الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -

Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Département d'Informatique

Mémoire de Fin d'études Master

Filière : Informatique

Option : Master Académique

Thème :

Manipulations avancées d'un système de médiation sémantique à base d'ontologie

Encadré Par :

M^me^ AGGOUNE Aicha

Présenté par :

CISSE Hawa Founè,

SOGOBA Colette Nassa

Juin 2016

Remerciements

En préambule de ce mémoire, nous rendons grâce à Allah le Très Miséricordieux qui nous a accordé la santé et le courage pour mener à bien ce travail.

Ces remerciements vont à l'endroit de l'ensemble du corps professoral et administratif de la faculté des Mathématiques, Informatique et des Sciences de la matière de l'Université 08 Mai 1945 de Guelma, pour la qualité, la richesse de leur formation et pour tous les efforts déployés pour assurer une formation actualisée à leurs étudiants.

Nous remercions notre encadreur Mme « Aggoune Aicha » pour sa patience, son indulgence et ses précieux conseils qui ont constitué un apport considérable sans lequel ce travail n'aurait pas pu être mené au bon port. Qu'elle trouve dans ces mots, nos considérations face à son attache à l'engagement et au travail bien fait.

Nous tenons également à remercier les membres du jury qui nous font l'honneur d'examiner ce modeste travail.

Nous remercions nos familles respectives pour tout le soutien qu'elles nous apportent.

Enfin, nous remercions toutes les personnes qui ont apporté leur contribution à la réalisation de ce travail.

Hawa et Colette

Dédicaces

C'est avec le cœur rempli de joie que je dédie ce mémoire

A mes chers et bien aimés parents, ***Aissata Maïga*** *et* ***Oumar Cissé****, en reconnaissance de leur amour, des valeurs nobles qu'ils m'ont inculquées, et de tous les sacrifices consentis pour me permettre d'atteindre cette étape de ma vie,*

Que ce modeste travail soit l'exaucement de leurs vœux et qu'ils y trouvent l'expression de mon éternelle gratitude ;

A mon frère ***Amadou*** *et mes sœurs* ***Safi****,* ***Kadi****,* ***Assiatou*** *et* ***Mariam****, pour tout l'amour et le soutien qu'ils m'ont apporté ;*

A mon cher et tendre ***Monzon Togora****, à qui je ne pourrai que dire merci, pour sa patience et son soutien ;*

A ma « famille de Guelma », qui, tout au long de ces cinq années loin des miens, m'a apporté assistance et réconfort ;

A mon estimée binôme ***Colette****, pour son courage et sa détermination ;*

A toute ma famille ;

A tous mes amis ;

A toutes les personnes qui de près ou de loin m'ont soutenue et encouragée.

Dédicaces

Toute la Gloire et l'honneur à notre Seigneur et Sauveur Jésus-Christ.

C'est avec un énorme plaisir, un cœur ouvert et une immense joie, que je dédie mon travail à mes très chers, respectueux et magnifiques parents **Sogoba Josué** et **Diallo Marcelle**
Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.
Je vous remercie pour tout le soutien et l'amour que vous me portez depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.
Que ce modeste travail soit l'exaucement de vos vœux, le fruit de vos innombrables sacrifices. Puisse Dieu, le très haut vous accorder santé, bonheur et longue vie.

A **Coulibaly Christophe Oumar**
Merci énormément pour ton soutien plus que précieux, merci pour ton grand cœur et toutes tes qualités qui seraient trop longue à énumérer. Ma vie ne serait pas aussi magique sans ta présence et ton amour. Je t'aime de tout mon cœur.

A mes sœurs et frères **Coulibaly Marie Fanta**, **Rosalie Djeneba**, **Apolos Zié** et **Abel Abraham**.

A mes oncles, tantes, beaux-frères et toute ma famille pour leurs conseils et leurs encouragements.

A mes ami(e) s et mes camarades de classe (promotion 2011).

A l'ensemble de la communauté étrangère estudiantine de Guelma.

A tous mes enseignants, pour leur bienveillance et leur contribution.

A mes très chers amis **Adama Dembélé** et **Seydou Doumbia** pour tout le soutien qu'ils mon apporté, merci infiniment du fond du cœur

A mon binôme en particulier avec qui j'ai passé des durs et bons moments pour l'accomplissement de ce projet.

A toutes les personnes
Qui m'ont aidée à franchir un horizon dans ma vie.....

Résumé

Le nombre croissant de données accessibles à travers les réseaux (internet, intranet, etc.) pose le problème de l'intégration de sources de données préexistantes, souvent distantes et hétérogènes, afin de faciliter leur interrogation par un large public. Une des premières approches proposées en intégration de données prône la construction de *médiateurs*. Un médiateur joue le rôle d'interface de requêtes entre un utilisateur et des sources de données. Il donne à l'utilisateur l'impression d'interroger un système homogène et centralisé en lui évitant d'avoir à trouver les sources de données pertinentes pour sa requête, de les interroger une à une, et de combiner lui-même les informations obtenues. L'intégration sémantique des sources permet d'améliorer ce processus par la résolution des conflits sémantiques existants. Les ontologies représentent un élément clé pour modéliser un tel système de médiation sémantique. Cette étude se situe dans le cadre de l'intégration sémantique des données hétérogènes contenues dans un corpus de documents via l'approche médiateur et d'y effectuer des manipulations avancées sur ce système.

Mots-clés : Ontologie, matching, documents hétérogènes, médiation sémantique, intégration de données.

Sommaire

Introduction générale 7

Chapitre I: Recherche sémantique d'information

Introduction 9

1. La recherche d'information 9

1.1. Définitions 9

1.2. Concepts de bases 9

2. Système de recherche d'information (SRI) 10

2.1. Définition 10

2.2. Architecture d'un système de recherche d'information 11

2.3. Le processus de recherche d'information 12

2.3.1. Indexation : 13

2.3.2. Appariement : 15

2.3.3. Résultats : 15

2.4. Modèles de recherche d'information 15

2.4.1. Modèle booléen 16

2.4.2. Modèle vectoriel 16

2.4.3. Modèle probabiliste 16

3. Web sémantique 17

3.1. Définition 17

3.2. Principales composantes du web sémantique 17

3.3. Représentation des connaissances : 18

4. Recherche sémantique d'information 18

4.1. Les ontologies 19

4.1.1. Rôles des ontologies 20

4.1.2. Types d'ontologies 20

4.1.3. Composants d'une ontologie 21

4.1.4. Méthodologie de construction d'une ontologie 22

4.2. Recherche d'information guidée par les ontologies 23

4.3. Langages d'ontologie 23

4.4. Outils d'édition d'ontologie 25

5. Evaluation de SRI 26

5.1. Les mesures d'évaluation d'un SRI 27

5.1.1. Rappel 27

5.1.2. Précision 27

5.1.3. Courbe Rappel/Précision : 27
6. Domaines d'application du web sémantique 28
Conclusion 28
Chapitre II: Intégration sémantique des sources de données hétérogènes
Introduction 29
1. Hétérogénéité des données 29
1.1. Hétérogénéité sémantique 29
1.2. Hétérogénéité structurelle 29
1.3. Hétérogénéité syntaxique 30
1.4. Hétérogénéité de communication 30
2. Intégration de données 30
3. Approches d'intégration de données 31
3.1. Selon le degré de matérialisation 31
3.1.1. Approche de médiation (Virtuelle) 31
3.1.2. Approche d'entrepôt de données (Matérialisée) 32
3.2. Selon le mapping entre schémas 32
3.2.1. Approche GAV 32
3.2.2. Approche LAV 33
3.2.3. Approche hybride 34
3.3. Bilan 34
4. Processus d'intégration de données à base de médiateur 35
4.1. Approche manuelle 35
4.2. Approches Semi-automatiques 35
4.3. Approche automatique 35
5. Architecture d'un système de médiation 35
5.1. Couche de Médiateur 36
5.2. Couche d'Adaptateur 37
5.3. Couche de source de données 37
6. Approches de médiation sémantique à base d'ontologie 37
6.1. Approche à base d'une seule ontologie 38
6.2. Approche à base de multiples ontologies 38
6.3. Approche hybride 39
7. Mesure de similarités entre les concepts d'une ontologie 39
7.1. Approche basée sur les arcs 39
Similarité de « Wu & Palmer » 40

7.2. Approche basée sur les nœuds 40
Similarité de « Resnik 1995 » 40
Similarité de « Lin » 40
7.3. Approche hybride 41
Similarité de « Jiang et Conrath » 41
7.4. Approches basées sur l'espace vectoriel 41
Similarité de « Jaccard » 41
Similarité de « Cosinus » 42
Conclusion 42
Chapitre III: Conception du système de médiation sémantique
Introduction 43
1. Objectifs de notre Système 43
2. Description des différents niveaux du système de médiation 43
2.1. Niveau Médiateur 43
2.2. Niveau Source de données 44
2.3. Niveau Adaptateur 44
3. Conception de l'ontologie globale 45
3.1. Présentation du domaine des problèmes écologiques 45
3.2. Construction de l'ontologie 'ONTEPRO' 47
3.2.1. Spécification des besoins 48
3.2.2. Conceptualisation 48
4. Processus d'intégration sémantique 50
4.1. Formulation des requêtes et reconstruction des résultats 50
4.2. Traitement des requêtes : 52
Conclusion 53
Chapitre IV: Expérimentation et Evaluation des résultats
Introduction 54
1. Les outils logiciels utilisés 54
1.1. NetBeans 54
1.2. Protégé 2000 55
1.3. WordNet 56
1.4. Adobe PDF Illustrator 57
2. Les package utilisés 57
2.1. OWLViz 58
2.2. Jena 58

2.3. WS4J 59
2.4. Lucene 59
3. Les interfaces de l'application 59
3.1. Interface principale 59
3.2. Interface d'intégration de documents hétérogènes 60
3.3. Interface d'interrogation avancée de schéma du médiateur 61
3.4. Interface de manipulation avancée de source de données 62
4. Evaluation de performances 63
5. Comparaison avec les travaux similaires 64
6. Quelques extraits du code source 66
Conclusion 67
Conclusion générale 68
Références 69

Liste des Figures

Figure 1.1: Architecture d'un SRI 11
Figure 1.2: Processus de recherche d'information 13
Figure 1.3: Les opérations effectuées lors de l'indexation 14
Figure 1.4: Types d'ontologie 21
Figure 1.5: Cycle de vie d'une ontologie 22
Figure 1.6: Les quatre ensembles de documents résultats en RI. 26
Figure 1.7: Courbe Rappel/Précision 28
Figure 2.1: Système d'intégration à base de Médiateur 30
Figure 2.2: Approche GAV 33
Figure 2.3: Approche LAV 33
Figure 2.4: Architecture d'un système de médiation 36
Figure 2.5: Simple ontologie 38
Figure 2.6: Ontologie multiple 38
Figure 2.7: Ontologie hybride 39
Figure 3.1: Architecture conceptuelle du système de médiation sémantique 45
Figure 3.2: Domaine de la pollution 47
Figure 3.3: Modèle conceptuel de l'ontologie 'ONTEPRO' 50
Figure 4.1: Interface de NetBeans 55
Figure 4.2: Interface de Protégé 2000 56
Figure 4.3: Interface de WordNet 57
Figure 4.4: Interface de l'Adobe Illustrator 57
Figure 4.5: Interface d'OWLViz 58
Figure 4.6: Interface principale de système 60
Figure 4.7: Fenêtre d'interrogation du système de médiation 60
Figure 4.8: Interface d'interrogation avancée de schéma du médiateur 61
Figure 4.9: Interface de visualisation de l'ontologie 62
Figure 4.10: Interface de mise à jour de la source de données 63

Liste des Tableaux

Tableau 3.1: Types d'hétérogénéités utilisées 44
Tableau 3.2: Classes et hiérarchie des classes 48
Tableau 3.3: Extrait des propriétés des classes 49
Tableau 3.4: Relations IS-A entre les concepts de l'ontologie 49
Tableau 4.1: Evaluation de performances 64
Tableau 4.2: Comparaison avec trois travaux similaires 65

Introduction générale

L'expansion du nombre de sources d'information réparties, hétérogènes et autonomes disponibles sur le Web a rendu indispensable l'intégration de données provenant de différentes sources. Cette évolution quantitative des données et la diversité des formats des sources ont conduit les entreprises à dédier une part importante de leur budget aux technologies de l'information pour l'intégration de données.

Il existe deux grandes approches pour l'intégration de sources d'information.
L'approche *entrepôt de données* consiste à voir cette intégration comme la construction de bases de données réelles, appelées *entrepôts*, regroupant les informations pertinentes pour les applications considérées. L'utilisateur pose alors ses requêtes ou lance un traitement directement sur les données stockées dans l'entrepôt. Les problèmes posés par la construction d'un entrepôt à partir de plusieurs bases de données spécialisées concernent la définition de son schéma, son peuplement et sa mise à jour, en fonction des différentes sources d'information à partir desquelles il est construit. Les travaux récents sur l'intégration de données semi-structurées ou la migration de données d'un format dans un autre relèvent de cette approche.

La deuxième approche dite *médiateur* consiste à fonder l'intégration de données sur l'exploitation de vues abstraites décrivant le contenu des différentes sources d'information. Les données ne sont pas stockées au niveau du médiateur et ne sont accessibles qu'au niveau des sources de données. L'interrogation et la détermination des sources de données pertinentes nécessitent la construction de *plans de requêtes* dont l'exécution permettra d'obtenir l'ensemble des réponses à partir des sources disponibles.

Dans ce mémoire, nous nous focalisons sur l'approche de médiateur qui présente l'intérêt de pouvoir construire un système d'interrogation de sources de données sans toucher aux données qui restent stockées dans leur source d'origine. Un médiateur donne à l'utilisateur l'illusion d'interroger un système homogène et lui évite d'avoir à trouver lui-même les sources de données pertinentes pour sa requête. Le schéma d'une source de données, dit *schéma local*, représente les structures servant de conteneur pour stocker des données dans cette source.

