents événements survenus par les habitants ou les changements environnementaux qui affectent directement la consommation d'énergie dans la maison intelligente. Le système proposé repose essentiellement sur ces données pour analyser et appliquer les décisions visant à mettre à jour la base de connaissances et à économiser l'énergie sur l'ensemble de la ville intelligente.

<i>Capteur</i>	<i>Rôle de capteur</i>	<i>Nombre</i>
Capteur de température	Un appareil utilisé pour mesurer la température (int / ext)	12
Capteur d'humidité	Un appareil mesure l'humidité relative dans une atmosphère	10
Capteur optique	Utilisé pour mesurer la quantité physique de rayons lumineux	16
Capteur de pression	Un appareil qui détecte la pression	6
Capteur de proximité	Un appareil qui détecte la présence ou l'absence d'un objet	14
Capteur d'ouverture / fermeture	Utilisé pour détecter les objets fermés / ouverts (fenêtre, porte, etc.)	32
Capteur de gaz	Un appareil utilisé pour détecter la présence de divers gaz	10
Détecteur de fumée	Un appareil qui détecte la fumée (particules et gaz)	10

Tableau 7 Les capteurs utilisés et leurs rôles dans la maison intelligente.

IV.5.1 Description des scénarios

Pour tester le système, nous avons émis l'hypothèse suivante : nous supposons un jour d'été (26 août 2019) dans la ville d'El Oued pendant une période de 24 heures (voir figure 20). Le scénario est caractérisé par certaines propriétés: durée, événement, acteur, lieu et appareil. La séquence des événements affecte les états des appareils (allumage/extinction), et donc la consommation d'énergie dans tout la maison. Le tableau 8 montre les étapes du scénario.

<i>Temps</i>	<i>Evènement</i>	<i>Acteur</i>	<i>Lieu</i>	<i>Appareil activé</i>
00:00 Am-06:00 Am	Temps de dormir	Tout le monde	Chambre, salle 1 et 2	Climatiseur
06:00 Am-06:20 Am	Se lever	Mère	Chambre à coucher	Lumières, pompe à eau
06:20 Am-06:45 Am	Préparer le petit déjeuner	Mère	Cuisine	Lumières, cafetière
06:45 Am-07:00 Am	Se lever	Tous les autres	Chambre, salle 1 et 2	Lumières, pompe à eau
07:00 Am-07:20 Am	Prendre le petit déjeuner	Tout	Cuisine	-
07:20 Am-07:50 Am	Se préparer à sortir	Père, enfants	S B, chambre	Lumières, pompe à eau
07:20 Am-08:50 Am	Nettoyage de Maison	Mère	Toute la maison	Vide, pompe à eau
08:50 Am-10:00 Am	Regarder la télévision	Tout le monde	Salon	Air conditionné, TV
10:00 Am-11:20 Am	Préparer le déjeuner	Mère	Cuisine	Micro-ondes, mixeur,
10:00 Am-11:30 Am	Lire des livres	Grand-père	Salon	-
11:30 Am-01:00 Pm	Déjeuner	Tout le monde	Cuisine	Lumières, micro-ondes
01:00 Pm-02:30 Pm	Regarder la télévision	Tout le monde	Salon	Climatiseur, TV, lumières
02:30 Pm-03:00 Pm	Retourner à la maison	Père, enfants	Toute la maison	-
03:00 Pm-03:30 Pm	Prends une douche	Père, enfants	Salle de bain	Pompe à eau, lumières
03:30 Pm-06:30 Pm	Repos	Tout le monde	Salon	Climatiseur, TV
06:30 Pm-08:30 Pm	Préparer / manger le dîner	Mère / Tous	Cuisine	Micro-ondes, mixeur,
08:30 Pm-10:30 Pm	Regarder TV / réviser	Tout le monde	Salon, salle 1 et 2	Climatiseur, TV, lumières
10:30 Pm-06:00 Am	Aller au coucher	Tout le monde	Chambre, salle 1 et 2	Climatiseur

Tableau 8 Un scénario simple de routines quotidiennes dans la maison intelligente.

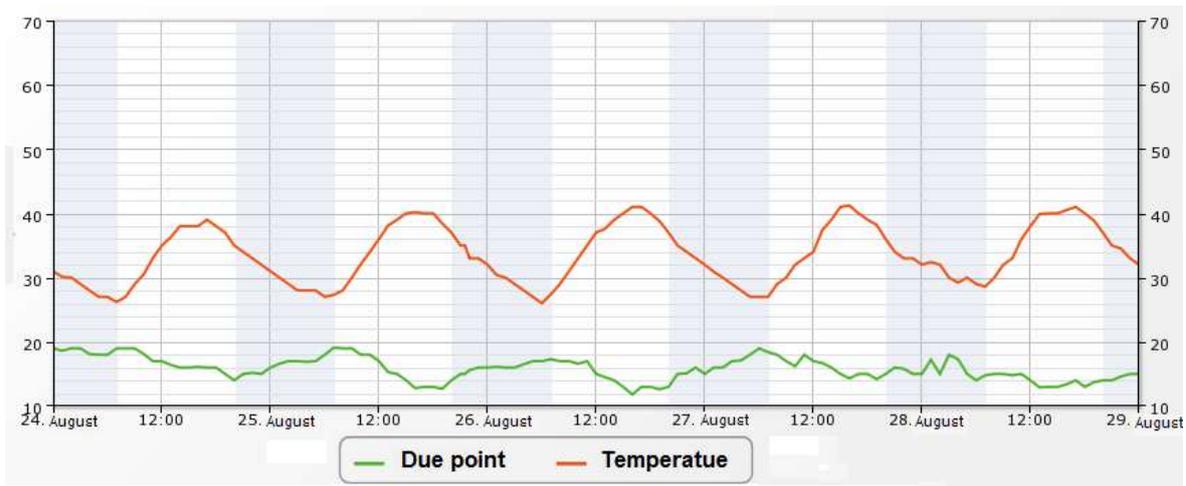


Figure 20 La température réelle du 26 août 2019 (“Infoclimat”, 2020).

Après avoir créé un scénario, nous devrions connaître la consommation d'énergie dans chaque maison intelligente. Pour cette raison, nous calculons le taux de consommation de chaque appareil sur la journée d'expérimentation. Le tableau 9 présente les valeurs obtenues.

Dispositif en marche	Puissance (W/h)	Nombre d'appareil	Temps de fonctionnement (m)	Énergie consommée (W)
Eclairage	25	20	360	3000
Climatiseur 9000 BTU	900	1	570	8550
Climatiseur 6000 BTU	700	3	450	15750
Réfrigérateur	200	1	1440	4800
Lave-vaisselle	1300	1	90	1950
Four micro-onde	1150	1	80	1533
Mixeur électrique	300	1	80	399
Machine à café	700	1	50	583
aspirateur	800	1	90	1200
Ventilateur	70	1	210	245
Échangeur d'air	370	1	240	1480
TV LCD	150	1	280	700
Démodulateur	25	1	280	116
Fer à repasser	850	1	30	425
Portable	70	1	120	140
L'horloge	5	2	1440	240
Ordinateur de bureau	80	1	90	120
Pompe à eau	1200	1	155	3100
Rasoir électrique	15	1	30	7.5
Caméra	25	1	1440	600

Tableau 9 Consommation d'énergie dans une maison intelligente pendant la période de 24 heures.

Pour obtenir l'énergie totale consommée par chaque appareil, nous utilisons la formule suivante (1) :

$$E_{di}(wh) = \sum_{i=1}^n (end\ time\ d_i - start\ time\ d_i) * P \quad (1)$$

Pour calculer l'énergie totale consommée dans chaque maison intelligente, nous appliquons la formule suivante (2) :

$$E_t(wh) = \sum_{i=1}^n E_{di}(wh) \quad (2)$$

Dont " E " est l'énergie (Wh), " di " est le type d'appareil, " P " la consommation de l'appareil.

Un aperçu général dans le tableau précédent montre que la consommation électrique élevée est générée par la climatisation (plus de 55 %). Comme la température reste élevée toute la journée, les résidents préfèrent rester chez eux pour éviter la chaleur élevée à l'extérieur et ont besoin d'une température adéquate, ce qui exprime la consommation énorme dans ce scénario.

IV.6 Mise en œuvre

Nous avons utilisé JAVA pour développer les modules du système et l'éditeur IDE Eclipse. Pour manipuler et traduire l'ontologie en OWL, nous avons utilisé l'API Protégé5. Après avoir créé des classes avec leurs propriétés et leurs relations, nous avons ajouté SWRLTab pour appliquer les règles SWRL et les requêtes SQWRL, le module de raisonnement (c'est une étape pour déduire de nouvelles connaissances contextuelles basées sur les contextes disponibles) qui vise à dériver de nouvelles connaissances en exploitant les données disponibles et en explorant le contexte de l'ontologie tel que (profil humain, présence personnelle, appareil consommant de l'énergie, etc.). Ces connaissances sont utilisées pour réduire la consommation d'énergie et maximiser le confort des résidents. De plus, la figure 21 présente l'interface graphique de gestion des agents à distance JADE (Java Agent Développement Environnement) qui décrit les différents agents créés dans chaque couche, alors que chaque couche tient un agent responsable de la communication avec les autres agents en envoyant un message. Ils peuvent coopérer et tirer parti de leur autonomie, de leur modularité, de leur distribution et de leur intelligence pour réduire le temps de calcul élevé nécessaire à la gestion d'un grand nombre de données.