Un utilisateur pose ses requêtes dans les termes du schéma du médiateur. Ce schéma, souvent appelé aussi *schéma global*, regroupe l'ensemble des prédicats modélisant le domaine d'application du système. Il joue donc le rôle de l'ontologie du domaine qui fournit un vocabulaire structuré servant de support à l'expression de requêtes et c'est par son intermédiaire que se fait l'intégration.

La mise en correspondance entre le schéma global et les schémas locaux des sources est appelée *mapping*. Il existe deux principales approches pour définir le mapping : l'approche LAV (*Local As View*) et l'approche GAV (*Global As View*).
Dans l'approche GAV, le schéma global est exprimé à l'aide de vues sur les schémas locaux, contrairement à l'approche LAV où les sources locales sont décrites en fonction du schéma global.

Le médiateur ne peut pas évaluer directement les requêtes qui lui sont posées car il ne contient pas de données. Ces dernières sont en fait stockées de façon distribuée dans des sources indépendantes. Le médiateur ne dispose que de vues abstraites des données stockées dans les

sources. L'intégration effective des sources se fait via des adaptateurs (*wrappers* en anglais) qui traduisent les requêtes exprimées par les vues dans le langage de requêtes spécifique accepté par chaque source.

Les réponses à une requête posée à un médiateur peuvent être calculées par *réécriture en termes de vues* de cette requête. Le problème consiste alors à trouver une requête qui, selon le choix de conception du médiateur est équivalente à la requête de l'utilisateur mais n'utilise que des vues. Cette requête, nommée *réécriture*, est exprimée en termes des vues disponibles dans un langage particulier, appelé *langage de réécritures*. Les réponses à la requête posée sont alors obtenues en évaluant les réécritures de cette requête sur les extensions des vues.

Afin de réduire l'hétérogénéité sémantique de source de données, l'association de l'ontologie dans le médiateur joue un rôle très important. En effet, le système de médiation sémantique permet d'améliorer le processus d'intégration par la résolution des conflits sémantiques existants.

L'objectif de ce travail est de réaliser un système d'intégration sémantique des données hétérogènes en se basant sur l'approche médiateur (intégration virtuelle) et utilisant une ontologie de domaine pour résoudre les conflits sémantiques tout en préservant l'autonomie des sources.

Pour traiter les problèmes d'hétérogénéité, nous utilisons une approche de médiation dans laquelle l'ontologie joue un rôle central. Cette approche consiste à utiliser deux types d'ontologies, une formelle et l'autre linguistique. Pour le premier type, nous construisons une ontologie de domaine afin de définir formellement les concepts d'un domaine particulier ainsi que les relations entre ces concepts. Pour le deuxième type, nous utilisons l'ontologie linguistique Wordnet qui permet de définir le sens des mots et les relations qui existent entre eux.

Plan du manuscrit :

Le présent mémoire est organisé de la façon suivante :

- Le premier chapitre, intitulé « Recherche sémantique d'information » présente les concepts de base de la recherche d'information, puis décrit les concepts qui ont incorporés la sémantique dans la recherche d'information notamment le web sémantique et les ontologies.
- Le deuxième chapitre, « Intégration sémantique de sources de données hétérogènes » dans lequel nous présentons une description générale des systèmes d'intégration ainsi que les différentes méthodes utilisées pour l'intégration des données hétérogènes tout en précisant les différents types d'hétérogénéité.
- Le troisième chapitre « Conception du système de médiation sémantique » est dédié à la conception d'un système de médiation sémantique à base d'ontologie. Le processus d'intégration sémantique y est également expliqué.
- Le quatrième et dernier chapitre intitulé « Expérimentation et Evaluation des résultats » est consacré à la réalisation du système de médiation sémantique conçu dans le chapitre 3.

CHAPITRE 01

RECHERCHE SEMANTIQUE D'INFORMATION

Introduction

La recherche d'information (RI) est apparue comme une discipline de recherche afin d'apporter une solution aux problèmes liés à l'accès aux informations contenues dans des grands volumes de documents. Plusieurs techniques de recherches d'information ont été proposées au fil du temps, parmi lesquelles on peut citer: la classification des documents par catégorie, la création d'indexes, etc.

L'essor de l'internet a vu apparaitre une importante croissance du nombre de ressources d'informations disponibles, ainsi que du nombre d'utilisateurs accédant aux informations de ces ressources. Cependant, les résultats obtenus par ces utilisateurs ne sont pas toujours ceux recherchés; Ce qui fait donc apparaitre l'insuffisance des techniques proposées par la recherche d'information classique. Plusieurs travaux de recherche ont été effectués dans ce domaine, des techniques permettant un accès efficace et rapide aux informations ont été proposées.

Dans ce chapitre, nous présenterons les concepts de base de la recherche d'information. Puis nous détaillerons la recherche sémantique d'information objet de ce chapitre.

1. La recherche d'information

1.1. Définitions

Plusieurs définitions de la recherche d'information ont été proposées, nous en retenons ces quelques-unes:

- *Définition1* : La recherche d'information est une activité dont la finalité est de localiser et de délivrer des granules documentaires à un utilisateur en fonction de son besoin en informations. [36]
- *Définition 2* : La recherche d'information est une branche de l'informatique qui s'intéresse à l'acquisition, l'organisation, le stockage, la recherche et la sélection d'information. [35]

  Définition 3 : La recherche d'information est une discipline de recherche qui intègre des modèles et des techniques dont le but est de faciliter l'accès à l'information pertinente pour un utilisateur ayant un besoin en information. [37]

A partir de ces définitions, nous pouvons déduire que la recherche d'information a pour objectif d'extraire d'un ensemble de documents les informations pertinentes qui reflètent un besoin d'information.

1.2. Concepts de bases

La gestion des informations est réalisée par des outils informatiques appelés Systèmes de Recherche d'Information (SRI). Ces systèmes ont pour but de permettre à l'utilisateur d'accéder à des documents qui contribuent à combler son besoin d'information, exprimé sous forme de requête, qui motive sa recherche. Des définitions ci-dessus, nous pouvons dégager les concepts clés suivants:

- *Document* :

Le mot document vient du latin *documentum*, qui signifie « enseignement », « démonstration ». Au moyen âge, ce terme désigne les textes manuscrits utilisés pour enseigner [2]. Le sens du mot a évolué pour prendre au 18ème siècle le sens d'un écrit servant de preuve, ou de renseignement. Actuellement, l'organisation internationale de normalisation(ISO) définit un document comme un « ensemble formé par un support et une information, généralement enregistrée de façon permanente, et tel qu'il puisse être lu par l'homme et la machine ».

Un document se caractérise par:

- Un contenu : un ensemble d'unités informationnelles juxtaposées dans un ordre bien déterminé, par un émetteur dans un contexte particulier pour communiquer un message à un récepteur.
- Un support : il sert d'intermédiaire physique entre l'utilisateur et l'information contenue dans le document. Il participe au sens donné à l'information. Par exemple, un même contenu n'a pas la même perception s'il est enregistré sur support audio, sur support vidéo ou bien écrit sur un papier.
- Des fonctionnalités : le document comporte deux fonctions principales :
  - ✓ La conservation, le document sert ici comme une trace de l'activité de l'homme.
  - ✓ La communication, le document permet de faire connaitre le point de vue de son auteur ce qui permet la propagation des idées et du savoir.

Dans le contexte de la recherche d'information, on appelle document, toute unité qui peut constituer une réponse à un besoin informationnel de l'utilisateur. Il peut être un texte, un morceau de texte, une page web, une image, une vidéo, etc.

- *Corpus de documents* :

Le corpus de documents est une collection d'informations exploitables et accessibles par un utilisateur, ou simplement, c'est l'ensemble de documents dans lequel l'utilisateur cherche une information.
Généralement le corpus constitue des représentations simplifiées mais bien étudiées des documents, de telle manière que la gestion (ajout, suppression d'un document) et l'interrogation (recherche) soient optimales.

- *Besoin d'information* :

[Web 9] définit le besoin d'information comme «la nécessité ressentie de combler une déficience constatée d'information, une lacune, un défaut ou une anomalie ».
Trois types de besoin d'information ont été définis par [3] :

- Besoin vérificatif : L'utilisateur cherche à vérifier une information avec des données connues et il sait souvent comment y accéder. Le besoin ici est plutôt précis.
- Besoin thématique connu : L'utilisateur cherche à clarifier, ou à compléter des connaissances dans un sujet connu. Le besoin peut être exprimé de façon incomplète.
- Besoin thématique inconnu : L'utilisateur cherche de nouvelles informations dans un sujet qui ne lui est pas familier. Ce besoin est toujours exprimé de façon incomplète.

- *Les requêtes* :

La requête est l'expression du besoin informationnel de l'utilisateur. C'est elle qui initie le processus de recherche. Elle peut être exprimée en langage naturel, booléen, graphique, etc.

2. Système de recherche d'information (SRI)

2.1. Définition

Un système de recherche d'information (SRI) est un système informatique qui sert d'interface entre une collection de documents et des utilisateurs [2]. Il leur permet de retrouver les documents dont le contenu correspond le mieux à leur besoin d'information.
Un SRI inclus un ensemble de procédures et d'opérations qui permettent la gestion, le stockage, l'interrogation, la recherche, la sélection et la représentation de cette masse d'informations.

Pertinence : c'est une notion fondamentale et cruciale dans la recherche d'information mais elle est souvent difficile à appréhender. Plusieurs définitions de cette notion ont été proposées : [Web 10]

- La correspondance entre un document et une requête, une mesure d' « aptitude informative » du document à la requête;
- Un degré de relation (chevauchement, relativité,…) entre le document et la requête ;
- Un degré de la surprise qu'apporte un document, qui a un rapport avec le besoin de l'utilisateur ;
- Une mesure d'utilité du document pour l'utilisateur ;

Deux niveaux de pertinence ont été définis :

- *La pertinence utilisateur* [34]:

Elle est subjective et dépend de la perception de l'utilisateur sur l'information fournie par le système en réponse à sa requête. Deux utilisateurs peuvent avoir des jugements différents sur un même document sélectionné pour une même requête. Elle peut évoluer dans le temps d'une recherche au fur et à mesure que l'utilisateur acquiert plus de connaissances sur le sujet recherché.

- *La pertinence système* [34]:

Elle est déterministe et objective. Elle se calcule par une fonction de pertinence qui varie selon le modèle de recherche considéré.

2.2. Architecture d'un système de recherche d'information

En général, l'architecture d'un système de recherche d'information intègre les fonctionnalités suivantes [2]:

- La représentation des modules de documents et de requêtes (indexation) ;
- Le module de recherche d'information;
- La reformulation de requêtes

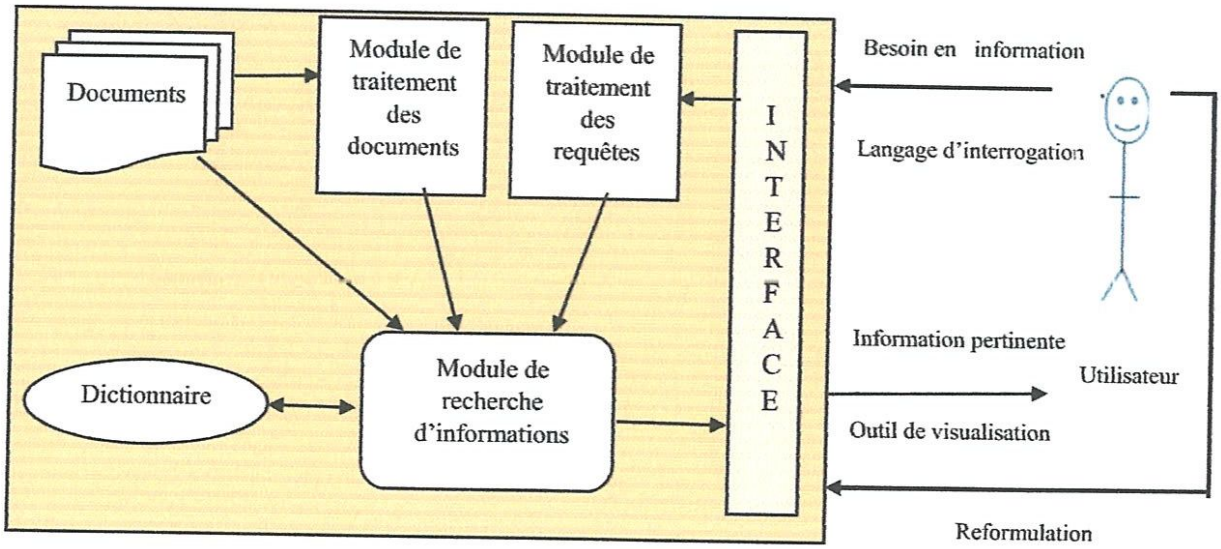

Figure 1.1: *Architecture d'un SRI*

- *La représentation* :

Pour faire la correspondance entre un document et une requête, il faut comparer les termes qui apparaissent dans ces deux éléments. Cependant, les collections comportent souvent un grand nombre de documents de tailles élevées ce qui implique un processus de comparaison lent et complexe. Afin de simplifier ce processus de comparaison, le document doit subir une transformation pour en faciliter la manipulation. Des mots « représentants » ou mots

véhiculant le contenu sémantique de documents et de requêtes y sont donc extraits et organisés. Cette transformation est appelée indexation.

- *La recherche* :

Dans ce processus, le système « détermine » les documents correspondants à la requête de l'utilisateur. Cette décision est basée sur un ensemble de fonctions mathématiques afin de pouvoir évaluer certaines relations, notamment la relation d'appariement entre la requête et les documents. Dans le processus d'appariement documents-requêtes, le système calcule un score de correspondance entre la représentation de chaque document et celle de la requête. Ce score peut être binaire (pertinent ou non) ou multivalué pour exprimer un degré de pertinence système, c'est-à-dire, le jugement que le système avait porté sur la pertinence du document vis-à-vis de la requête. La pertinence système est calculée à partir d'une fonction de similarité appelée RSV (Q, D) (Retrieval Status Value) où Q est une requête et D un document. Pour une requête donnée, le système retourne des documents en ordre décroissant du score de pertinence.

- *Reformulation de requêtes* :

L'objectif de cette étape est de rapprocher la pertinence système de la pertinence utilisateur. Il s'agit de modifier la requête de l'utilisateur en rajoutant des termes qui peuvent provenir de différentes sources. Toujours dans le but d'avoir un SRI plus performant, une ressource externe (ontologie, hiérarchie de concepts, vocabulaire contrôlé d'un theasaurus) peut être utilisée durant les processus d'indexation, de recherche et de reformulation de requêtes.

2.3. Le processus de recherche d'information

Un système de recherche d'information manipule un corpus de documents qu'il transpose à l'aide d'une fonction d'indexation en un corpus indexé. Ce corpus lui permet de résoudre des requêtes traduites à partir de besoins utilisateur. Un tel système repose sur la définition d'un modèle de recherche d'information qui effectue ces deux transpositions et qui fait correspondre les documents aux requêtes. La transposition d'un document en un document indexé repose sur un modèle de document. De même, la transformation du besoin utilisateur en requête repose sur un modèle de requête. Enfin, la correspondance entre une requête et des documents s'établit par une relation de pertinence [4]. La figure suivante présente les différentes étapes d'un processus de recherche d'information.

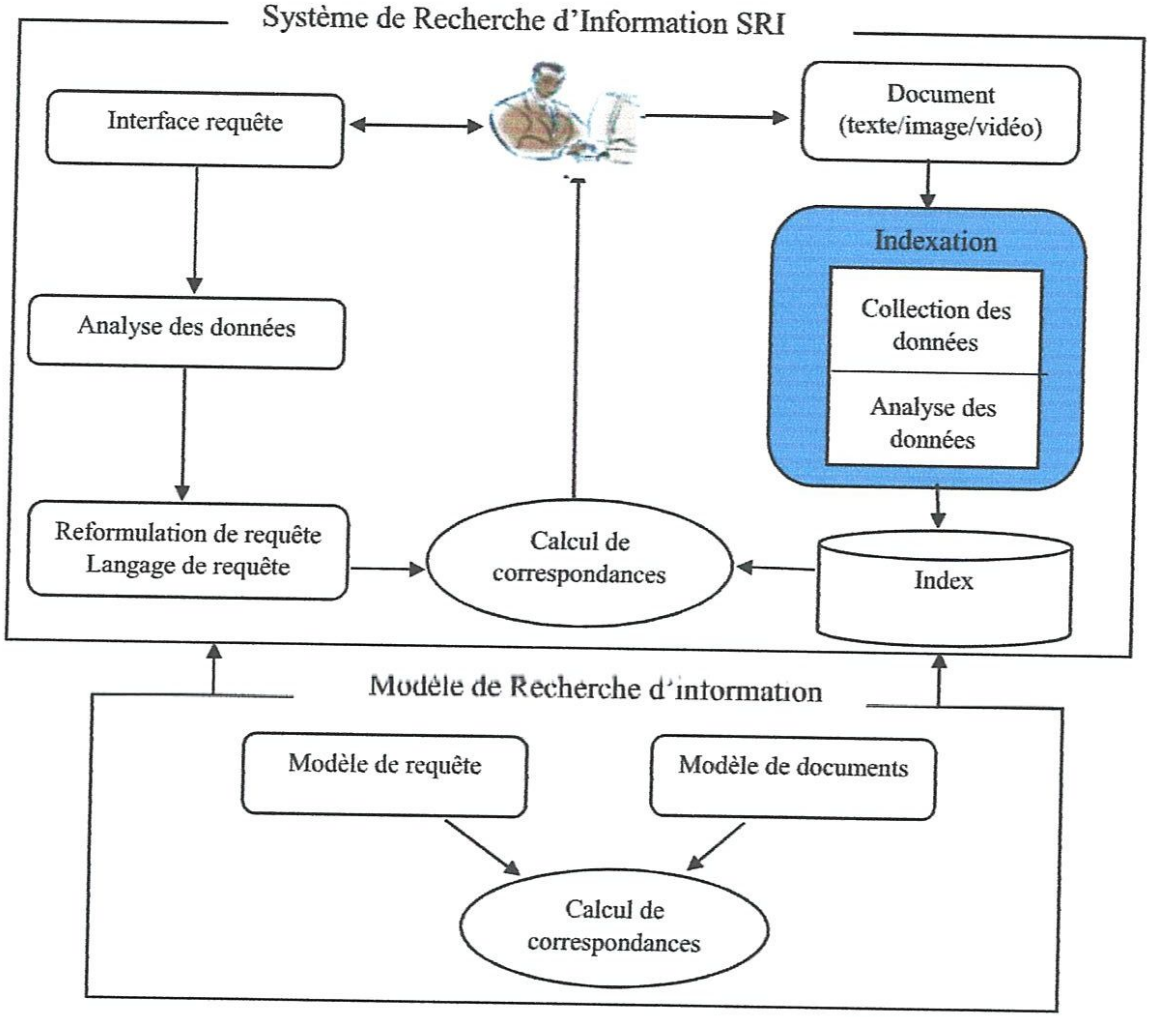

Figure 1.2: *Processus de recherche d'information*

2.3.1. Indexation :

L'indexation consiste à extraire des documents les mots les plus discriminants encore appelés index [38]. Cette première tâche est généralement effectuée en marge du processus de recherche car, la construction des index peut être assez longue en fonction du nombre de documents de la collection ainsi que de la taille des documents. Les index ont un caractère réducteur car tous les termes d'un document ne sont pas importants à prendre en compte pour la recherche. L'indexation peut se faire de 3 manières différentes [1]:

a) Indexation manuelle : chaque document est analysé par un spécialiste du domaine correspondant ou par un documentaliste,
b) Indexation automatique : chaque document est analysé à l'aide d'un processus entièrement automatisé,
c) Indexation semi-automatique : le choix final reste au spécialiste du domaine correspondant ou documentaliste, qui intervient souvent pour établir des relations sémantiques entre mots-clés et choisir les termes significatifs.

Bien que l'indexation se base sur des techniques relativement établies, il peut y avoir plusieurs indexations différentes d'un même texte, aussi valables les unes que les autres, en fonction de l'usage qui doit en être fait et du public auquel elles s'adressent.

L'indexation se décompose en trois phases schématisées dans la figure 1.3. - L'extraction des termes du document. - La sélection des termes discriminatifs pour un document. - La pondération des termes.
Dans le cas d'un document textuel, quatre opérations effectuées lors de l'indexation [39]:

1. *Analyse lexicale :* transformer un document textuel en un ensemble de termes (lexème). Pendant cette opération, la ponctuation, la casse, et la mise en page sont supprimées.

2. *Sélection :* consiste à garder que les termes discriminatifs pour un document, en utilisant par exemple un anti-dictionnaire (tel que du système SMART) qui permet de ne pas conserver les mots qui ne reflétant pas le contenu informationnel des documents, à savoir : les articles, les pronoms, les prépositions,…etc.

3. *Radicalisation :* consiste à éliminer les différences non significatives des mots (préfixes et suffixes) et de garder la partie commune appelée le radical (ou la racine).

4. *Pondération :* consiste à attribuer une valeur numérique appelée poids à un terme d'indexation, de manière à représenter son pouvoir de discrimination pour chaque document de la collection.

La figure suivante illustre ces différentes opérations qui se composent l'indexation.

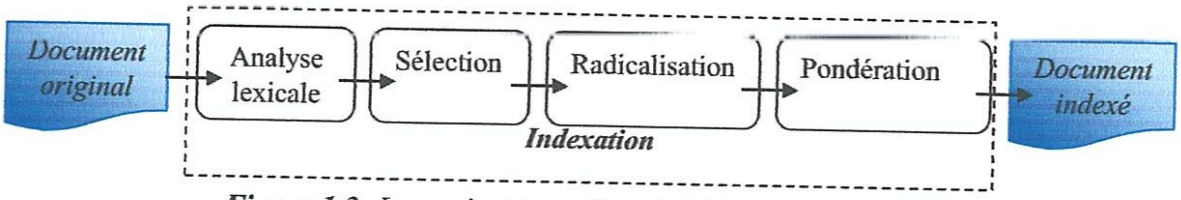

Figure 1.3: *Les opérations effectuées lors de l'indexation*

Nous détaillons dans la suite l'opération la plus importante qui est la pondération des termes où nous présentons les techniques de pondération les plus utilisées.

Plusieurs techniques de pondération ont vu le jour, la plupart d'entre elles sont basées sur les facteurs « tf» et « idf», qui permettent de combiner les pondérations locale et globale d'un terme :

- *tf (term-frequency) :* est la fréquence du terme dans le document (pondération locale) c'est-à-dire le nombre d'occurrences d'un terme dans le document. Il peut être utilisé tel quel ou selon plusieurs déclinaisons (log (tf), présence/absence,. . .).

- *idf (Inverse of Document Frequency) :* ce facteur mesure l'importance d'un terme dans toute la collection (pondération globale). Un terme qui apparaît souvent dans la base documentaire ne doit pas avoir le même impact qu'un terme moins fréquent. Il est généralement exprimé comme suit: log (N/df), où df est le nombre de documents contenant le terme et N est le nombre total de documents de la base documentaire.

Le poids d'un terme j dans le document i s'écrit alors généralement [40]:

$$Poids_i(j) = tf_{ij} \times idf_j$$

La mesure *«tf×idf»* donne une bonne approximation de l'importance du terme dans le document, particulièrement dans les corpus de documents de taille homogène. Cependant, elle ne tient pas compte d'un aspect important du document : sa longueur. En général, les documents les plus longs ont tendance à utiliser les mêmes termes de façon répétée, ou à utiliser plus de termes pour décrire un sujet. Par conséquent, les fréquences des termes dans

les documents seront plus élevées, et les similarités à la requête seront également plus grandes. Pour cette raison nous avons utilisé la formule normalisée suivante [41] :

$$TF=\frac{\sum_{di} tf(termej)}{N}$$

2.3.2. Appariement :

Les SRI intègrent un processus de recherche/décision qui permet de sélectionner l'information jugée pertinente pour l'utilisateur. A cet effet, une mesure de similitude (correspondance) entre la requête indexée et les descripteurs des documents de la collection est calculée. Seuls les documents dont la similitude dépasse un seuil prédéfini sont sélectionnés par le SRI.
La fonction de correspondance est un élément clé d'un SRI, car la qualité des résultats dépend de l'aptitude du système à calculer une pertinence des documents la plus proche possible du jugement de pertinence de l'utilisateur.
Il existe deux types d'appariement :

- ✓ *Appariement exact :*

Le résultat est une liste de documents respectant exactement la requête spécifiée avec des critères précis. Les documents retournés ne sont pas triés.

- ✓ *Appariement approché :*

Le résultat est une liste de documents censés être pertinents pour la requête. Les documents retournés sont triés selon un ordre de mesure. Cet ordre reflète le degré de pertinence document/requête.

2.3.3. Résultats :

Il s'agit de l'expression du besoin d'information de l'utilisateur dans la forme imposée par le système, la recherche dans le corpus, et la présentation des résultats. Cette phase nécessite un modèle de représentation du besoin de l'utilisateur, appelé modèle de requêtes, ainsi qu'une fonction de correspondance qui doit évaluer la pertinence des documents par rapport à la requête. La réponse du système est un ensemble de références à des documents qui obtiennent une valeur de correspondance élevée. Cet ensemble est généralement présenté sous la forme d'une liste ordonnée suivant la valeur de correspondance [11].

La satisfaction de l'utilisateur concernant les documents retournés par le système, n'est pas toujours acquise. Certains systèmes permettent aux utilisateurs de marquer parmi les documents résultats ceux qu'ils jugent pertinents ou non pertinents. Ces jugements sont alors pris en compte pour définir une nouvelle requête, il s'agit du processus de reformulation. Ce processus n'est pas toujours automatique, une stratégie classique d'utilisation des systèmes de recherche d'information consiste à reformuler manuellement la requête en tenant compte des documents pertinents et non pertinents obtenus [6].

2.4. Modèles de recherche d'information

Un modèle de RI a pour rôle de fournir une formalisation du processus de RI et un cadre théorique pour la modélisation de la mesure de pertinence [12]. Il existe un grand nombre de modèles de RI textuelle développé dans la littérature. Ces modèles ont en commun le vocabulaire d'indexation basé sur le formalisme mots clés et diffèrent principalement par le modèle d'appariement requête-document. Le vocabulaire d'indexation $V = \{ti\}, i \in \{1, \ldots, n\}$ est constitué de n mots ou racines de mots qui apparaissent dans les documents. Selon [12], un modèle de RI est défini par un quadruplet (D, Q, F, R (q,d)) : où

– D est l'ensemble de documents
– Q est l'ensemble de requêtes

– F est le schéma du modèle théorique de représentation des documents et des requêtes
– R (q, d) est la fonction de pertinence du document d à la requête q
Nous présentons dans la suite les principaux modèles de RI : le modèle booléen, le modèle vectoriel et le modèle probabiliste.

2.4.1. Modèle booléen

Le modèle booléen [13] est basé sur la théorie des ensembles. Dans ce modèle, les documents et les requêtes sont représentés par des ensembles de mots clés. Chaque document est représenté par une conjonction logique des termes non pondérés qui constitue l'index du document. Un exemple de représentation d'un document est comme suit : d = t1∧t2∧t3...∧tn. Une requête est une expression booléenne dont les termes sont reliés par des opérateurs logiques (OR, AND, NOT) permettant d'effectuer des opérations d'union, d'intersection et de différence entre les ensembles de résultats associés à chaque terme. Un exemple de représentation d'une requête est comme suit : q = (t1∧t2)∨(t3∧t4). La fonction de correspondance est basée sur l'hypothèse de présence/absence des termes de la requête dans le document et vérifie si l'index de chaque document *dj* implique l'expression logique de la requête *q*. Le résultat de cette fonction est donc binaire est décrit comme suit : RSV (q, d) = {1,0}.