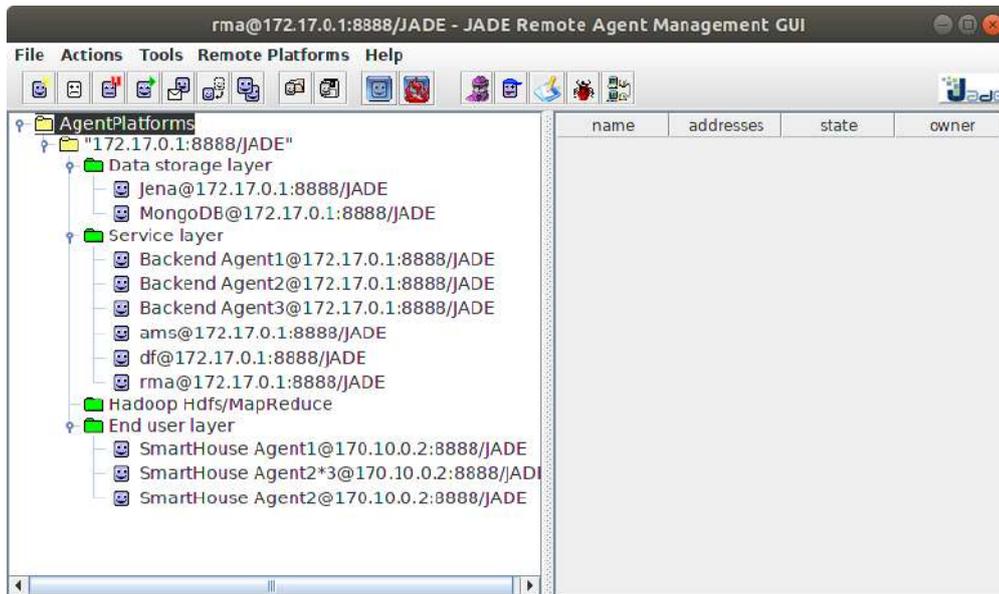


Figure 21 L'implémentation JADE de différentes couches d'agents.

L'environnement utilisé pour la mise en œuvre du système est un cluster de 5 machines (maître 4 Go de RAM, disque dur 250 Go, 4 esclaves 2 Go de RAM et disque dur 100 Go). Apache Hadoop a été installé comme infrastructure dans ce cluster, et d'autres frameworks tels que Kafka, MongoDB et Jena ont été installés aussi. La plate-forme Jade a été également installée dans ce cluster. Dans chaque maison intelligente, il y a une machine à faible performance (1 Go de RAM et 100 Go de disque dur) qui est utilisé comme une machine client pour le serveur Big Data et la plate-forme JADE est installée dans cette machine (voir tableau 10).

<i>Environnement</i>	<i>Machine</i>	<i>Nombre</i>	<i>Description</i>
Environnement 1	Machine à faible performance	Dans chaque maison	CPU 1 Core 1 GB RAM 100 GB HDD
Environnement 2	Master	1	CPU 2 Core 4 GB RAM 250 GB HDD
	Slave	4	CPU 1 Core 2 GB RAM 100 GB HDD

Tableau 10 Illustration des matériels utilisés pour la mise en œuvre.

L'agent local dans chaque maison réalise le processus concernant la collecte de données à partir des appareils et des capteurs dans chaque maison. Ensuite, l'annotation des données est faite en utilisant l'ontologie avant de les envoyer à l'agent de service en mode synchronisé. Ensuite, les agents utilisateurs finaux reçoivent un message de retour contenant la commande qui sera exécutée ultérieurement. S'il n'y a pas de message pour une raison quelconque, l'agent local réagit en fonction de son comportement local. D'autre part, l'agent de service recharge l'ontologie lors de la phase de démarrage, en se préparant à recevoir des messages de toutes les maisons. Il commence ensuite à analyser les données acquises en utilisant la base de connaissances et leurs règles stockées en fonction du contexte actuel. Il invoque une solution pour chaque maison et envoie un message à l'agent utilisateur final qui agit afin de réduire la consommation d'énergie à leur niveau (voir algorithme 1 et 2).

Algorithme 1 d'envoi des données des maisons à l'agent de service :

```

begin
loadontolog(http://127.0.0.1/Onto-SB);{importe ontology from local }
ACLMessage cfp;{ creation sending msg}
ACLMessage receive;{creation recieving msg}
while(isalive){repeat}
{ data[]=sensorGather();{ gathering sensors data}
dataAnt=annotation(data[]){ semanticate data collected}
cfp.setContent(dataAnt);{load send msg contenet}
send(cfp,serviceAgent){sent msg to service agent}
receive = receive();{wait msg for a cyclic period /synchron/}
if(receive != null){if not embty msg}
Actions = getAction(receive.getContent()){get action}
} else
{ Actions=generAction(dataAnt){execute local action }}
for(Action in Actions){browse the action liste}
runActions( Action);{execute action}}
end.

```

Algorithme 2 planification des tasks au niveau d'agent de service:

begin

loadontology(http://127.0.0.1/Onto-SB);{ import ontology from local}

ACLMessage receive;{ creat recieve msg}

ACLMessage reply { creat response msg}

Array houses; { creat houses list}

while(isalive) { if the agent is awake}

{ receive = receive();{ receive msg}

houses = handle(receive); { handele the received msg}

planifiedAction(houses,Resource){ invok the solution for each house and their appliances}

for(house in houses){ repeat for each house in list}

{ reply = house.receive.createReply(); { creat response for the received msg}

reply.setContent(house.getActions());{ load msg for each house contained action}

send(reply);{ send response msg} }

}

end.

Le système proposé fournit non seulement une solution d'économie d'énergie locale pour chaque maison intelligente, mais offre également une surveillance générale de l'ensemble du réseau électrique. Dans ce contexte, l'agent de service utilise ses capacités de raisonnement et effectue un contrôle cyclique du rapport entre la demande totale d'énergie et l'énergie disponible offerte par le fournisseur. Lorsque la consommation dépasse l'offre, l'agent de service envoie un message aux agents des utilisateurs finaux pour qu'ils augmentent le seuil de refroidissement. Après des tests répétitifs, la consommation d'énergie atteint un niveau d'équilibre et empêche la panne du réseau. L'algorithme suivant démontre la stratégie utilisée :

Algorithme 3 ajustement de la consommation totale :

```

TotalPoweredEnergy;
TotalRequiredEnergy;
ACLMMessage reply;
Response response;
    CyclicBehaviour(this)
    { for (house in houses)
        TotalRequiredEnergy = house.EnergyConsumed;
        If ( TotalRequiredEnergy>TotalPoweredEnergy)
            response.Actions.add("DecreaseAirCondition:1");
        else
            response.Actions.add("IncreaseAirCondition:1");
        for (house in houses)
            { reply.addReceiver(house.Agent);
              reply.setContent(response)
              reply.send();}
    }

```

IV.7 Conclusion

L'avancée significative des applications contemporaines (villes intelligentes, réseaux sociaux, soins de santé, IoT, etc.) nécessite une vaste émergence des technologies Big Data et sémantiques dans ces domaines. Dans ce chapitre, nous avons clarifié et expliqué le rôle important de Big Data et de l'ontologie avec le paradigme SMA pour améliorer l'interopérabilité des systèmes. Nous avons décrits l'ontologie des maisons intelligentes (Onto-SB), qui est un outil puissant pour représenter les connaissances du domaine et fournit un cadre structurel pour organiser les données sur les bâtiments intelligents. Ainsi

les règles d'inférence utilisées pour diminuer la consommation d'énergie. L'environnement d'étude a été décrit avec les différents paramètres qui sont en relation. De fait que cette approche est basée sur un modèle multi couches, ce qui met en évidence l'efficacité et la robustesse de système proposé qui peut assurer un haut niveau d'autonomie et d'extensibilité.

Chapitre V

Expérimentation et Résultats

V.1 Introduction

L'objectif de ce chapitre est d'évaluer la performance de l'approche proposée par des différentes métriques. La première métrique est le taux de consommation d'énergie, tandis que la deuxième est le temps d'exécution des tâches. L'expérience effectuée consiste à comparer les performances de notre approche avec celles d'autres approches utilisées dans des travaux précédents. Comme le taux de consommation et le temps de traitement sont des critères pour évaluer l'approche proposée. Nous montrons clairement comment la combinaison des différentes technologies peut améliorer la qualité des résultats obtenus.

V.2 Première simulation

La première expérience porte sur la consommation d'énergie dans la ville intelligente, où nous avons utilisé le scénario précédent. Après avoir appliqué des règles dans plusieurs cas, l'approche proposée permet une économie d'énergie significative dans divers dispositifs tels que les climatiseurs, l'éclairage, la télévision, la pompe à eau, etc. Pour éteindre un tel appareil, nous utilisons un autre critère lorsque la lumière ou la température extérieure s'adapte aux préférences de l'habitant (figure 22).

Pour la consommation totale, le système intervient pour ajuster le seuil de préférence afin de réduire la demande totale. Le tableau 11 montre l'énergie économisée dans différents appareils en utilisant certaines règles qui donnent des résultats significatifs dans la réduction de la consommation. Environ 15,62 % de l'énergie économisée dans cette expérience par rapport à la consommation du système MAS-GA (Lejdel and Kazar, 2018) (sans ontologie et Hadoop).

<i>Appareils</i>	<i>Temps de fonctionnement (m)</i>	<i>Energie consommée (W)</i>	<i>Heure d'arrêt (m)</i>	<i>Energie préservée (W)</i>	<i>Energie préservée moyen %</i>	<i>Règle utilisée</i>
Eclairage	360	3000	100	830	27.66	Règle 1,2
Climatiseur 9000 BTU	570	8550	70	1050	12.28	Règle 3,4
Climatiseur 6000 BTU	450	15750	90	2700	17.14	Règle 3,4
Ventilateur de plafond	210	245	25	29.16	11.90	Règle 3
Aspirateur	240	1480	33	203.5	13.75	Règle 6,7
TV LCD	280	700	40	100	14.28	Règle e 7
Démodulateur	280	116	40	16.66	14.36	Règle 7
Pompe à eau	155	3100	25	500	16.12	Règle 5,7

Tableau 11 Énergie économisée dans divers appareils avec les règles utilisées.

Les résultats obtenus dans le tableau ci-dessus sont calculés pendant le déroulement de scénario précédent (tableau 8) en fonction de consommation de différents dispositifs (tableau 9). A chaque fois, on applique les règles appropriées afin d’optimiser la consommation d’énergie.

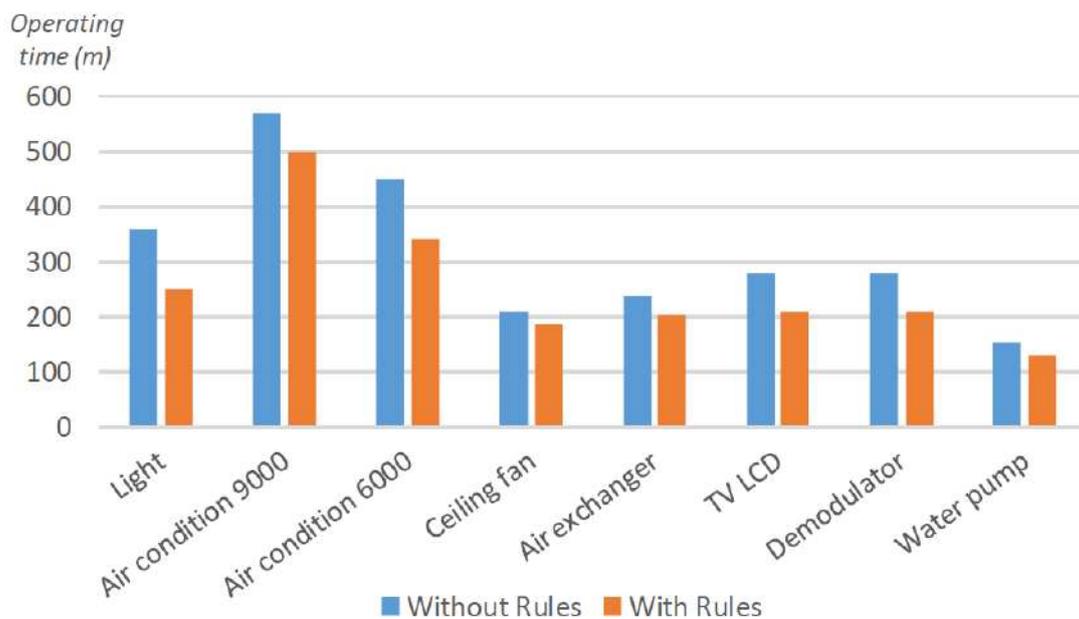


Figure 22 Temps de fonctionnement des appareils avec et sans règles.

La figure 23 montre les résultats obtenus dans plusieurs scénarios (présentés en rouge) ; nous avons utilisé 20 maisons intelligentes, où la consommation d'énergie est réduite dans chaque cas avec une moyenne de 1300 Kw/h.

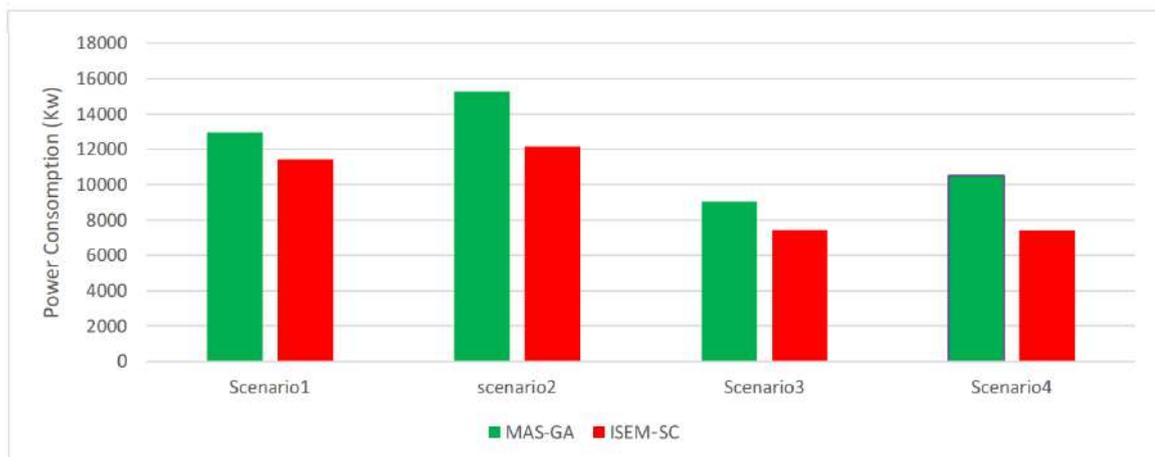


Figure 23 Consommation d'énergie dans la ville intelligente pour différents scénarios.

V.3 Deuxième simulation

La deuxième phase d'expérimentation vise à valider le critère de durée de traitement. C'est pourquoi nous avons appliqué la simulation de notre cadre sur le cluster supposée (5 machines : Master avec 4 Go de RAM, 100 Go de disque dur, et 4 Esclaves avec 2 Go de RAM, et 250 Go de disque dur), la figure 24 présente l'initialisation des paramètres pour l'environnement d'expérimentation. Les scénarios sélectionnés supposent un environnement dans différents cas : une, 5, 10 et 20 maisons intelligentes.

```

1 // Initialize the CloudSim library
2 CloudSim.init(num_user, calendar, trace_flag);
3 // Second step: Create Datacenters
4 //Datacenters are the resource providers in CloudSim. We need at list one of them to run a CloudSim simulation
5 @SuppressWarnings("unused")
6 Datacenter datacenter0 = createDatacenter("Datacenter_0");
7 //Third step: Create Broker
8 DatacenterBroker broker = createBroker();
9 int brokerId = broker.getId();
10
11 //Fourth step: Create one virtual machine
12 vmlist = new ArrayList < Vm > ();
13
14 //VMMaster description
15 int vmidm = 0;
16 int mipism = 1000;
17 long sizem = 100000; //image size (MB)
18 int ram = 4096; //8192; //vm memory (MB)
19 long bw = 1000;
20 int pesNumberm = 2; //number of cpus
21 String vmm = "Xen"; //VMM name
22
23 //create Master
24 Vm vmmaster = new Vm(vmidm, brokerId, mipism, pesNumberm, ram, bw, sizem, vmm, new CloudletSchedulerTimeShared());
25
26 //VM description
27 int vmid = 1;
28 int mips = 750;
29 long size = 250000; //image size (MB)
30 int ram = 2048; //vm memory (MB)
31 long bw = 1000;
32 int pesNumber = 1; //number of cpus
33
34 //create four Slaves
35 Vm vm1 = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared());
36 vmid++;
37 Vm vm2 = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared());
38 vmid++;
39 Vm vm3 = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared());
40 vmid++;
41 Vm vm4 = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared());
42 //add the VMs to the vmlist
43 vmlist.add(vmmaster);
44 vmlist.add(vm1);
45 vmlist.add(vm2);
46 vmlist.add(vm3);
47 vmlist.add(vm4);
48 //submit vm list to the broker
49 broker.submitVmList(vmlist);

```

Figure 24 Initialisation des paramètres du système de simulation.