2.4.2. Modèle vectoriel

Le modèle vectoriel proposé par Salton dans le système SMART [13], est un modèle algébrique où l'on représente les documents et les requêtes par des vecteurs dans un espace multidimensionnel dont les dimensions sont les termes issus de l'indexation.
L'index d'un document *dj* est le vecteur = (w1, j ,w2,j ,w3,j , ..,wn, j), où wk, j ∈ [0, 1] dénote le poids du terme *tk* dans le document *dj*. Une requête est également représentée par un vecteur = (w1, q, w2, q, w3, q, .., wn, q), où *wk, q* est le poids du terme *tk* dans la requête q.
La fonction de correspondance mesure la similarité entre le vecteur requête et les vecteurs documents. Une mesure classique utilisée dans le modèle vectoriel est le cosinus de l'angle formé par les deux vecteurs : RSV (q, d) = cos (q, d).
Plus les vecteurs sont similaires, plus l'angle formé est petit, et plus le cosinus de cet angle est grand. A l'inverse du modèle booléen, la fonction de correspondance évalue une correspondance partielle entre un document et une requête, ce qui permet de retrouver des documents qui ne reflètent pas la requête qu'approximativement. Les résultats peuvent donc être ordonnés par ordre de pertinence décroissante.
D'une manière générale, les résultats de recherche tendent à prouver que les systèmes de recherche vectoriels sont plus performants en terme de précision que les systèmes de recherche booléens.

2.4.3. Modèle probabiliste

Ce modèle est fondé sur le calcul de la probabilité de pertinence d'un document pour une requête [15], [14]. Le principe de base consiste à retrouver des documents qui ont en même temps une forte probabilité d'être pertinents, et une faible probabilité d'être non pertinents [16]. Etant donné une requête utilisateur *Q* et un document *D*, il s'agit de calculer la probabilité de pertinence du document pour cette requête. Deux possibilités se présentent : *R*, *D* est pertinent pour *q* et D, n'est pas pertinent pour q. Les documents et les requêtes sont représentés par des vecteurs booléens dans un espace à n dimensions. Un exemple de représentation d'un document dj et une requête q est le suivant :
dj = (w1,j ,w2,j ,w3,j , ..,wn,j), q = (w1,q,w2,q,w3,q, ..,wn,q). Avec wk,j ∈ [0, 1] et wk,q ∈ [0 1]. La valeur de wk,j (resp. wk,q) indique si le terme tk apparaît ou non dans le document *dj* (resp. q).

Le modèle probabiliste évalue la pertinence du document dj pour la requête q. Un document est sélectionné si la probabilité que le document d soit pertinent, notée *p(R/D)*, est supérieure à la probabilité que d soit non pertinent pour *q*, notée *p(R/D)* où *R* est l'événement de pertinence et D est l'événement de non pertinence. Le score d'appariement entre le document D et la requête *Q*, noté RSV (Q, D) est donné par :

$$\text{RSV (D, Q)} = \frac{\text{P(PERT|D,Q)}}{\text{P(NPERT|D,Q)}}$$

Ces probabilités sont estimées par de probabilités conditionnelles selon qu'un terme de la requête est présent, dans un document pertinent ou dans un document non pertinent. Cette mesure de similarité entre la requête et les documents peut se calculer par différentes formules. Ce modèle a donné lieu à de nombreuses extensions. Il est à l'origine du système OKAPI. Le modèle Okapi BM25 a été développé par Robertson en 1994 dans lequel le calcul du poids d'un terme dans un document intègre des aspects relatifs à la fréquence locale des termes, leur rareté et la longueur des documents.

Dans notre étude nous devons choisir le modèle de recherche le plus utilisé et le plus adapté aux documents issue des sources hétérogènes.

3. Web sémantique

Depuis sa création, le web a connu un succès considérable et il est en train de devenir peu à peu le premier outil pour la production, la publication, la diffusion et le partage de l'information. Cependant la répartition à travers le monde d'un tel réseau d'informations, la croissance accrue du nombre de publications et la liberté totale d'y accéder ont révélé plusieurs limites et inconvénients. En effet, le web ne dispose pas d'outils pour décrire et structurer ses ressources de manière satisfaisante afin de permettre un accès pertinent à l'information. Par exemple, les liens entre les pages web, bien que porteurs de sens pour les utilisateurs, n'ont aucune signification exploitable par les machines. C'est pour pallier ces insuffisances que Tim Berners Lee a proposé dans [17] d'étendre le web vers un web où l'information possédera un sens bien défini permettant ainsi aux applications d'exploiter directement la sémantique des ressources et de coopérer avec l'utilisateur afin de lui faciliter ses tâches.

3.1. Définition

Le web sémantique est une infrastructure qui permet l'utilisation de connaissances formalisées en plus du contenu informel que l'on peut trouver dans le web [5]. Cette infrastructure s'appuie sur un certain niveau de consensus portant, par exemple, sur les langages de représentation ou sur les bases de connaissance utilisées. Ainsi, elle permet, le plus automatiquement possible, l'interopérabilité et les transformations entre les différents formalismes. Grâce à la formalisation de connaissances, cette infrastructure peut faciliter la mise en œuvre de calculs et de raisonnements complexes tout en offrant des garanties supérieures sur leur validité. Mais restreindre le web sémantique à cette infrastructure serait trop limitatif. Sur la base de sémantiques bien définies pour ses ressources, le web sémantique pourra fournir aux utilisateurs, par le moyen d'agents logiciels, des services automatiques et avancés [5]. Comme l'écrivent en substance « le web sémantique est une extension du web courant, dans laquelle on donne à une information un sens bien défini pour permettre aux ordinateurs et aux personnes de travailler en coopération ».

3.2. Principales composantes du web sémantique

Le Web Sémantique a été proposé en se basant sur les critiques adressées au web. Ces critiques s'articulent autour des éléments suivants [7]:

– Certes HTML a permis de tisser tout un réseau d'informations par ses liens hypertextes, mais il n'a donné aucune sémantique à ces liens ce qui les rend pratiquement inexploitables par les machines,
– Les métadonnées utilisées sont non structurées et limitées dans leurs usages.
– Il est difficile de faire des inférences et des raisonnements sur les connaissances décrites dans les documents publiés sur le web vu l'absence de modèles permettant la représentation sémantique de ces connaissances [7].
Tous ces problèmes ont fait l'objet de différents travaux de recherche qui ont convergé vers plusieurs solutions parmi lesquelles celles qui semblent les plus essentielles :
– Proposer des langages et des formalismes de représentation et de structuration des connaissances (représenter le contenu sémantique des ressources du web).
– Rendre disponibles des ressources conceptuelles (des modèles) représentées dans ces langages modélisant les connaissances et facilitant leur accès et leur partage : les ontologies.
– Proposer des métadonnées explicites, c'est-à-dire qui suivent un modèle et qui sont exprimées dans des langages définis formellement.

3.3. Représentation des connaissances :

Le fonctionnement du Web Sémantique est fondé sur la possibilité d'accès des machines à l'ensemble des informations éparpillées sur le web. Le W3C (World Wide Web Consortium) ainsi que les chercheurs travaillant dans le domaine de l'intelligence artificielle ont beaucoup travaillé sur ce point et ont proposé plusieurs langages de représentation des connaissances afin de faciliter cet accès. XML (eXtensible Markup Language), RDF (Ressource Description Framework), RDFS (Ressource Description Framework Schema), DAML+OIL, OWL (Web Ontology Language) présentent un sous ensemble des langages disponibles (les plus utilisés) pour la représentation des connaissances dans le cadre du Web Sémantique. Ces langages, offrent un degré important d'expressivité pour représenter les connaissances simples et complexes (sous forme d'ontologie ou de métadonnées) décrites dans les ressources du Web [6].

Une annotation (ou méta-données) est une information descriptive facilitant l'accès, la recherche et l'utilisation d'une ressource. En termes de documentation, les annotations sémantiques décrivent le lien entre les entités se trouvant dans le document et leurs descriptions sémantiques représentées dans une ressource [6]. Elles permettent ainsi de désambiguïser le contenu du document pour un traitement automatique (ex. recherche documentaire, résumé…). Avec l'expansion du Web Sémantique, plusieurs outils ont été proposés pour permettre la génération automatique ou manuelle de ces annotations sémantiques. Par exemple. Annotea, MnM, KIM, S-CREAM. Beaucoup d'autres systèmes ont été proposés pour la génération (semi-)automatique d'annotations sémantiques [7]. Ils proposent, le plus souvent, des annotations en RDF (devenu un standard pour la représentation des méta-données). Par ailleurs, la majorité de ces systèmes s'intéressent surtout à la tâche d'instanciation de concepts et proposent des techniques d'instanciation de formes simples de relations pouvant exister entre ces concepts (synonymie, hypermimie) [6].

4. Recherche sémantique d'information

L'intérêt d'utiliser des ressources sémantiques en recherche d'information est de pouvoir retourner, lors d'une recherche, les documents qui partagent avec la requête le maximum de concepts (sens des mots) plutôt que le maximum de mots-clés. Les ressources sémantiques (thésaurus, ontologies, etc.) ont un apport considérable pour le traitement des documents. Leur utilisation en RI peut intervenir lors de la phase de recherche ou lors de la phase d'indexation [42].

La phase d'indexation consiste à construire au préalable une structure d'accès aux documents qui facilitera la phase de recherche. Plus la phase d'indexation est sophistiquée, plus la phase de recherche sera facile. Les documents peuvent être indexés par un groupe de concepts, où on sait qu'un tel document traite des concepts A et B mais où on ne connaît pas les relations entre eux dans le document. Pour traiter les relations entre les concepts d'un document, une solution consiste à attribuer à chaque document une description sémantique où les concepts sont représentés avec leurs relations sémantiques [43]. Cette représentation confère un grand pouvoir d'expression mais elle peut ralentir les traitements et la construction des descriptions sémantiques associées à chaque document.

La phase de recherche consiste à retrouver les documents les plus pertinents par rapport à une requête donnée. Pour intégrer la sémantique en RI, plusieurs langages de représentation sémantique d'information ont été défini, le plus utilisé actuellement est l'*ontologie*.

La recherche sémantique de l'information est une des principales motivations du Web sémantique. Un système de recherche sémantique peut être vu comme un outil qui répond à des requêtes formulées avec les concepts et les relations entre eux en les alignant avec des annotations sémantiques des documents cibles.

4.1. Les ontologies

La notion d'ontologie intéresse à la fois l'ingénierie des connaissances, la linguistique et la philosophie. Initialement, l'ontologie était un domaine de la philosophie concernant « l'étude de l'être en tant qu'être, c'est-à-dire l'étude des propriétés générales de ce qui existe ». Or, progressivement les objets produits par cette discipline ont pris le nom d'ontologies [8].

Cette notion a été reprise par les chercheurs dans le domaine de l'intelligence artificielle et utilisée dans le cadre de construction des systèmes à base de connaissances. L'idée était de séparer, d'un côté, la modélisation des connaissances d'un domaine, et d'un autre côté, l'utilisation de ces connaissances (i.e. le raisonnement).
Dans ce contexte, plusieurs définitions des ontologies ont été proposées. La première a été proposée par [9]: « Une ontologie définit les termes et les relations de base du vocabulaire d'un domaine ainsi que les règles qui permettent de combiner les termes et les relations afin de pouvoir étendre le vocabulaire ».
Cette définition descriptive donne un premier aperçu sur la manière de construire une ontologie, à savoir l'identification des termes et des relations d'un domaine ainsi que les règles pouvant s'appliquer sur ces derniers.
Deux années plus tard, [10] donne la définition qui est devenue la plus utilisée dans la littérature : «Une ontologie est une spécification explicite d'une conceptualisation».
La conceptualisation se réfère ici à l'élaboration d'un modèle abstrait d'un domaine du monde réel en identifiant et en classant les concepts pertinents décrivant ce domaine. La formalisation consiste à rendre cette conceptualisation exploitable par des machines.
Dans cette même logique [18] proposent leur définition : « Une ontologie est une théorie logique proposant une vue explicite et partielle d'une conceptualisation ».
Depuis, de nombreuses définitions, à la fois complémentaires et précises, ont vu le jour. [19] soulignent la dépendance entre la formalisation de l'ontologie et l'application dans laquelle elle va être utilisée : «Une ontologie organise dans un réseau des concepts représentant un domaine. Son contenu et son degré de formalisation sont choisis en fonction d'une application».
Donc une ontologie traduit un consensus explicite sur la formalisation des connaissances d'un domaine pour faciliter le partage et la réutilisation de ces connaissances par les membres d'une communauté ou par des agents logiciels. C'est dans cette optique que les ontologies se présentent comme un pilier du web sémantique, car elles permettent de faire communiquer les

hommes et les machines en utilisant la sémantique partagée par les différents acteurs du web et en décrivant ses ressources [6].

4.1.1. Rôles des ontologies

Historiquement, la notion d'ontologie est apparue pour satisfaire des besoins d'interopérabilité dans les systèmes informatiques et de réutilisation. On attend d'elles qu'elles améliorent la communication non seulement entre machines, mais aussi entre humains et machines ou encore entre humains par le biais de logiciels. Les propriétés de ce type de structure de données ont permis de diversifier leur utilisation à différentes applications, en particulier la gestion des connaissances et le Web sémantique. Elles sont utilisées pour : [8]

- Résoudre des problèmes de compréhension et faciliter le partage des connaissances entre personnes de spécialités différentes
- Assurer l'interopérabilité entre applications à base de connaissances ;
- Accéder à des ressources hétérogènes ;
- Permettre la réutilisation de modèles de connaissances;
- Faciliter la communication entre agents logiciels ;
- Annoter des ressources à l'aide de méta-données ;
- Améliorer les processus de recherche d'informations.