À des fins de simulation, elle est réalisée à l'aide du logiciel CloudSim, qui est un cadre de modélisation et de simulation d'environnements hétérogènes avec des interfaces et des services de cloud computing (Alwasel *et al.*, 2019). Il fournit les infrastructures et les capacités nécessaires pour évaluer et analyser les applications Big Data avec les fonctionnalités de MapReduce (interactions, tâches distribuées et flux de données). Nous avons créé un centre de données, qui contient un hôte avec des machines virtuelles (1 maître et 4 esclaves). Cependant, chaque maison intelligente est représentée par une tâche (cloudlet). Le broker attribue et envoie les tâches à la VM maître dans laquelle il les redirige vers la VM esclave. Ensuite, les esclaves exécutent les tâches en parallèle et

transmettent les résultats. La figure suivante présente certains des résultats de la simulation (figure 25 et 26) :

```
Starting CloudSimOntology...
Starting Simulation for 10 Smarthouses...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.0: Broker: Trying to Create VM #1 in Datacenter_0
0.0: Broker: Trying to Create VM #2 in Datacenter_0
0.0: Broker: Trying to Create VM #3 in Datacenter_0
0.0: Broker: Trying to Create VM #4 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: VM #1 has been created in Datacenter #2, Host #0
0.1: Broker: VM #2 has been created in Datacenter #2, Host #0
0.1: Broker: VM #3 has been created in Datacenter #2, Host #0
0.1: Broker: VM #4 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
0.1: Broker: Sending cloudlet 1 to VM #1
0.1: Broker: Sending cloudlet 2 to VM #2
0.1: Broker: Sending cloudlet 3 to VM #3
0.1: Broker: Sending cloudlet 4 to VM #4
0.1: Broker: Sending cloudlet 5 to VM #1
0.1: Broker: Sending cloudlet 6 to VM #2
0.1: Broker: Sending cloudlet 7 to VM #3
0.1: Broker: Sending cloudlet 8 to VM #4
0.1: Broker: Sending cloudlet 9 to VM #1
0.1: Broker: Sending cloudlet 10 to VM #2
2.1: Broker: Cloudlet 0 received
5.433333333333333: Broker: Cloudlet 3 received
5.433333333333333: Broker: Cloudlet 7 received
5.433333333333333: Broker: Cloudlet 4 received
5.433333333333333: Broker: Cloudlet 8 received
8.097333333333333: Broker: Cloudlet 1 received
8.097333333333333: Broker: Cloudlet 5 received
8.097333333333333: Broker: Cloudlet 9 received
8.097333333333333: Broker: Cloudlet 2 received
8.097333333333333: Broker: Cloudlet 6 received
8.097333333333333: Broker: Cloudlet 10 received
8.097333333333333: Broker: All Cloudlets executed. Finishing...

Simulation completed for 10 Smarthouses.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time   Finish Time
0             SUCCESS  2                 0       02.00  00.10       02.10
3             SUCCESS  2                 3       05.33  00.10       05.43
7             SUCCESS  2                 3       05.33  00.10       05.43
4             SUCCESS  2                 4       05.33  00.10       05.43
8             SUCCESS  2                 4       05.33  00.10       05.43
1             SUCCESS  2                 1       08.00  00.10       08.10
5             SUCCESS  2                 1       08.00  00.10       08.10
9             SUCCESS  2                 1       08.00  00.10       08.10
2             SUCCESS  2                 2       08.00  00.10       08.10
6             SUCCESS  2                 2       08.00  00.10       08.10
10            SUCCESS  2                 2       08.00  00.10       08.10
Average excution time:0.8
CloudSimOntology finished!
```

Figure 25 Simulation de scénarios de temps d'exécution dans 10 maisons intelligentes.

```
Starting CloudSimOntology...
Starting Simulation for 20 Smarthouses...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.0: Broker: Trying to Create VM #1 in Datacenter_0
0.0: Broker: Trying to Create VM #2 in Datacenter_0
0.0: Broker: Trying to Create VM #3 in Datacenter_0
0.0: Broker: Trying to Create VM #4 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: VM #1 has been created in Datacenter #2, Host #0
0.1: Broker: VM #2 has been created in Datacenter #2, Host #0
0.1: Broker: VM #3 has been created in Datacenter #2, Host #0
0.1: Broker: VM #4 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
0.1: Broker: Sending cloudlet 1 to VM #1
0.1: Broker: Sending cloudlet 2 to VM #2
0.1: Broker: Sending cloudlet 3 to VM #3
0.1: Broker: Sending cloudlet 4 to VM #4
0.1: Broker: Sending cloudlet 5 to VM #1
0.1: Broker: Sending cloudlet 6 to VM #2
0.1: Broker: Sending cloudlet 7 to VM #3
0.1: Broker: Sending cloudlet 8 to VM #4
0.1: Broker: Sending cloudlet 9 to VM #1
0.1: Broker: Sending cloudlet 10 to VM #2
0.1: Broker: Sending cloudlet 11 to VM #1
0.1: Broker: Sending cloudlet 12 to VM #2
0.1: Broker: Sending cloudlet 13 to VM #3
0.1: Broker: Sending cloudlet 14 to VM #4
0.1: Broker: Sending cloudlet 15 to VM #0
0.1: Broker: Sending cloudlet 16 to VM #1
0.1: Broker: Sending cloudlet 17 to VM #2
0.1: Broker: Sending cloudlet 18 to VM #3
0.1: Broker: Sending cloudlet 19 to VM #4

Simulation completed for 20 Smarthouses.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time   Finish Time
0             SUCCESS  2                 0       00.11  00.10       00.21
1             SUCCESS  2                 1       01.00  00.10       01.10
5             SUCCESS  2                 1       01.00  00.10       01.10
9             SUCCESS  2                 1       01.00  00.10       01.10
13            SUCCESS  2                 1       01.00  00.10       01.10
17            SUCCESS  2                 1       01.00  00.10       01.10
2             SUCCESS  2                 2       01.00  00.10       01.10
6             SUCCESS  2                 2       01.00  00.10       01.10
10            SUCCESS  2                 2       01.00  00.10       01.10
14            SUCCESS  2                 2       01.00  00.10       01.10
18            SUCCESS  2                 2       01.00  00.10       01.10
3             SUCCESS  2                 3       01.00  00.10       01.10
7             SUCCESS  2                 3       01.00  00.10       01.10
11            SUCCESS  2                 3       01.00  00.10       01.10
15            SUCCESS  2                 3       01.00  00.10       01.10
19            SUCCESS  2                 3       01.00  00.10       01.10
4             SUCCESS  2                 4       01.00  00.10       01.10
8             SUCCESS  2                 4       01.00  00.10       01.10
12            SUCCESS  2                 4       01.00  00.10       01.10
16            SUCCESS  2                 4       01.00  00.10       01.10
20            SUCCESS  2                 4       01.00  00.10       01.10
Average excution time:1.00
CloudSimOntology finished!
```

Figure 26 Simulation de scénarios de temps d'exécution dans 20 maisons intelligentes.

Pour évaluer le temps de traitement, nous comparons l'efficacité de l'architecture en termes de temps d'exécution avec l'ICA-BEMS (Degha et al., 2019) dans différents cas. La figure 25 montre les résultats obtenus qui démontrent l'efficacité du système proposé (grâce aux outils Big Data) et maintient une durée approximativement courte (de 1 sec) contrairement à la durée d'évolution de l'autre système.

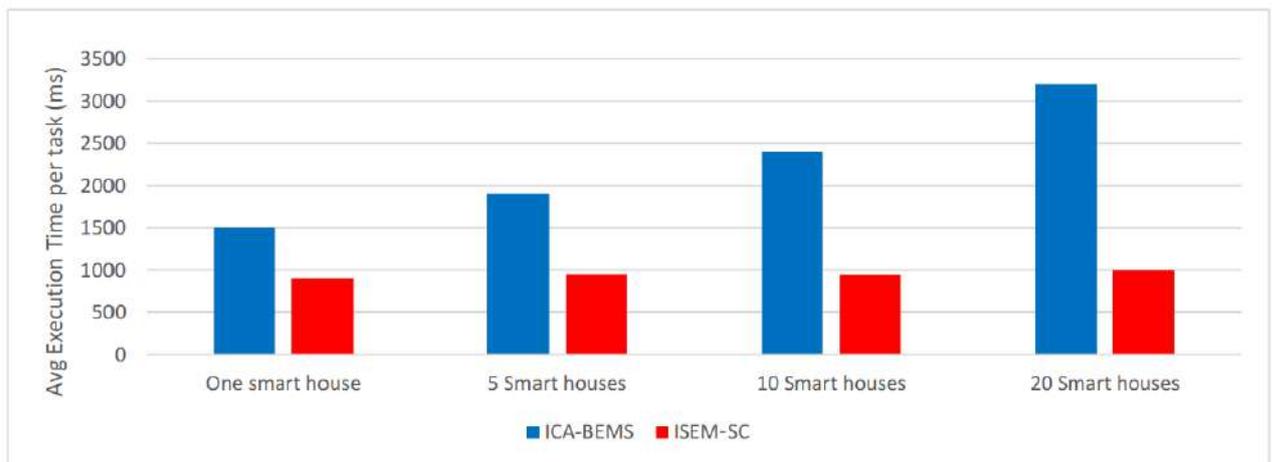


Figure 27 Le temps d'exécution dans différents scénarios.

V.4 Discussion

La configuration de l'environnement de simulation a été créée selon les critères mentionnés ci-dessus. Les résultats obtenus prouvent clairement l'utilité de combiner les technologies (ontologie, Big Data et MAS) pour réduire la consommation d'énergie sans affecter le confort des habitants. Ces outils réduisent le coût du matériel utilisé dans chaque maison, assurent un haut niveau de disponibilité du système, offrent une évolutivité aux villes intelligentes en exploitant l'environnement Big Data qui peut traiter un grand volume de données en temps réel et répond principalement au défi du volume et de la vitesse. Il assure l'autonomie et réduit le flux de données sur le réseau en utilisant le SMA. En outre, le système surveille la demande d'énergie en période de pointe et offre au fournisseur d'énergie une bonne opportunité de s'occuper de la stratégie de distribution et d'assurer une politique durable et à long terme en exploitant les connaissances et en analysant l'historique des données. Il contribue donc à réduire la pollution de l'environnement et à promouvoir la durabilité environnementale (voir tableau 12).

Aspects Travaux	Ontology	MAS	Big Data	Réduire l'énergie	Cout de matériel	confort de Résident	Efficacité
MAS-GA	Non	Oui	Non	Moyen	Moyen	Élevé	Moyen
ICA-BEMS	Oui	Non	Non	Moyen	Élevé	Moyen	Bas
ISEM-SC	Oui	Oui	Oui	Élevé	Bas	Élevé	Élevé

Tableau 12 Comparaison des technologies utilisées et leurs avantages dans les travaux expérimentés.

Avantages Travaux	Intelligence	Comprendre le comportement	Cooperation	Flux de donnees	Autonomie	Prédiction	Traitement au temps reel	Temps de traitement	Evolutivité	Disponibilité
MAS-GA	<i>Oui</i>	<i>Oui</i>	<i>Oui</i>	<i>Moyen</i>	<i>Oui</i>	<i>Oui</i>	<i>Non</i>	<i>Moyen</i>	<i>Oui</i>	<i>Moyen</i>
ICA-BEMS	<i>Oui</i>	<i>Oui</i>	<i>Non</i>	<i>Élevé</i>	<i>Non</i>	<i>Oui</i>	<i>Non</i>	<i>Élevé</i>	<i>Non</i>	<i>Bas</i>
ISEM-SC	<i>Oui</i>	<i>Oui</i>	<i>Oui</i>	<i>Bas</i>	<i>Oui</i>	<i>Oui</i>	<i>Oui</i>	<i>Bas</i>	<i>Oui</i>	<i>Élevé</i>

Tableau 13 Comparaison des caractéristiques principales d'économie d'énergie dans les travaux expérimentés.

V.5 Conclusion

Dans ce chapitre, nous avons présenté les résultats obtenus lors des expérimentations réalisées pour chaque scénario. Pour valider l'architecture proposée, nous avons examiné deux métriques de base, le taux de consommation d'énergie et le temps d'exécution des tâches. Deux expériences de simulation ont été établies à cet effet. Une économie d'énergie efficace avec un taux de 15,62% est enregistrée lors de la première

expérimentation. En outre, dans la deuxième expérimentation au cours de différents scénarios, le système maintient une durée de traitement approximativement courte (de 1 seconde) contrairement à la durée d'évolution des autres systèmes. Les résultats reflètent clairement les opportunités offertes lorsque les différentes technologies sont combinées ensemble pour fournir un système fiable permettant de réduire efficacement l'énergie dans les villes intelligentes. Le système proposé permet d'avoir une meilleure vision la gestion d'énergie et de la protection de l'environnement tout en assurant le confort des habitants.

Conclusion générale

Au cours des dernières années, l'évolution massive de données dans tous les domaines engendre un véritable défi pour découvrir, exploiter et partager ces immenses variétés de données. Cette explosion exponentielle est produite par le développement de nombre des entreprises et leurs activités, et surtout l'apparition et la progression remarquable des réseaux sociaux (Facebook, Twitter, Instagram, YouTube, etc.). Cette réalité provoque une situation paradoxale, comment permettre l'évolution du Big Data et de faire exploiter et profiter de ces données au même temps ? La sémantisation de données apparaît comme une solution prometteuse pour surmonter ce défi. L'intégration de la sémantique dans le Big Data devient une pratique essentielle de la vie quotidienne. Car, elle offre la possibilité d'interaction et de partage des connaissances. L'utilisation de l'ontologie donne aussi une grande capacité pour émerger et partager des informations hétérogènes.

Dans cette thèse, nous avons donné en premier lieu un état de l'art sur les Big Data avec une description de ces aspects, leur évolution, les technologies et outils de Big Data avec les domaines d'application. Ensuite, un aperçu général est évoqué sur le web sémantique et les ontologies, leurs composants, les différents types d'ontologies, processus de construction, les langages et les éditeurs d'ontologies. Dans la partie qui suit, nous avons examiné le concept d'intégration de la sémantique dans le Big Data tout en éclaircissant le rôle et l'importance de ce processus, les approches d'ontologies utilisées, cette partie est conclue par une comparaison et discussion de quelques travaux qui ont utilisé les techniques d'intégration dans des domaines d'actualité.

Notre contribution est exposée dans le chapitre 5, qu'il s'agit de la gestion de l'énergie dans les villes intelligentes. Cette dernière est devenue l'un des défis les plus critiques qui se posent dans les solutions de durabilité. L'évolution intensive des données génère une énorme quantité de données distribuées et hétérogènes, ce qui constitue un sérieux challenge pour traiter le grand volume et l'hétérogénéité de ces données. Pour résoudre ces problèmes, des outils Big Data et des technologies d'ontologies ont été utilisés pour réaliser une efficace économie d'énergie dans les villes intelligentes. Dans cette partie, nous avons proposé un système intelligent de gestion de l'énergie dans les villes intelligentes (ISEM-SC) qui vise à réduire la consommation d'énergie. Dans ce contexte, nous avons d'abord passé en revue le contexte de l'intégration des Big Data et de la sémantique avec les travaux connexes. Ensuite, nous avons présenté une architecture multi couches. Le système

Conclusion générale

proposé combine différentes technologies (Big Data, ontologie et MAS) utilisées pour réduire la consommation d'énergie sans affecter le confort des habitants.

Ce cadre offre plusieurs opportunités, à savoir :

- ✓ Réduire la consommation d'énergie dans chaque maison intelligente et donc dans toutes la ville intelligente;
- ✓ Réduire le coût du matériel utilisé dans chaque maison et traite les données en temps court;
- ✓ Garantir un niveau élevé de disponibilité du système ;
- ✓ Offrir une évolutivité en tirant parti de l'environnement Big Data qui peut traiter d'énormes quantités de données en temps réel ;
- ✓ Offrir une autonomie et réduire le flux de données sur le réseau en utilisant le SMA;
- ✓ Surveiller la demande d'énergie en période de pointe et évite les pannes de réseau;
- ✓ Fournir une politique efficace de distribution d'énergie à long terme ;
- ✓ Contribuer également à réduire la pollution de l'environnement et à promouvoir la durabilité environnementale.

Les résultats obtenus dans différents scénarios montrent une économie d'énergie efficace avec un taux de 15,62%. En outre, le système maintient une durée de traitement approximativement courte (de 1 seconde) contrairement à la durée d'évolution des autres systèmes.

La contribution de cette thèse se caractérise dans deux visions essentielles :

- ✓ Elle consiste en premier lieu de donner une description générale de la sémantique dans le contexte Big Data tout en montrant la position de chaque domaine par rapport à l'autre et comment la sémantique peut aider à résoudre les inconvénients associant le développement du Big Data.
- ✓ Ce travail propose une approche efficace qui apporte des solutions plus complètes d'intégration d'ontologies dans l'environnement Big Data. Dont on a utilisé une diversité technologies (Big Data, ontologie et SMA) pour réduire la consommation d'énergie, promouvoir le changement de comportement des habitants, offrir le confort requis et mettre en œuvre une politique énergétique efficace à long terme dans un environnement intelligent et durable.

Perspectives

Dans les travaux futurs, nous nous concentrerons sur les capacités fournies par les Big Data, l'ontologie et le SMA d'une part, modéliser une architecture étendue capable de combiner la gestion de l'énergie (électrique et Gaz), avec les sources renouvelables (énergie solaire), et d'autre part créer des modèles adaptable pour chaque client en matière des besoins et la saison (été/hiver). Ce qui donne un système complet pour le contrôle efficace d'énergie dans les villes intelligentes afin d'offrir une vie aisée aux citoyens et garantir la durabilité des villes intelligentes.

Liste des travaux

1. Sayah, Z., Kazar, O. and Ghenabzia, A. (2018), "Semantic integration in Big Data applications opportunities and challenges", *1st International Conference on Artificial Intelligence and Its Applications AIAP'2018. EL OUED, ALGERIA.*, pp. 187–193.
2. Sayah, Z., Kazar, O. and Ghenabzia, A. (2020), "Semantic Integration in Big Data":, *Journal Of Mobile Multimedia*, Vol. 15 No. 3, pp. 191–238.
3. Sayah, Z., Kazar, O., Lejdel, B., Laouid, A. and Ghenabzia, A. (2020), "An Intelligent System for Energy Management in Smart Cities based on Big Data and Ontology", *Smart and Sustainable Built Environment*, article in press, available at:<https://doi.org/10.1108/SASBE-07-2019-0087>.
4. Ghenabzia, A., Kazar, O., Merizig, A., Sayah, Z., & Zoubeidi, M. (2019, July). Drilling Stuck Pipe Prediction in Algerian Oil Fields: Time Series Forecasting Approach. In *International Conference on Advanced Intelligent Systems for Sustainable Development* (pp. 262-275). Springer, Cham.
5. Ghenabzia, A., Kazar, O., Merizig, A., Sayah, Z., & Zoubeidi, M. (2020). Multi-agent approach for data mining-based bagging ensembles to improve the decision process for big data. *International Journal of Information and Communication Technology*, 17(4), 380-402.