4.1.2. Types d'ontologies

[20] définissent deux grandes typologies d'ontologies : une typologie fondée sur la structure de la conceptualisation et l'autre fondée sur le sujet de la conceptualisation.
Dans la première typologie, ils distinguent trois catégories à savoir

- Les ontologies terminologiques (lexiques, glossaires…);
- Les ontologies d'information (schéma d'une BD);
- Les ontologies des modèles de connaissances.

Dans la deuxième typologie, qui est la plus citée, ils distinguent quatre catégories :

- *Les ontologies d'application* : elles contiennent toutes les informations nécessaires pour modéliser les connaissances pour une application particulière.
- *Les ontologies de domaine :* elles fournissent un ensemble de concepts et de relations décrivant les connaissances d'un domaine spécifique.
- *Les ontologies génériques* (dites aussi de haut niveau) : elles sont similaires aux ontologies de domaine, mais les concepts qui y sont définis sont plus génériques et décrivent des connaissances tels que l'état, l'action, l'espace et les composants. Généralement, les concepts d'une ontologie de domaine sont des spécialisations des concepts d'une ontologie de haut niveau.
- *Les ontologies de représentation* (méta-ontologies) : elles fournissent des primitives de formalisation pour la représentation des connaissances. Elles sont généralement utilisées pour écrire les ontologies de domaine et les ontologies de haut niveau. Exemples : Frame Ontology [10] et RDF Schema Ontology [Web 7].

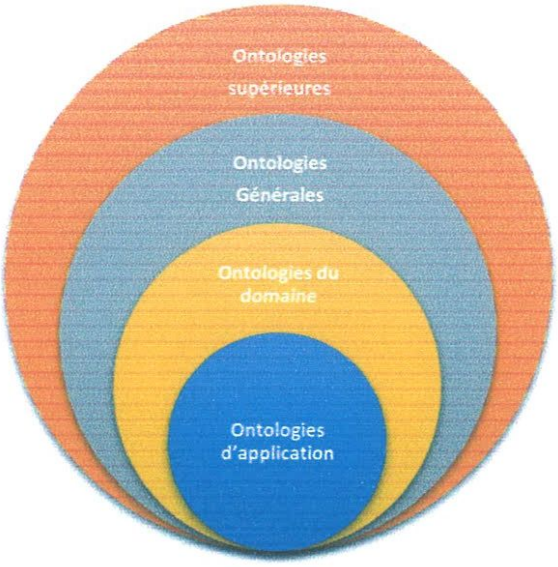

Figure 1.4: *Types d'ontologie*

4.1.3. Composants d'une ontologie

Les connaissances intégrées dans les ontologies sont formalisées en mettant en jeu cinq types de composants: les concepts, les relations, les fonctions, les axiomes et les instances [10].

- Les concepts, aussi appelés termes ou classes de l'ontologie, correspondent à tout ce qui peut être évoqué comme la description d'une tâche, d'une fonction, d'une action, d'une stratégie ou d'un processus de raisonnement, etc. Selon [21] ces concepts peuvent être classifiés selon plusieurs dimensions :
  - *niveau d'abstraction :* concret ou abstrait ;
  - *atomicité :* élémentaire (électron) ou composée (atome) ;
  - *niveau de* réalité : réel ou fictif.
- Les relations traduisent les interactions existant entre les concepts présents dans le domaine ciblé. Ces relations sont formellement définies comme tout sous-ensemble d'un produit cartésien de n ensembles, c'est à dire R: $C_1 \times C_2 \times \ldots \times C_n$. Les relations peuvent être de différents types: *sous-classe-de* (généralisation/spécialisation) ; Partie-de (agrégation ou composition) ; Associée-à ; Instance de, etc.
- Les fonctions sont des cas particuliers de relations dans lesquelles le nième élément (extrant) de la relation est défini de manière unique à partir des n-1 premiers (intrants). Formellement, les fonctions sont définies ainsi : F: $C_1 \times C_2 \times \ldots \times C_{n-1} \times C_n$. Comme exemple de fonctions binaires, nous avons la fonction *mère de*, et comme exemple de fonction ternaire, *Prix-voiture-occasion*, qui calcule le prix d'une voiture d'occasion selon le modèle de la voiture, la date de construction et le nombre de kilomètres.
- Les axiomes constituent des assertions, acceptées comme vraies, à propos des abstractions du domaine traduites par l'ontologie. Ils peuvent intervenir dans la définition des concepts et des relations, dans la vérification de l'exactitude d'informations indiquées dans l'ontologie ou dans la déduction d'une nouvelle information.
- Les instances aussi appelés individus sont des entités constituant la définition extensionnelle de l'ontologie.

4.1.4. Méthodologie de construction d'une ontologie

Les méthodologies peuvent porter sur l'ensemble du processus et guider l'ontologiste dans toutes les étapes de la construction. Bien qu'aucune méthodologie générale n'ait pour l'instant réussi à s'imposer, de nombreux critères de construction d'ontologies ont été proposés: ENTERPRISE, TOVE et METHONTOLOGY étant les méthodologies les plus représentatives.

Nous allons développer dans ce qui suit, la méthodologie dite « METHONTOLOGY » qui se situe entre le GL (Génie Logiciel) et l'IC (Ingénierie des Connaissances). Elle présente un certain nombre de phases spécifiées de manière très détaillée, notamment la phase de conceptualisation. De ce fait, nous allons adopter cette méthodologie pour notre travail.

L'approche « METHONTOLOGY » distingue les étapes suivantes [24]:

- Spécification : cette étape a pour but de fournir une description claire du problème étudié ainsi que la façon de le résoudre. Elle permet de préciser l'objectif, la portée et le degré de granularité de l'ontologie qui sera construite.
- Acquisition des connaissances : Se fait par l'expert du domaine, textes d'articles (text mining), méta-données de base de données etc. Dresser une liste de questions de compétences.
- Conceptualisation : Elle consiste à identifier et à structurer les connaissances du domaine, à partir des sources d'informations. L'acquisition de ces connaissances peut s'appuyer à la fois sur l'analyse de documents et sur l'interview des experts du domaine. Une fois que les concepts sont identifiés par leurs termes, leur sémantique est décrite dans un langage semi-formel (tables et graphes) à travers leurs propriétés, leurs instances connues et les relations qui les lient entre eux.
- Intégration : utiliser ou spécialiser une ontologie existante
- Formalisation : cette étape consiste en la transcription du modèle conceptuel de l'ontologie dans un langage formel de représentation de connaissances.
- Documentation : Produire des définitions formelles, informelles, complètes, pour préciser la signification des termes de l'ontologie ; donner des exemples.
- Implémentation : Cette étape consiste à formaliser le modèle conceptuel obtenu dans l'étape précédente par un formalisme de représentation d'ontologie telles que les logiques de description. Puis, à coder l'ontologie dans un langage opérationnel du Web Sémantique.
- Evaluation : Déterminer l'adéquation de l'ontologie pour l'application visée ; évaluation à faire de façon pragmatique.
- Maintenance : cela peut s'agir d'une maintenance corrective ou évolutive de l'ontologie (nouveaux besoins de l'utilisateur), ce qui permet la validation et l'évolution de celle-ci. Cette activité est généralement faite par le constructeur et des experts du domaine. La validation se base sur l'exploitation des services d'inférences associés aux logiques de description, et qui sont offerts par des raisonneurs.

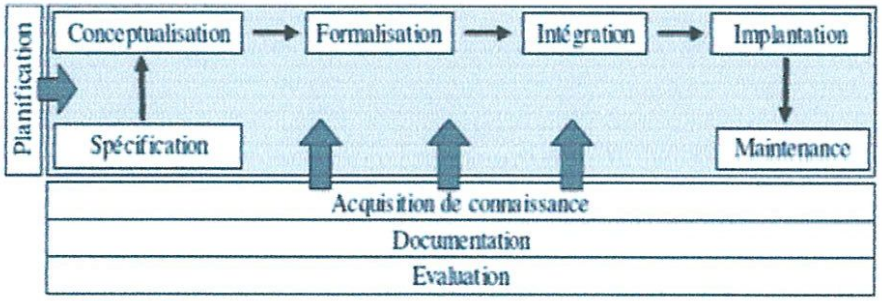

Figure 1.5: *Cycle de vie d'une ontologie [Web 2]*

4.2. Recherche d'information guidée par les ontologies

L'utilisation typique du Web actuel consiste en la recherche d'information qui peut être d'ordre professionnel (veille stratégique/technologique, recherche d'articles…) ou d'ordre personnel (recherche de personnes ou de produits). Pour faciliter ces tâches, plusieurs moteurs de recherche ont vu le jour (Google, Yahoo, Altavista…). Ces outils, bien qu'ils répondent à une bonne partie des besoins des utilisateurs, présentent quelques problèmes critiques :

– La masse énorme des documents retournés,
– La sensibilité au vocabulaire utilisé dans la requête,
– Le résultat fractionné en pages Web,
– La variabilité des langages utilisés sur le web et le non structuration des documents, ce qui rend cette tâche de plus en plus laborieuse.

La réflexion sur le web sémantique a été essentiellement fondée sur ce problème de la recherche d'informations. En effet, les ontologies peuvent améliorer la pertinence d'une recherche et ce, en recherchant des documents faisant référence à un concept précis au lieu de se baser sur des mots-clés qui peuvent être ambigus.

Prenons l'exemple d'une personne anglo-saxonne qui cherche à trouver l'adresse d'un installateur de fenêtres ; en tapant la requête « Windows installation » dans n'importe quel moteur de recherche, elle obtiendra des milliers de pages traitant l'installation du système d'exploitation de Microsoft et les problèmes qui en résultent, mais elle aura beaucoup de mal à trouver l'information qu'elle recherchait [6].

Avec l'utilisation d'une ontologie, un moteur de recherche fera la différence entre un site sur lequel 'Windows' désigne un logiciel et un autre sur lequel il désigne une fenêtre.

Cette recherche basée sur les ontologies se présente comme une recherche intelligente qui repose sur la sémantique des ressources et sur les concepts contenus dans les documents qui leur sont associés. Ces ontologies peuvent ainsi, d'une part, guider la création d'annotations sous la forme de métadonnées sur les ressources, et d'autre part, décrire leurs contenus de manière à la fois formelle et signifiante pour être exploitable aussi bien par les humains que par les machines.

Dans cette optique, plusieurs systèmes de recherche d'informations à base d'ontologies ont été proposés, parmi lesquels on peut citer : Ontobroker, Sesame et Corese. La différence entre ces systèmes réside essentiellement dans le langage de représentation et le moteur d'inférence sur les connaissances imbriquées dans les annotations : Ontobroker utilise F-Logic [22], Sesame utilise SQL92SAIL (du SQL adapté à RDF), et Corese utilise les graphes conceptuels [23].

4.3. Langages d'ontologie

Il existe plusieurs langages de spécification d'ontologies (ou langage d'ontologies) qui ont été développés pendant les dernières années, et ils deviendront sûrement des langages d'ontologie dans le contexte du Web sémantique. Certains d'entre eux sont basés sur la syntaxe de XML (Extended Markup Language), tels que XOL (Ontology Exchange Language), SHOE (Simple HTML Ontology Extension - qui a été précédemment basé sur le HTML), OML (Ontology Markup Language), RDF (Resource Description Framework), RDF Schéma. Les deux derniers sont des langages créés par des groupes de travail du World Wide Web Consortium (W3C).

Trois langages additionnels sont établis sur RDF(S) pour améliorer ses caractéristiques: OIL (Ontology Inference Layer), DAML+OIL (DARPA Agent Markup Language + Ontology Inference Layer) et OWL (Web Ontology Language). [24]

➢ RDF (Ressource Description Framework)

Le W3C a adopté le langage RDF [Web 7] comme formalisme standard de représentation. Utilisant la syntaxe XML, qui constitue déjà un standard, RDF permet de décrire des ressources Web en termes de ressources, propriétés et valeurs.

- Les ressources : Toutes choses décrites par des expressions RDF sont appelées des ressources (une page Web entière ou partie d'une page Web, un site Web, etc.). Elles sont identifiées par une URI (Universal Ressource Identifier) et ne sont pas obligatoirement accessibles via le Web (ex., un livre imprimé);
- Les propriétés : il s'agit de caractéristiques, d'attributs ou de relations utilisées pour décrire des ressources particulières ;
- Les valeurs : la valeur de la propriété peut être une autre ressource ou elle peut être littérale ; c.-à-d., une ressource (spécifiée par une URI) ou une simple chaîne ou autre type de données primitif défini par XML.

➢ RDF Schema (RDF Vocabulary Description Language)

RDF, langage dédié à l'expression d'assertions sur les relations entre objets, s'est heurté à la nécessité de définir les propriétés des classes dont ces objets sont instances. Cependant, l'extension à RDFS [Web 7] permet de déclarer les propriétés et le type des ressources.
RDFS fournit les primitives de modélisation permettant de construire des ontologies simples. Beaucoup lui ont donc reproché son manque d'expressivité par comparaison avec ce que les langages de représentation des connaissances pouvaient faire. Dans l'optique d'une utilisation d'ontologies sur le Web, le langage RDF(S) a été enrichi par l'apport du langage OIL (Ontology Interchange Language [25]) qui permet d'exprimer une sémantique à travers le modèle des frames tout en utilisant la syntaxe de RDF(S).

➢ OIL (Ontology Inference Layer)

Le langage OIL, combine les primitives de modélisation utilisées dans le langage de *frames* et le raisonnement formel des logiques de description pour exprimer des ontologies sur le Web [25]. Par rapport au langage RDFS, OIL reprend la distinction faite en logique de description entre concept *primitif* (décrit par des conditions nécessaires d'appartenance) et concept *défini* (décrit par des conditions nécessaires et suffisantes d'appartenance).
OIL offre de nouvelles primitives permettant de définir des classes à l'aide de mécanismes ensemblistes (intersection de classes, union de classes, complémentaire d'une classe).