References

- Abbes, H. and Gargouri, F. (2018), “MongoDB-Based Modular Ontology Building for Big Data Integration”, *Journal on Data Semantics*, Vol. 7 No. 1, pp. 1–27.
- Alatrish, E.S. (2013), “Comparison Some of Ontology Editors”, *Journal of Management Information Systems*, Vol. 8 No. 2, pp. 18–24.
- Alvarez, M.G., Morales, J. and Kraak, M.J. (2019), “Integration and exploitation of sensor data in smart cities through event-driven applications”, *Sensors (Switzerland)*, Vol. 19 No. 6, p. 1372.
- Alwasel, K., Calheiros, R. N., Garg, S., Buyya, R., & Ranjan, R. (2019), “BigDataSDNSim: A Simulator for Analyzing Big Data Applications in Software-Defined Cloud Data Centers”, *ArXiv Preprint ArXiv:1910.04517*, available at: <https://arxiv.org/abs/1910.04517> (accessed 13 October 2019).
- Anvari-Moghaddam, A., Rahimi-Kian, A., Mirian, M. S., & Guerrero, J.M. (2017), “A multi-agent based energy management solution for integrated buildings and microgrid system”, *Applied Energy*, Vol. 203, pp. 41–56.
- Bansal, S.K. and Kagemann, S. (2015), “Semantic Extract-Transform-Load framework for Big Data Integration”, *Computer*, Vol. 48 No. 3, pp. 42–50.
- Bello-orgaz, G., Jung, J.J. and Camacho, D. (2016), “Social big data : Recent achievements and new challenges”, *Information Fusion*, Elsevier B.V., Vol. 28, pp. 45–59.
- Beneventano, D. and Vincini, M. (2019), “Foreword to the Special Issue: ‘Semantics for Big Data Integration’”, *Information*, MDPI AG, Vol. 10 No. 2, p. 68.
- Bermúdez-Edo, M., Della Valle, E. and Palpanas, T. (2016), “Semantic Challenges for the Variety and Velocity Dimensions of Big Data”, *International Journal on Semantic Web and Information Systems*, Vol. 12 No. 4, pp. 1–13.
- Berners-Lee, T., Hendler, J. and O, T.L. (2001), “The Semantic Web”, *In Scientific American*, Vol. 284 No. 5, pp. 34–43.
- Bhadani, A. and Jothimani, D. (2016), “Big Data : Challenges , Opportunities , and Realities”, *Effective Big Data Management and Opportunities for Implementation*, IGI Global, pp. 1–24.
- Bokolo, A. J., Petersen, S. A., Ahlers, D., & Krogstie, J. (2019), “API deployment for big data management towards sustainable energy prosumption in smart cities-a layered architecture perspective”, *International Journal of Sustainable Energy*, Vol. 39 No. 3, pp. 263–289.
- Bourekache, S., Kazar, O., Kahloul, L., Gargouri, F. and Aïcha-Nabila, B. (2016), “Un Environnement Sémantique à Base d’Agents pour la Formation à Distance (E-Learning).”, *10ième Édition de La Conférence Sur Avancés Des Systèmes Décisionnels-ASD 2016*,.
- Bourekache, S. (2014), "Un environnement sémantique à base d'agents pour la formation à distance (E-Learning)", Doctoral dissertation, Université de Biskra.
- Boury-Brisset, A.C. (2013), “Managing Semantic Big Data for Intelligence”, *In STIDS Proceedings*, pp. 41–47.
- Bukhari, S.A.C., Bashir, A.K. and Malik, K.M. (2018), “Semantic Web in the Age of Big Data : A Perspective”, *OSF Preprints*, available at:<https://doi.org/10.31219/osf.io/mwjtg>.
- Butt, A. A., Khan, S., Ashfaq, T., Javaid, S., Sattar, N. A., & Javaid, N. (2019), “A cloud and fog based architecture for energy management of smart city by using meta-heuristic techniques”, *15th*

Bibliographie

- International Wireless Communications and Mobile Computing Conference, IWCMC 2019*, IEEE, pp. 1588–1593.
- Calvanese, D. (2015), “Ontologies for Data Integration”, *IJCAI 2015 Workshop on Formal Ontologies for Artificial Intelligence.*, Buenos-Aires, Argentina, pp. 1–67.
- Chahdi, H. (2017), “*Apports Des Ontologies à l’analyse Exploratoire Des Images Satellitaires*” (Doctoral Dissertation), Montpellier - France.
- Chandrasekaran, B., Josephson, J.R. and Benjamins, V.R. (1999), “What Are Ontologies , and Why Do We Need Them ?”, *IEEE Intelligent Systems and Their Applications*, Vol. 14 No. 1, pp. 20–26.
- Chen, H., Chiang, R.H.L. and Storey, V.C. (2012), “Business intelligence and analytics: from big data to big impact”, *MIS Quarterly*, Vol. 36 No. 4, pp. 1165–1188.
- Chen, M., Mao, S. and and Liu, Y. (2014), “Big Data : A Survey”, *Mobile Networks and Applications*, Vol. 19 No. 2, pp. 171–209.
- Chodorow, K., Bradshaw, S. and Eoin, B. (2019), *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, edited by Spencer, A., 3rd ed., O’Reilly Media., USA, available at: <http://shop.oreilly.com/product/0636920049531.do> (accessed 10 January 2020).
- Coneglian, C.S., Fusco, E., Segundo, J.E.S., Junior, V.A.P. and de Castro Botega, L. (2016), “Ontological Semantic Agent in the Context of Big Data: A Tool Applied to Information Retrieval in Scientific Research”, *In New Advances in Information Systems and Technologies.*, Springer, cham, pp. 307–316.
- Corcho, O., Fernández-López, M. and Gómez-Pérez, A. (2003), “Methodologies , tools and languages for building ontologies . Where is their meeting point ?”, *Data & Knowledge Engineering*, Vol. 46 No. 1, pp. 41–64.
- Degha, H.E., Laallam, F.Z. and Said, B. (2019), “Intelligent context-awareness system for energy efficiency in smart building based on ontology”, *Sustainable Computing: Informatics and Systems*, Vol. 21, pp. 212–233.
- Delgoshaei, P., Heidarinejad, M. and Austin, M. (2018), “Combined Ontology-Driven and Machine Learning Approach To Monitoring of Building Energy Consumption”, *2018 Building Performance Modeling Conference and SimBuild*, ASHRAE, Chicago, IL, IBPSA-USA, available at: https://www.techstreet.com/standards/combined-ontology-driven-and-machine-learning-approach-to-monitoring-of-building-energy-consumption?product_id=2026638 (Accessed: 12 May 2019).
- Ding, L., Kolari, P., Ding, Z., Avancha, S., Finin, T. and Joshi, A. (2007), “Using Ontologies in the Semantic Web : A Survey”, *Ontologies. Springer, Boston, MA*, pp. 79–113.
- Djilani, Z. (2017), *Donner Une Autre Vie à Vos Besoins Fonctionnels : Une Approche Dirigée p Ar l ’ Entreposage et l ’ Analyse En Ligne, Doctorat Desertation . ISAE-ENSMA Ecole Nationale Supérieure de Mécanique et d’Aérotechnique-Poitiers.*
- Dritsa, D. and Biloría, N. (2018), “Towards a multi-scalar framework for smart healthcare”, *Smart and Sustainable Built Environment*, Vol. 7 No. 1, pp. 33–52.
- Eine, B., Jurisch, M. and Quint, W. (2017), “Ontology-Based Big Data Management”, *Systems*, Vol. 5 No. 3, p. 45.
- Ejaz, W., Naeem, M., Shahid, A., Anpalagan, A., & Jo, M. (2017), “Efficient Energy Management for the Internet of Things in Smart Cities”, *IEEE Communications Magazine*, Vol. 55 No. 1, pp. 84–91.
- Emani, C.K., Cullot, N. and Nicolle, C. (2015), “Understandable Big Data : A survey”, *Computer Science*