➢ DAML+OIL (DARPA Agent Markup Langage)

DAML+OIL est issu de l'union de deux langages: le langage OIL et DAML-ONT [26]. L'intégration de OIL rend possible les inférences compatibles avec les logiques de description, essentiellement les calculs des liens de subsomption.
Ce nouveau langage supporte désormais les types de données primitifs, et la définition d'un certain nombre d'axiomes comme l'équivalence de classes ou de propriétés. Il est équivalent à une logique de descriptions très expressive, la logique *SHIQ*. Pour cette logique, il existe des procédures de décision correctes et complètes, et un certain nombre d'outils l'implémentant sont disponibles : OILed, OntoEdit, Protégé2000, et WebODE.

➢ OWL (Ontology Web Language)

Le langage OWL (Ontology Web Language) est destiné à être utilisé lorsque les informations contenues dans les documents doivent être traitées par des applications

logicielles, par opposition aux situations où le contenu est simplement présenté aux humains, cas traditionnel de l'utilisation du Web d'aujourd'hui [Web 8].
Pour des raisons de modularité, OWL a été fractionné en trois langages distincts :

- OWL Lite ne contient qu'un sous-ensemble réduit des constructeurs disponibles, mais son utilisation assure que la comparaison de types pourra être calculée. [27] montrent ainsi que ce langage est dans la même classe de complexité que la logique de description SHIF(D) et que le système Racer [28], par exemple l'implémente complètement.

- OWL DL contient l'ensemble des constructeurs, mais avec des contraintes particulières sur leur utilisation qui assurent la décidabilité de la comparaison de types. Par contre, la grande complexité de ce langage semble rendre nécessaire une approche heuristique.

- OWL Full, sans aucune contrainte est la fusion de RDFS et de OWL DL (c'est-à-dire qu'il abolit la distinction forcée entre classes et ressources); le problème de comparaison de types y est vraisemblablement indécidable.

Il existe également des langages d'interrogation basés sur la reconnaissance de graphe RDF : RDQL, SPARQL et nRQL. Le principe est de décrire un graphe RDF à l'aide de variables. Les résultats d'une requête étant les valeurs des variables pour lesquelles il existe un graphe RDF dans l'ontologie correspondant au graphe défini par la requête.

- RDQL (*RDF Data Query Language*) : est un langage d'interrogation de données définies en RDF. Ce langage n'est pas standardisé, il existe de nombreuses implémentations.
- SPARQL : est une amélioration de RDQL, ce langage est en cours de standardisation au niveau du W3C. SPARQL offre une syntaxe quasi identique à RDQL, mais y ajoute notamment les opérateurs UNION et OPTIONAL dans la clause WHERE. L'opérateur UNION définit la disjonction de triplets RDF.
- nRQL (new Racerpro query Language) : est le langage d'interrogation de Racer. Comme RDQL et SPARQL, nRQL est basé sur la recherche de graphes RDF. Sa syntaxe est proche des deux autres, sauf pour sa notation préfixée des opérateurs

4.4. Outils d'édition d'ontologie

Aujourd'hui, le nombre d'éditeurs d'ontologies a considérablement augmenté. Il existe différents éditeurs qui utilisent des formalismes variés et offrent différentes fonctionnalités.
Parmi ces outils on trouve : OILed, OntoEdit, Web ode, DOE, Protégé2000… voici la description de quelques-uns :

- OILEd [29]

OILEd a été conçu pour éditer des ontologies dans le langage de représentation OIL, il est souvent considéré comme une simple interface de la logique de description SHIQ. Cet éditeur offre également les services d'un raisonneur, FaCT qui permet de tester la satisfiabilité des définitions de classes et de découvrir des subsomptions restées implicites dans l'ontologie. L'outil dispose de mécanismes pour la classification et le contrôle de la cohérence des ontologies. La version 3.4 d'OILEd est gratuite et disponible sur le site web d'OILEd.

- OntoEdit [30]

OntoEdit est également un environnement de construction d'ontologies indépendant de tout formalisme. Des outils graphiques dédiés à la visualisation d'ontologies sont inclus dans l'environnement. ONTOEDIT intègre un serveur destiné à l'édition d'une ontologie par

plusieurs utilisateurs. Un contrôle de la cohérence de l'ontologie est assuré à travers la gestion des ordres d'édition.

- Ontolingua [31]

Ontolingua de l'Université Stanford. Le serveur Ontolingua est le plus connu des environnements de construction d'ontologies en langage Ontolingua. Il consiste en un ensemble d'environnements et de services qui supportent la construction en coopération d'ontologies, entre des groupes séparés géographiquement. Il supporte plusieurs langages et dispose de traducteurs permettant de passer de l'un à l'autre.

- ONTOSAURUS [32]

ONTOSAURUS de l' « Information Science Institute » de l'Université de Southern California. Ontosaurus consiste en un serveur utilisant LOOM comme langage de représentation des connaissances, et en un serveur de navigation créant dynamiquement des pages HTML qui affichent la hiérarchie de l'ontologie; le serveur utilise des formulaires HTML pour permettre à l'usager d'éditer l'ontologie. Des traducteurs du LOOM en Ontolingua, KIF, KRSS et C++, ont été développés.

- Protégé2000 [33]

Protégé2000 est une interface modulaire, développée au Stanford Medical Informatics de l'Université de Stanford7, permettant l'édition, la visualisation, le contrôle (vérification des contraintes) d'ontologies, l'extraction d'ontologies à partir de sources textuelles, et la fusion semi-automatique d'ontologies. Le modèle de connaissances de Protégé2000 est issu du modèle des frames et contient des classes (concepts), des slots (propriétés) et des facettes (valeurs des propriétés et contraintes), ainsi que des instances des classes et des propriétés. De nombreux plug-ins sont disponibles ou peuvent être ajoutés par l'utilisateur.

5. Evaluation de SRI

Pour comparer les systèmes de recherche d'information entre eux, des mesures d'efficacité ont été introduites. Ces mesures partaient du principe que la pertinence d'un document par rapport à une requête était binaire : un document est pertinent ou non, car les premiers systèmes construits étaient de modèle booléen. Donc, le corpus se divise en deux parties. Par conséquent, il fallait évaluer les SRI en fonction de leur capacité de retrouver l'ensemble des documents pertinents. Suite à une recherche, le corpus se divise en quatre ensembles (A, B, C, D) de documents schématisé dans la figure suivante.

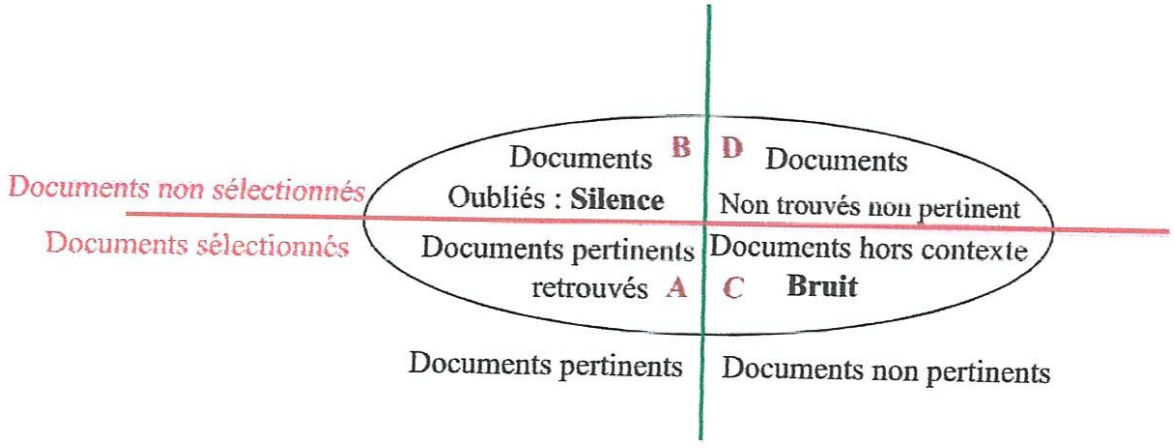

Figure 1.6: *Les quatre ensembles de documents résultats en RI.*

5.1. Les mesures d'évaluation d'un SRI

5.1.1. Rappel

Le rappel est défini par le nombre de documents pertinents retrouvés au regard du nombre de documents pertinents que possède le corpus de documents. [44]

Cela signifie que lorsque l'utilisateur interroge le moteur de recherche, il souhaite voir apparaître tous les documents qui pourraient répondre à son besoin d'information. Si cette adéquation entre le questionnement de l'utilisateur et le nombre de documents présentés est importante alors le taux de rappel est élevé. À l'inverse si le système possède de nombreux documents intéressants mais que ceux-ci n'apparaissent pas dans la liste des réponses, on parle de silence. Le silence s'oppose au rappel.

$$\text{Rappel} = \frac{\text{Nombre de documents pertinents retrouvés}}{\text{Nombre de documents pertinents}}$$

5.1.2. Précision

La précision est le nombre de documents pertinents retrouvés rapporté au nombre de documents total proposé par le moteur de recherche pour une requête donnée [45].

Le principe est le suivant : quand un utilisateur interroge une base de données, il souhaite que les documents proposées en réponse à son interrogation correspondent à son attente. Tous les documents retournés superflus ou non pertinents constituent du bruit. La précision s'oppose à ce bruit documentaire. Si elle est élevée, cela signifie que peu de documents inutiles sont proposés par le système et que ce dernier peut être considéré comme "précis". On calcule la précision avec la formule suivante :

$$\text{Précision} = \frac{\text{Nombre de documents pertinents retrouvés}}{\text{Nombre de documents retrouvés}}$$

5.1.3. Courbe Rappel/Précision :

Il est intéressant pour étudier la qualité de l'ordonnancement des scores de similarités obtenus par chaque document avec la requête, de regarder la précision ou le rappel du sous ensemble de n premiers documents retrouvés. Dans ce contexte, une autre manière de représenter le couple précision-rappel en fonction de n est de représenter pour chaque n, le nuage des points des couples précision-rappel. Ainsi, en reliant les points par un segment, pour construire l'allure de la courbe Rappel/Précision [46]. Cette courbe calcule la précision pour les 11 valeurs prédéfinies du rappel, de 0 % à 100 % par un pas de 10%. Elle oppose le rappel en abscisse et la précision en ordonnée. L'idée est de décrire la pertinence de l'ensemble de documents obtenus à l'issue d'une requête sur un système quelconque. La courbe est globalement décroissante. La figure suivante illustre un exemple d'évaluation de SRI en utilisant la courbe Rappel/précision.

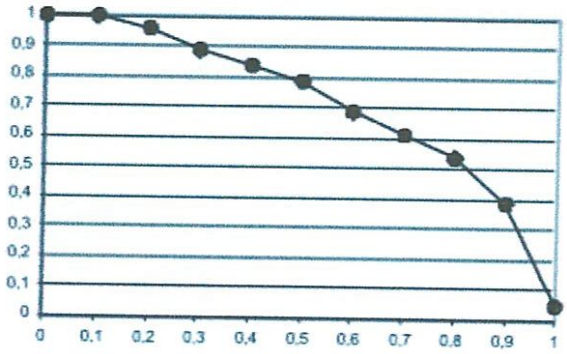

Figure 1.7: *Courbe Rappel/Précision*

Cette courbe montre que la précision décroit au fur et à mesure que des documents non pertinents sont restitués, le rappel ne varie pas, puis un document pertinent est restitué, alors la précision et le rappel augmentent, puis continuent de croître tant que les documents restitués sont pertinents et décroissent sinon. Plus cette courbe décroît tardivement, meilleur est l'algorithme à l'origine de l'ordonnancement étudié.

6. Domaines d'application du web sémantique

Le Web devient un vecteur communicationnel véritablement utilisable par tous. Ainsi les machines, elles aussi, peuvent consulter et interpréter les informations. Il s'agit d'arriver à un Web "intelligent", où les informations ne seraient plus stockées mais "comprises" par les ordinateurs. Le Web sémantique offre une plate-forme grâce à laquelle les données peuvent être partagées et réutilisées par les applications, les entreprises et les communautés.

Il permet une meilleure exploitation des sources d'informations [Web 1]

- Recherche d'information: une ontologie peut jouer le rôle de méta-data servant d'index dans un répertoire d'informations.
- Intégration des sources hétérogènes : dans une application "entrepôt de données", une ontologie peut jouer le rôle d'un schéma conceptuel commun reliant entre elles plusieurs sources d'informations hétérogènes.
- Interface Homme-Machine : la visualisation de l'ontologie permet à l'utilisateur de comprendre le vocabulaire utilisé par le SI et de mieux formuler ses requêtes.
- Requêtes : une ontologie linguistique peut permettre de comprendre les requêtes (représentation du contenu) de l'utilisateur formulées en langue naturelle.

Conclusion

Dans ce chapitre, nous avons présenté les concepts de base de la recherche d'information. En particulier, nous avons exposé les différentes étapes du processus de recherche d'information, et explicité les bases de l'indexation, de l'appariement et des modèles de recherche. Puis nous avons décrit les concepts qui ont incorporés la sémantique dans la recherche d'information notamment le web sémantique et les ontologies.

Le but d'un système de recherche d'information est de rechercher l'information pertinente pour une requête utilisateur. Dans le prochain chapitre, nous présenterons le rôle des ontologies dans l'intégration sémantique des sources de données hétérogènes.

CHAPITRE 02

INTEGRATION SEMANTIQUE DES SOURCES DE DONNEES HETEROGENES

Introduction

L'explosion du nombre de sources d'information accessibles à travers le Web multiplie et la quantité de données gérées au sein des bases de données des entreprises a rendu indispensable l'intégration de données provenant de différentes sources.
L'intégration des données est le processus par lequel plusieurs sources de données autonomes, réparties et sous forme hétérogène (où chaque source est associée à un schéma local) sont intégrées sous forme de source unique représentée par un schéma global.
Les solutions à l'intégration d'information proposées dans le cadre du Web sémantique tireront parti des recherches concernant les approches médiateurs et les entrepôts de données.
Un système de médiation a pour rôle d'offrir à l'utilisateur une vue uniforme et une interrogation transparente des informations sans que l'utilisateur n'ait le souci de la provenance des informations ni de leur format d'origine.