Bibliographie

- Review, Elsevier Inc., Vol. 15, pp. 70–81.
- Estrada, E., Vargas, M.P.M., Gómez, J., Negron, A.P.P., López, G.L. and Maciel, R. (2019), “Smart cities big data algorithms for sensors location”, *Applied Sciences (Switzerland)*, Vol. 9 No. 19, p. 4196.
- Fensel, D. (2001), “Ontologies”, *In Ontologies. Springer, Berlin, Heidelberg.*, pp. 11–18.
- Ghafour, S.A. (2004), *Méthodes et Outils Pour l'Intégration Des Ontologies, Rapport de Stage de DEA, Laboratoire d'InfoRmatique En Images et Systèmes d'information LIRIS, Lyon.*
- De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A. and Rosati, R. (2018), “Using ontologies for semantic data integration”, in Springer, C. (Ed.), *In A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years*, pp. 187–202.
- Gómez-pérez, A., Corcho, O. and Madrid, U.P. De. (2002), “Ontology Languages for the Semantic Web”, *IEEE Intelligent Systems*, Vol. 17 No. 1, pp. 54–60.
- Grolinger, K., Capretz, M.A.M. and Seewald, L. (2016), “Energy consumption prediction with big data: Balancing prediction accuracy and computational resources”, *2016 IEEE International Congress on Big Data (BigData Congress)*, IEEE, pp. 157–164.
- Gruber, T.R. (1995), “Toward Principles for the Design Toward Principles for the Design of Ontologies Used for Knowledge Sharing”, *International Journal Human-Computer*, Vol. 43 No. 5–6, pp. 907–928.
- Guarino, N. (1997), “Semantic Matching: Formal Ontological Distinctions for Information Organization, Extraction, and Integration”, *International Summer School on Information Extraction. Springer, Berlin, Heidelberg*, pp. 139–170.
- Guarino, N., Oberle, D. and Staab, S. (2009), “What Is an Ontology?”, pp. 1–17.
- Guido, A.L. and Paiano, R. (2010), “Semantic Integration of Information Systems”, *International Journal of Computer Networks and Communications (IJCNC)*, Vol. 2 No. 1, pp. 48–64.
- Hadi, H.J., Shnain, A.H., Hadishaheed, S. and Ahmad, A.H. (2015), “Big Data and Five V's Characteristics”, *International Journal of Advances in Electronics and Computer Science*, Vol. 2 No. 1, pp. 16–23.
- Hamdouni, M. El, Hanafi, H., Bouktib, A. and Bahra, M. (2018), “Sentiment Analysis in Social Media with a Semantic Web Based Approach: Application to the French Presidential Elections 2017”, *Proceedings of the Mediterranean Symposium on Smart City Applications. Springer, Cham*, pp. 470–482.
- Hamida, S. (2014). Une approche basée agent mobile pour le m-service web sémantique (Doctoral dissertation, Université Mohamed Khider Biskra).
- Harmouch, F. Z., Ebrahim, A. F., Esfahani, M. M., Krami, N., Hmina, N., & Mohammed, O.A. (2019), “An optimal energy management system for real-time operation of multiagent-based microgrids using a T-cell algorithm”, *Energies*, Vol. 12 No. 15, p. 3004.
- Van Heijst, G., Schreiber, A.T. and Wielinga, B.J. (1997), “Using explicit ontologies in KBS development”, *International Journal of Human-Computer Studies*, Vol. 46 No. 2–3, pp. 183–292.
- Horrocks, I., Giese, M., Kharlamov, E. and Waaler, A. (2016), “Using Semantic Technology to Tame the Data Variety Challenge”, *IEEE Internet Computing*, Vol. 20 No. 6, pp. 62–66.
- Horrocks, I., Inference, N., Patel-schneider, P.F., Technologies, L., Boley, H., Tabet, S., Grosz, B., et al. (2004), “SWRL: A Semantic Web Rule Language Combining OWL and RuleML”, No. May.
- Howell, S., Rezgui, Y., Hippolyte, J. L., Jayan, B. and Li, H. (2017), “Towards the next generation of smart grids: Semantic and holonic multi-agent management of distributed energy resources”, *Renewable and Sustainable Energy Reviews*, Vol. 77, pp. 193–214.

Bibliographie

- Hui, J., Li, L. and Zhang, Z. (2018), “Integration of Big Data : A Survey”, *International Conference of Pioneering Computer Scientists, Engineers and Educators*. Springer, Singapore, Springer Singapore, pp. 101–121.
- “Infoclimat”. (2020), , available at: <https://www.infoclimat.fr/observations-meteo/archives/13/juillet/2019/el-oued/60559.html?graphiques> (accessed 10 January 2020).
- Jawhar, I., Mohamed, N. and Al-Jaroodi, J. (2018), “Networking architectures and protocols for smart city systems”, *Journal of Internet Services and Applications*, Springer London, Vol. 9 No. 1, p. 26.
- Jeong, S.R. and Ghani, I. (2014), “Semantic Computing for Big Data : Approaches , Tools , and Emerging Directions”, *KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS*, Vol. 8 No. 6, pp. 2022–2042.
- Kertiou, I., Benharzallah, S., Kahloul, L., Beggas, M., Euler, R., Laouid, A. and Bounceur, A. (2018), “A dynamic skyline technique for a context-aware selection of the best sensors in an IoT architecture”, *Ad Hoc Networks*, Elsevier B.V., Vol. 81, pp. 183–196.
- Kim, J., Wang, G. and Bae, S.T. (2014), “A Survey of Big Data Technologies and How Semantic Computing Can Help”, *International Journal of Semantic Computing*, Vol. 8 No. 1, pp. 99–117.
- Kofinas, P., Dounis, A.I. and Vouros, G.A. (2018), “Fuzzy Q-Learning for multi-agent decentralized energy management in microgrids”, *Applied Energy*, Vol. 219, pp. 53–67.
- Kott, J. and Kott, M. (2019), “Generic ontology of energy consumption households”, *Energies*, Vol. 12 No. 19, pp. 1–19.
- Koutsomitropoulos, D.A. and Kalou, A.K. (2017), “A standards-based ontology and support for Big Data Analytics in the”, *ICT Express*, Elsevier B.V., Vol. 3 No. 2, pp. 57–61.
- Laallam, F.Z., Kherfi, M.L. and Benslimane, S.M. (2014), “A survey on the complementarity between database and ontologies : principles and research areas”, *Int. J. Computer Applications in Technology*, Vol. 49 No. 2, pp. 166–187.
- Labrinidis, A. and Jagadish, H. V. (2012), “Challenges and Opportunities with Big Data”, *Proceedings of the VLDB Endowment*, Vol. 5 No. 12, pp. 2032–2033.
- Lee, I. (2017), “Big data : Dimensions , evolution , impacts , and challenges”, *Business Horizons*, Vol. 60 No. 3, pp. 293–303.
- Lejdel, B. and Kazar, O. (2018), “Using a Hybrid Approach to Optimize Consumption Energy of Building and Increase Occupants’ Comfort Level in Smart City”, *Artificial Intelligence in Renewable Energetic Systems*, Springer, Cham, Tibaza, pp. 67–76.
- Lork, C., Choudhary, V., Ul Hassan, N., Tushar, W., Yuen, C., Ng, B.K.K., Wang, X., et al. (2019), “An Ontology-Based Framework for Building Energy Management with IoT”, *Electronics (Switzerland)*, Vol. 8 No. 5, pp. 1–15.
- Ma, Z., Schultz, M.J., Christensen, K., Værbak, M., Demazeau, Y. and Jørgensen, B.N. (2019), “The application of ontologies in multi-agent systems in the energy sector: A scoping review”, *Energies*, Vol. 12 No. 16, pp. 1–31.
- Mahapatra, C., Moharana, A.K. and Leung, V.C.M. (2017), “Energy management in smart cities based on internet of things: Peak demand reduction and energy savings”, *Sensors (Switzerland)*, Vol. 17 No. 12, pp. 1–21.
- Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. and Byers, A.H. (2011), *Big Data:*

Bibliographie

- The next Frontier for Innovation, Competition, and Productivity, Technical Report, McKinsey Global Institute.*, New York, New York, USA, available at: <https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/big-data-the-next-frontier-for-innovation> (accessed 18 February 2020).
- Merizig, A., Kazar, O. and Sanchez, M.L. (2019), “A multi-agent system approach for service deployment in the cloud”, *International Journal of Communication Networks and Distributed Systems*, Vol. 1 No. 23, pp. 69–92.
- Mezghani, E., Exposito, E., Drira, K. and Silveira, M. Da. (2015), “A Semantic Big Data Platform for Integrating Heterogeneous Wearable Data in Healthcare”, *Journal of Medical Systems*, Vol. 39 No. 12, p. 185.
- Mizoguchi, R. (2004), “Tutorial on ontological engineering Part 2: Ontology development, tools and languages”, *New Generation Computing*, Vol. 22 No. 1, pp. 61–96.
- Le Noac’H, P., Costan, A., A. and Bougé, L. (2017), “A performance evaluation of Apache Kafka in support of big data streaming applications”, *2017 IEEE International Conference on Big Data, Big Data 2017*, IEEE, pp. 4803–4806.
- Obrst, L., Grüninger, M., Baclawski, K., Bennett, M., Brickley, D., Berg-Cross, G. and ... and Lange, C. (2014), “Semantic Web and Big Data Meets Applied Ontology”, *In Ontology Summit 2014*, p. 15.
- Ostrowski, D., Rychtycky, N., MacNeille, P. and Kim, M. (2016), “Integration of Big Data Using Semantic Web Technologies”, *In 2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, IEEE, pp. 382–385.
- Oussous, A., Benjelloun, F., Ait, A. and Belfkih, S. (2017), “Big Data technologies : A survey”, *Journal of King Saud University - Computer and Information Sciences*, King Saud University, Vol. 30 No. 4, pp. 431–448.
- Özköse, H., Uđ, P.L.Q. and Gencer, C. (2015), “Yesterday , Today and Tomorrow of Big Data”, *Procedia-Social and Behavioral Sciences*, Vol. 195, pp. 1042–1050.
- Panahiazar, M., Taslimitehrani, V. and Jadhav, A. (2014), “Empowering Personalized Medicine with Big Data and Semantic Web Technology : Promises , Challenges , and Use Cases”, *IEEE International Conference on Big Data 2014*, pp. 790–795.
- Papastamatiou, I., Marinakis, V., Doukas, H., & Psarras, J. (2017), “A Decision Support Framework for Smart Cities Energy Assessment and Optimization”, *8th International Conference on Sustainability in Energy and Buildings, SEB-16, 11-13 September 2016, Turin, ITALY*, pp. 800–809.
- Pinto, H.S., Gómez-Pérez, A. and Martins, J.P. (1999), “Some Issues on Ontology Integration”, *In Proceedings of the IJCAI, Stockholm, Sweden. Stockholm, Sweden.*, pp. 1–7.
- Pinto, H.S. and Martins, J.P. (2001), “A Methodology for Ontology Integration”, *In Proceedings of the 1st International Conference on Knowledge Capture, Victoria, British Columbia, Canada.*, ACM, Victoria, British Columbia, Canada., pp. 131–138.
- Quboa, Q. and Mehandjiev, N. (2017), “Creating Intelligent Business Systems by Utilising Big Data and Semantics”, *In 2017 IEEE 19th Conference on Business Informatics (CBI), Vol. 2, IEEE.*, pp. 39–46.
- Raghavan, S., Simon, B.Y.L., Lee, Y.L., Tan, W.L. and Kee, K.K. (2020), “Data Integration for Smart Cities: Opportunities and Challenges”, in Alfred R., Lim Y., Haviluddin H., O.C. (Ed.), *Computational Science and Technology*, Springer Verlag, Singapore, pp. 393–403.

Bibliographie

- Rani, P. S., Suresh, R. M., & Sethukarasi, R. (2017), "Multi-level semantic annotation and unified data integration using semantic web ontology in big data processing", *Cluster Computing*, Vol. 22 No. 5, pp. 10401–10413.
- Rao, V.M., Kumari, V. V. and Silpa, N. (2015), "An Extensive Study on Leading Research Paths on Big Data Techniques and Technologies.", *International Journal of Computer Engineering & Technology*, Vol. 6 No. 12, pp. 20–34.
- Rodríguez-mazahua, L., Luis, J., Cervantes, S.J., Luis, J. and Alor-hernández, G. (2016), "A general perspective of Big Data : applications , tools ", *The Journal of Supercomputing*, Springer US, Vol. 72 No. 8, pp. 3073–3113.
- Saba, D., Sahli, Y., Abanda, F. H., Maouedj, R., & Tidjar, B. (2019), "Development of new ontological solution for an energy intelligent management in Adrar city", *Sustainable Computing: Informatics and Systems*, Vol. 21, pp. 189–203.
- Saettler, A.M., Llanes, K.R., Ivson, P., Nascimento, D.L.M. and Corseuil, E.T.L. (2017), "An Ontology-Driven Framework for Data Integration and Dynamic Service Composition : Case Study in the Oil & Gas Industry", *In 16th International Conference on WWW/Internet (ICWI)*, Vilamoura, Algarve, Portugal., pp. 79–86.
- Saggi, M.K. and Jain, S. (2018), "A survey towards an integration of big data analytics to big insights for value-creation", *Information Processing and Management*, Elsevier, Vol. 54 No. 5, pp. 758–790.
- Samuel, S.S.I. (2016), "A review of connectivity challenges in IoT-smart home", *3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, IEEE, pp. 1–4.
- Sayah, Z., Kazar, O. and Ghenabzia, A. (2018), "Semantic integration in Big Data applications opportunities and challenges", *1st International Conference on Artificial Intelligence and Its Applications AIAP'2018. EL OUED, ALGERIA.*, pp. 187–193.
- Sayah, Z., Kazar, O. and Ghenabzia, A. (2020), "Semantic Integration in Big Data":, *Journal Of Mobile Multimedia*, Vol. 15 No. 3, pp. 191–238.
- Sayah, Z., Kazar, O., Lejdel, B., Laouid, A. and Ghenabzia, A. (2020), "An Intelligent System for Energy Management in Smart Cities based on Big Data and Ontology", *Smart and Sustainable Built Environment*, Vol. in press, available at:<https://doi.org/10.1108/SASBE-07-2019-0087>.
- Sezer, O.B., Dogdu, E. and Ozbayoglu, A.M. (2018), "Context Aware Computing , Learning and Big Data in Internet of Things : A Survey", *IEEE Internet of Things Journal*, Vol. 5 No. 1, pp. 1–27.
- Shafik, W., Matinkhah, S.M. and Ghasemzadeh, M. (2020), "Internet of Things-Based Energy Management , Challenges , and Solutions in Smart Cities", *Journal of Communications Technology, Electronics and Computer Science*, Vol. 27 No. 1, pp. 1–11.
- Shvachko, K., Kuang, H., Radia, S. and Chansler, R. (2010), "The Hadoop Distributed File System", *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on. Ieee*, pp. 1–10.
- Soetedjo, A., Ismail Nakhoda, Y. and Saleh, C. (2019), "An embedded platform for testbed implementation of multi-agent system in building energy management system", *Energies*, Vol. 12 No. 19, p. 3655.
- Soomro, K., Bhutta, M.N.M., Khan, Z. and Tahir, M.A. (2019), "Smart city big data analytics: An advanced review", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 9 No. 5, pp. 1–25.
- Storey, V.C. and Song, I. (2017), "Data & Knowledge Engineering Big data technologies and Management :

Bibliographie

- What conceptual modeling can do”, *Data & Knowledge Engineering*, Elsevier B.V., Vol. 108 No. February, pp. 50–67.
- Thirunarayan, K. and Sheth, A. (2013), *Semantics-Empowered Approaches to Big Data Processing for Physical-Cyber-Social Applications*, *Semantics for Big Data: Papers from the AAAI Symposium*. AAAI Technical Report FS-13-04.
- Tsai, C.W., Lai, C.F., Chao, H.C. and Vasilakos, A. V. (2015), “Big data analytics : a survey”, *Journal of Big Data*, Springer International Publishing, pp. 1–32.
- Vandecasteele, A. (2012), *Modélisation Ontologique Des Connaissances Expertes Pour l ' Analyse de Comportements à Risque - Application à La Surveillance Maritime - "Doctoral Dissertation, Ecole Nationale Supérieure Des Mines de Paris*.
- Wache, H., Voegelé, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H. and Hübner, S. (2001), “Ontology-Based Integration of Information A Survey of Existing Approaches”, *IJCAI-01 Workshop: Ontologies and Information Sharing*, pp. 108–117.
- Wala, T., Chand, N. and Sharma, A.K. (2020), “Energy Efficient Data Collection in Smart Cities Using IoT”, in Singh P., Bhargava B., Paprzycki M., Kaushal N., H.W. (Ed.), *Handbook of Wireless Sensor Networks: Issues and Challenges in Current Scenario 's*, Springer, pp. 632–654.
- Wang, S.J. and Moriarty, P. (2019), “Energy savings from Smart Cities: A critical analysis”, *10th International Conference on Applied Energy (ICAE2018)*, Vol. 158, Elsevier Ltd, Hong Kong, China, pp. 3271–3276.
- Williams, J.W., Cuddihy, P., Mchugh, J., Aggour, K.S., Menon, A., Gustafson, S.M., Healy, T., et al. (2015), “Semantics for Big Data Access & Integration :Improving Industrial Equipment Design through Increased Data Usability”, *2015 IEEE International Conference on Big Data (Big Data)*, pp. 1103–1112.
- Wira, P., Szilagyi, I. and Wira, P. (2016), “Ontologies and Semantic Web for the Internet of Things - a survey Ontologies and Semantic Web for the Internet of Things – A Survey”, *42nd IEEE Industrial Electronics Conference (IECON 2016), Florence, Italy*, pp. 6949–6954.
- Yang, C., Huang, Q., Li, Z., Liu, K., & Hu, F. (2017), “Big Data and cloud computing : innovation opportunities and challenges”, *International Journal of Digital Earth*, Vol. 10 No. 1, pp. 13–53.
- Yang, X., Yang, M., Yang, D., & Huang, Y. (2018), “Research on Implementation of Knowledge Convergence Based on Apache Jena3”, *2018 International Conference on Computer Science, Electronics and Communication Engineering (CSECE 2018)*, Atlantis Press, Wuhan, China, available at:<https://doi.org/10.2991/csece-18.2018.101>.
- Yaqoob, I., Abaker, I., Hashem, T., Gani, A., Mokhtar, S., Ahmed, E., Badrul, N., et al. (2016), “Big data : From beginning to future”, *International Journal of Information Management*, Elsevier Ltd, Vol. 36 No. 6, pp. 1231–1247.
- Zikopoulos, P. and Eaton, C. (2011), “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, *McGraw-Hill Osborne Media*, McGraw-Hill Osborne Media.