Ce chapitre fournit une description générale des systèmes d'intégration et un aperçu plus détaillé de système de médiation sémantique et les méthodes spécifiques qui sont utilisées pour l'intégration de données.

1. Hétérogénéité des données

Avec l'apparition de l'internet et le développement des différentes représentations et formats des documents, les données peuvent être de plusieurs types : structurées (données relationnelles, données objet), semi-structurées (HTML, XML, graphes) ou même non structurées (texte, images, son). Dans un tel contexte, le besoin d'intégration se fait de plus en plus sentir. Cependant, pour répondre à ce besoin, le développement des applications d'intégration se voit contraint de composer avec la répartition des sources et l'hétérogénéité de leurs structures. De plus, il n'y a pas de langage universel d'interrogation de données hétérogènes mais plusieurs langages peuvent être utilisés. Par ailleurs, il n'y a pas de modèle de représentation unique pour les données intégrées. Plusieurs modèles, selon les caractéristiques des sources et les manipulations à effectuer peuvent être retenus.
La problématique abordée est comment utiliser ces données comme si elles constituaient une seule base de données homogène. L'intégration de données hétérogènes a conduit à de nombreux problèmes liés à la nature des données. Les données sont hétérogènes selon quatre dimensions : Sémantique, structurelle, syntaxique et communication.

1.1. Hétérogénéité sémantique

Elle est due aux conflits sémantiques dans les termes, les expressions, etc., qui sont adoptés par différents schémas de données mais exprimés de diverses manières [71]. Autrement dit, elle est due aux différentes interprétations pour les objets du monde réel. Mêmes valeurs issues des sources différentes peuvent signifier différentes choses. Par exemple, l'attribut Title dans une source de données signifie Job Title, tandis que dans une autre source de données, il signifie Person Title. En effet, les sources de données ont été conçues indépendamment par des concepteurs différents ayant des objectifs applicatifs différents. Chacun peut donc avoir un point de vue différent sur le même concept. L'interopérabilité sémantique de données présente un défi majeur dans le processus d'élaboration des systèmes d'intégration [39].

1.2. Hétérogénéité structurelle

Elle provient quand les sources adoptent différents modèles de données, structures de données ou schémas, par exemple les modèles de bases de données relationnelles ou orientées objets. De nombreux travaux concernant ce type d'hétérogénéité ont été proposés dans les contextes des bases de données fédérées et des multi-bases de données [40].

Elle est due aussi aux différents types attribués à la même donnée. Par exemple : l'attribut Phone peut être défini par le type String ou Number.

1.3. Hétérogénéité syntaxique

Elle apparaît lorsque différents vocabulaires sont utilisés pour représenter les données, par exemple le mot Professeur ou Prof. Ainsi, lorsque certaines données ne sont pas renseignées correctement et différentes abréviations attribuées aux mêmes données et vice versa [40].

1.4. Hétérogénéité de communication

Certains systèmes décrivent les données dans des interfaces web et d'autres par des applications implémentées par des langages de programmations différents. Ainsi, certaines applications peuvent offrir un langage d'interrogation direct, d'autres à travers des API [39].

2. Intégration de données

Au cours des vingt dernières années, trois architectures complémentaires de systèmes d'intégration de données se sont imposées : les entrepôts de données, les médiateurs et les systèmes pair-à-pair. Les entrepôts de données stockent localement une partie des données des sources distantes. L'avantage est d'avoir un temps de réponse faible puisque les requêtes posées sont évaluées localement. Toutefois, l'inconvénient majeur est qu'il faut trouver une bonne fréquence de rafraichissement des données de l'entrepôt afin qu'elles ne se périment pas.

Contrairement aux entrepôts, les médiateurs ne stockent pas de données localement. Ceci les rend moins efficaces que les entrepôts lors de l'évaluation de requêtes, mais permet de garantir que les réponses sont calculées sur des données non périmées : les données des sources.

Enfin, les systèmes pair-à-pair permettent le passage à l'échelle des applications d'intégration d'informations, lorsque le nombre de sources à intégrer n'est plus gérable efficacement via une architecture centralisée. Dans notre travail, nous nous intéressons à la deuxième architecture à base de Médiateur.

Dans les systèmes de médiation, l'intégration de données consiste à éliminer d'abord les conflits entre les données causées par leur hétérogénéité (telle que cela a été présenté dans la section ci-dessus) et ensuite les représenter dans un seul schéma cohérent. Tout système d'intégration doit fournir les solutions aux problèmes suivants : (i) l'identification et la spécification des correspondances entre des données sémantiquement liées, (ii) fournir une vue globale intégrée des données représentées à travers des conceptualisations différentes [41].

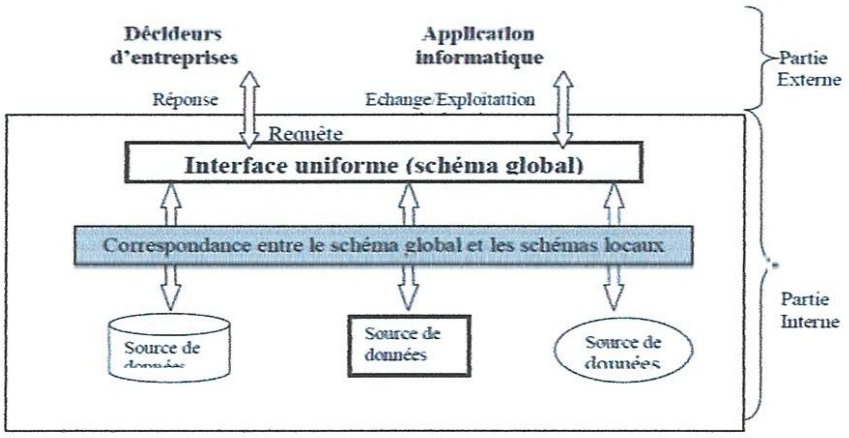

Figure 2.8: Système d'intégration à base de Médiateur [40]

Généralement, un système d'intégration comporte deux parties principales une *partie externe* et une *partie interne* (voir la figure 2.1):

- La partie externe :

Correspond aux utilisateurs du système intégré comme, par exemple, les décideurs d'une entreprise, ou d'autres systèmes.

- La partie interne :

Comprend d'une part les sources d'informations participant dans le processus d'intégration et d'autre part une interface (schéma global) permettant aux éléments de la partie externe d'accéder d'une manière transparente aux sources de données. Cette interface est une couche sans données propres et les éléments externes au système ne voient pas les sources internes. Celles-ci sont *encapsulées*, *cachées* par l'interface. Les éléments externes doivent pouvoir agir sur le système d'intégration d'une manière transparente via un *schéma global*.

3. Approches d'intégration de données

Pour faciliter la compréhension des caractéristiques, des avantages et des faiblesses des systèmes existants, nous citons deux critères de classification parmi ceux cités dans la littérature qui sont en relation avec notre travail, à savoir : (1) la représentation de données intégrées : virtuelle ou matérialisée, et (2) le sens de la mise en correspondance entre schéma global et les schémas locaux.

Comme cela a été mentionné plus haut, il existe essentiellement deux principales approches pour intégrer des données provenant de différentes sources: les systèmes de médiation et les entrepôts de données. Les systèmes de médiation permettent d'accéder par une même requête à des données se trouvant dans des sources différentes. Les entrepôts de données intègrent au sein d'une même base de données provenant de sources différentes, permettant ainsi d'effectuer des analyses et des fouilles de données sur la nouvelle base. Ceci constitue le premier point de classification :

3.1. Selon le degré de matérialisation

Les données à réutiliser peuvent être redéfinies sous forme d'une base de données virtuelle (système de médiation) ou matérialisée (entrepôt de données), assurant l'intégration logique des données sous-jacentes. Ce critère permet donc d'identifier le type de stockage de données du système d'intégration.

3.1.1. Approche de médiation (Virtuelle)

Cette approche consiste à développer une application chargée de jouer le rôle d'interface entre les sources de données locales et les applications d'utilisateurs. Ce type d'approche a été utilisé par un nombre important de projets [45]. Il repose sur deux composants essentiels : le médiateur et l'adaptateur.

- Le médiateur : chargé de la localisation des sources de données et des données pertinentes par rapport à une requête, il résout de manière transparente les conflits de données. Un ensemble de connaissances sur les sources permet au médiateur de générer un plan d'exécution pour traiter les requêtes d'utilisateurs.
- L'adaptateur : c'est un outil permettant à un (ou plusieurs) médiateur(s) d'accéder au contenu des sources d'informations dans un langage uniforme. Il fait le lien entre la représentation locale des informations et leur représentation dans le modèle de médiation.

Dans cette approche, les données ne sont pas stockées au niveau du médiateur. Elles restent dans les sources de données et ne sont accessibles qu'à ce niveau. L'intégration de données est fondée sur l'exploitation de vues abstraites décrivant de façon homogène et uniforme le

contenu des sources de données dans les termes du médiateur. Les sources de données pertinentes, pour répondre à une requête, sont calculées en fonction de la définition de ces vues. Le problème consiste à trouver une requête qui est équivalente à la requête de l'utilisateur. Les réponses à la requête posée sont ensuite obtenues en évaluant cette requête sur les extensions des vues. Le problème d'hétérogénéité sémantique entre les différentes sources d'information ne se pose pas pour l'utilisateur puisqu'il est résolu par le médiateur. L'approche médiateur présente l'intérêt de pouvoir construire un système d'interrogation de sources de données sans toucher aux données qui restent stockées dans leur source d'origine. Par contre, le médiateur ne peut pas évaluer directement les requêtes qui lui sont posées car il ne contient pas de données, ces dernières étant stockées de façon distribuée dans des sources indépendantes. Dans cette approche, deux problèmes apparaissent : comment construire le schéma intégré et comment effectuer la mise en correspondance entre les termes utilisés dans ce schéma et les structures des schémas des sources qui contiennent les données.

Un système de médiation est donc constitué d'un système global de gestion de médiateur, d'un ensemble d'adaptateurs et d'un ensemble de sources locales. Les sources locales sont intégrées dans le sens qu'un seul schéma global est construit à partir des schémas exportés par les sources locales. Des requêtes sont ainsi posées sur le schéma global et traitées par le médiateur. Les schémas des sources sont construits par des adaptateurs qui établissent la correspondance entre la source et le médiateur.

La plupart des systèmes de gestion de données hétérogènes et distribuées adoptent l'architecture à base de médiateurs et d'adaptateurs telle que définie par [45].

3.1.2. Approche d'entrepôt de données (Matérialisée)

Cette approche consiste à centraliser physiquement, dans un entrepôt de données, l'ensemble de données consolidées à partir de diverses sources (catalogues électroniques, bases de données relationnelles, Web, etc.). Le processus de construction d'un système d'intégration matérialisé se compose en quatre étapes principales [46]:

- l'extraction des données des sources de données,
- la transformation des données au niveau structurel,
- l'intégration sémantique des données,
- le stockage des données intégrées dans le système cible.

Ces étapes correspondent aux outils d'ETL (extract, transform and load) [48]. Les deux premières étapes sont habituellement prises en charge par le même composant logiciel, appelé adaptateur. Comme dans l'approche médiateur, l'objectif d'un adaptateur est d'aboutir à des données représentées dans un même format. Chaque adaptateur fournit ainsi une interface d'accès et de requêtes sur la source. Les données extraites sont ensuite intégrées en éliminant les conflits, puis stockées dans l'entrepôt [47].

3.2. Selon le mapping entre schémas

Un autre critère de classification est le mapping entre schémas qui sert à définir la relation entre le modèle global et celui des sources [49]. Pour ce faire, plusieurs approches et systèmes d'intégration ont été proposés dans la littérature. Nous distinguons principalement deux types d'approches: un schéma global défini comme vue sur les schémas locaux GAV (Global As View) ou des vues locales définies comme vues du schéma global LAV (Local As View) [50].

3.2.1. Approche GAV

Un schéma global est considéré comme une vue sur des schémas sources. GAV (Global As View), a été la première à être proposée pour l'intégration de données. Dans l'approche GAV, pour chaque relation utilisée dans le schéma global, on définit une vue composée des termes des relations des schémas sources. La transformation d'une requête écrite en utilisant le

schéma global en une requête écrite en utilisant des schémas sources est une opération simple, cette simplicité est considérée comme avantage de l'approche GAV.
Les inconvénients de cette approche est qu'il suppose que les sources à intégrer soient connues à l'avance, ainsi une modification sur l'ensemble des sources ou sur leurs schémas entraîne une reconsidération complète du schéma global.

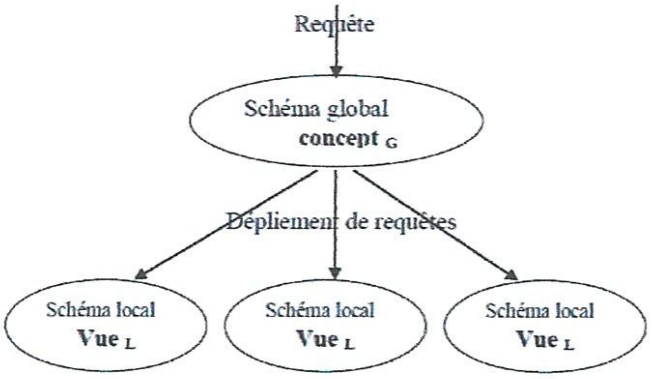

Figure 2.9: *Approche GAV [40]*

3.2.2. Approche LAV

Dans LAV (Local As View), chaque relation d'un schéma source est définie comme vue sur le schéma global. Dans le cas d'une approche LAV, la requête sur le schéma global doit être reformulée suivant les schémas sources. Dans cette approche, chaque source est spécifiée de manière indépendante.
Elle a l'avantage d'être très flexible par rapport à l'ajout (ou la suppression) de sources de données à intégrer, cela n'a aucun effet sur le schéma global, seules des vues doivent être ajoutées (ou supprimées). En contrepartie de cette flexibilité et cette simplicité de mise à jour est la complexité de la construction des réponses à une requête dans un médiateur conçu selon l'approche LAV. La reformulation de requêtes en termes de vues est la seule possibilité pour obtenir des requêtes exprimées en termes de vues, que l'on doit ensuite exécuter pour interroger les sources de données [42].

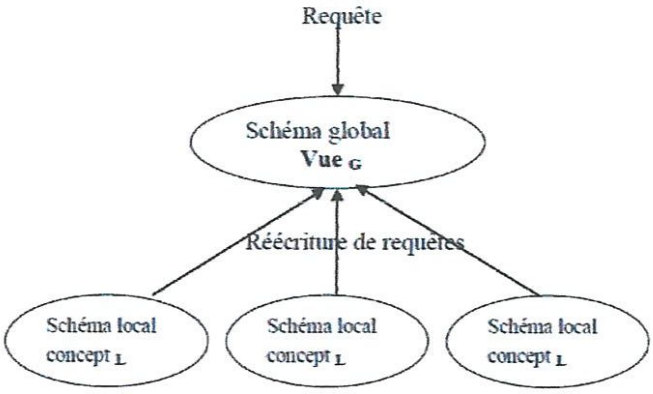

Figure 2.10: *Approche LAV [40]*

Les principaux systèmes développés autour de cette approche sont : Infomaster [56], PICSEL [52], Information Manifold [39], Razor [53], Internet Softbot [54], OBSERVER [55].
InfoMaster, par exemple, présente à l'utilisateur un catalogue virtuel, appelé schéma référencé, sur lequel il formule ses requêtes. Ce système utilise le langage KIF (Knowledge Interchange Format) pour définir une liste de règles qui associe les schémas des sources

intégrées aux vues du schéma référencé. Ces règles permettent aussi de convertir l'information afin de la mettre sous format adéquat et de l'adapter aux sources. Lors de l'ajout d'une nouvelle source, le système d'intégration ajoute de nouvelles règles représentant la vue de la nouvelle source.

3.2.3. Approche hybride

Il y a eu des tentatives de fusionner les approches LAV et GAV de sorte qu'il serait facile de reformuler la requête, comme c'est le cas dans l'approche GAV, ainsi que pour modifier les schémas locaux, comme c'est le cas dans l'approche LAV. Les approches GAV et LAV ont toutes les deux des inconvénients. Ainsi, d'autres s'approchent hybrides ont été proposées [56]. En effet, la littérature décrit plusieurs méthodes (Both As View [57], GLAV (Global Local As View) [53] et BGLAV (Both Global Local As View) [58].

3.3. Bilan

Une étude comparative sur les différentes approches nous permet de déduire les points suivants :

- L'approche de médiation peut être qualifiée d' «approche paresseuse » dans la mesure où il n'y a pas de matérialisation dans celle-ci, les requêtes migrent vers les sources de données. L'inconvénient majeur de cette approche est sa faible performance. Aussi, les sources de données ont des capacités différentes et les requêtes doivent toujours être traduites. Toutefois, elle présente l'avantage d'être plus souple (c'est-à-dire qu'il est facile d'y ajouter de nouvelles sources de données) et de distribuer l'effort entre le médiateur et l'adaptateur. De plus, les données y sont toujours fraîches. C'est la méthode la plus utilisée.

- L'approche entrepôt de données offre un accès très efficace à une copie des données. Les sources de données sont matérialisées au niveau du modèle global et les données migrent vers l'entrepôt. Cette approche est plus performante que la précédente et permet la personnalisation des données (nettoyage, filtrage). Cependant, la cohérence et la fraicheur des données, ainsi que la gestion des mises à jour restent difficiles à assurer à cause du volume important des données.

Par ailleurs, l'approche GAV est simple et facile à mettre en Œuvre, mais limite la vue de toutes les données possibles qui peuvent être générées. Le traitement de requêtes est un mécanisme absolument obligatoire dans l'intégration des bases de données. Il commence par une reformulation de celle-ci à partir du schéma global aux schémas à l'exportation des sources de données. Dans l'approche GAV, les requêtes sont reformulées simplement par le remplacement des éléments changeant [47]. Les liens sémantiques entre le schéma global et les sources sont définis en décrivant les prédicats du schéma global comme des requêtes sur les vues.

Et l'approche LAV, contrairement à GAV est plus complexe et difficile à mettre en Œuvre; par contre elle est plus extensible et offre de nouvelles sources qui définissent comment elles peuvent contribuer au schéma global. Dans cette approche, la reformulation des requêtes est beaucoup plus compliquée, et est connue comme étant la réponse aux requêtes formulées par l'utilisateur de vues [48]. Pour la définition des liens sémantiques entre le schéma global et les sources de données, selon l'approche LAV, les vues sont exprimées comme des requêtes sur le schéma global.

Lorsqu'une requête est reformulée, son optimisation a lieu, parce que les sources de données peuvent avoir des capacités limitées pour satisfaire les requêtes.

Une fois l'optimisation terminée et un plan d'exécution préparé, la requête est exécutée. Les requêtes sont soumises aux adaptateurs (wrappers) qui les reformulent dans la syntaxe de la source de données et pose ainsi une requête à la source de données concrète. Par la suite, les résultats sont reformulés dans la syntaxe et la sémantique du système et sont intégrés.

Dans ce mémoire, nous nous proposons de concevoir un système d'intégration suivant l'approche médiateur et utilisant la méthode LAV pour définir les relations entre le schéma global et les sources de données.

4. Processus d'intégration de données à base de médiateur

On distingue différentes approches pour effectuer l'intégration de données via l'approche de médiation : manuelle, semi-automatique et automatique.

4.1. Approche manuelle

Les premières approches proposées étaient des approches *manuelles*. Un processus d'intégration manuel permet d'automatiser l'intégration des données au niveau syntaxique. Les conflits sémantiques sont gérés manuellement et nécessitent la présence d'un expert humain pour interpréter la sémantique des données. Plusieurs systèmes ont été développés selon cette approche comme les systèmes multi-bases de données, la fédération des bases de données, le système TSIMMIS [47].... Ces approches manuelles deviennent impraticables lorsque le nombre de sources de données à intégrer est important, ou lorsque les sources évoluent fréquemment.

4.2. Approches Semi-automatiques

Afin d'apporter plus d'automatisation dans la résolution des conflits sémantiques, plusieurs travaux se sont tournés vers les ontologies. Une ontologie permet de fournir la sémantique des concepts d'un domaine de manière formelle.
L'approche semi-automatique repose sur des ontologies linguistiques et permettent d'automatiser partiellement la gestion des conflits sémantiques. Les ontologies linguistiques traitent des termes, et non des concepts. Ceci peut créer des conflits de noms.
MOMIS [59] est un exemple de projet reposant sur l'ontologie linguistique Wordnet pour l'intégration des sources.

4.3. Approche automatique

L'approche automatique consiste à associer aux données des sources une ontologie conceptuelle qui en définit le sens. Une ontologie conceptuelle traite les concepts d'un domaine donné. Elle est définie comme « une spécification formelle et explicite d'une conceptualisation partagée d'un domaine de connaissance » [10]. La sémantique du domaine est ainsi spécifiée formellement à travers des concepts, leurs propriétés ainsi que les relations entre les concepts. La référence à une ontologie permet d'éliminer automatiquement les conflits sémantiques entre les sources. La connexion entre les sources et les ontologies peut se faire de plusieurs manières [60]: définition des termes de la BD par les concepts ontologiques (Projet Buster [61]), annotation des données (Projet SHOE [62]), enrichissement de la source par des règles logiques (logique de description) (Projet Picsel [63])...

Avant de présenter les systèmes de médiation sémantique, nous allons au préalable décrire l'architecture d'un système de médiation classique

5. Architecture d'un système de médiation

L'architecture de médiation fut conçue en 1992 par Gio Wiederhold dans l'article fondateur «*Mediator in the architecture of future information systems*» [45]. Il y développe une nouvelle vision de l'architecture du traitement de l'information en entreprise, il tente de régler

la problématique de l'accès et de l'intégration de l'information en introduisant la notion de médiateur.

Le médiateur est décomposé par quatre composants essentiels qui sont : le schéma global, les mappings du schéma global avec les sources, les fonctions de réécriture de requêtes et les fonctions de composition des résultats. Tous ces composants sont répartis en trois couches principales : la couche de médiation, la couche des wrappers et la couche de sources de données.

La figure suivante illustre cette architecture.

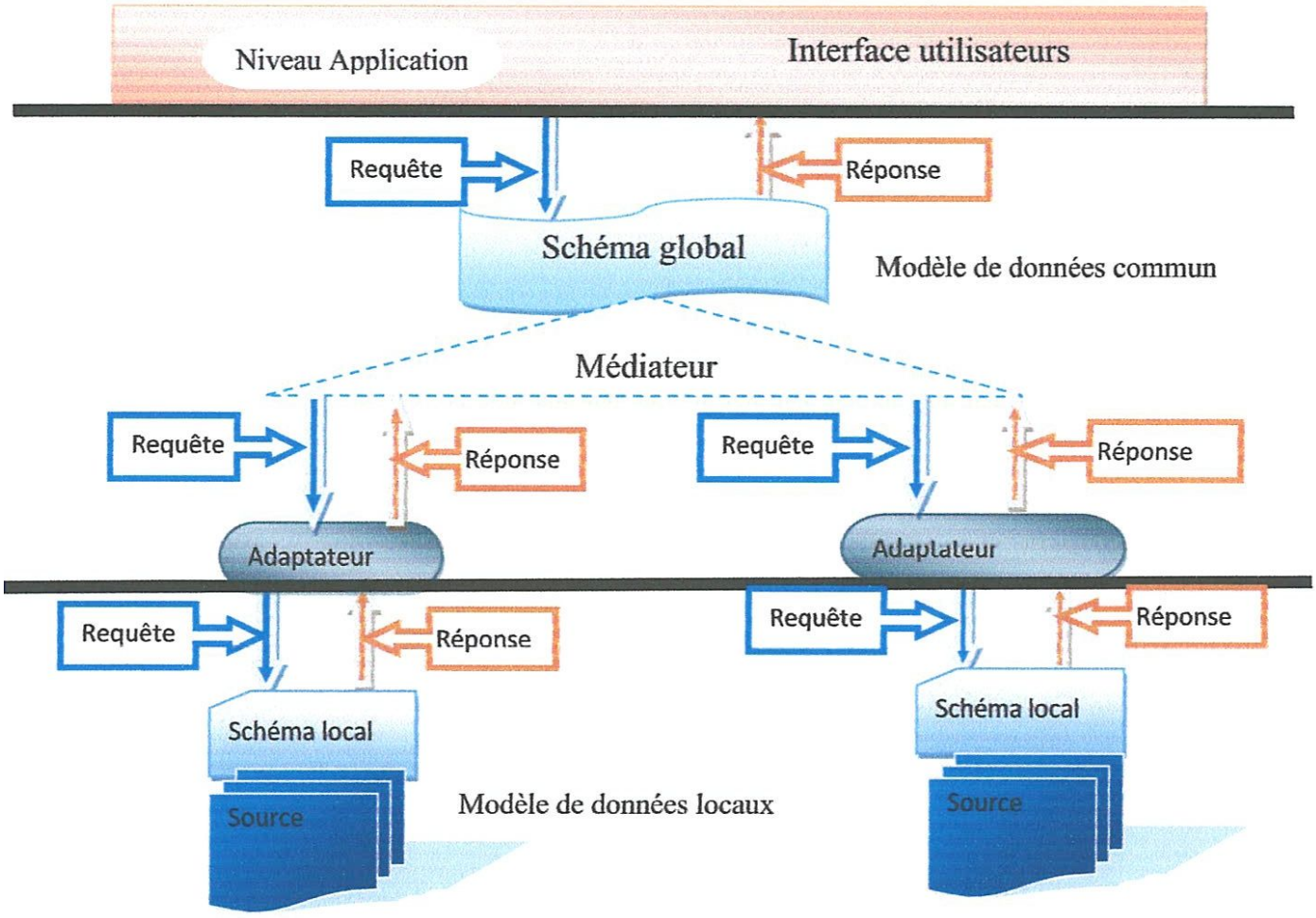

Figure 2.11: *Architecture d'un système de médiation*

En commençant par la couche la plus haute qui est la couche de médiateur.

5.1. Couche de Médiateur

Le rôle d'un médiateur est de collecter, nettoyer et combiner les données attendues par le système de médiation et produites par les différents adaptateurs [64]. C'est un module logiciel recevant directement la requête d'un usager et devant la traiter. Celui-ci doit localiser l'information nécessaire pour répondre à la requête, résoudre les conflits schématiques et sémantiques, interroger les différentes sources et intégrer les résultats partiels dans une réponse homogène et cohérente.

Quand une requête est soumise au médiateur, elle est posée indépendamment de la localisation des différentes données intervenant pour calculer le résultat. Cela introduit les difficultés suivantes :

- La décomposition d'une requête : il s'agit à partir d'une requête posée sur une vue intégrée, de localiser les données intervenant dans sa résolution, de produire des sous-requêtes spécifiques à chacune des sources, de les ordonner et éventuellement d'introduire des opérateurs au niveau du composant de médiation afin de compléter cet ensemble de sous-requêtes.
- La recomposition des résultats : une fois les sous-requêtes soumises à chacune des sources, il s'agit de savoir recomposer les différents résultats entre eux. Les résultats de chacune des sous-requêtes peuvent éventuellement faire l'objet d'un traitement additionnel soit parce que l'évaluateur de sous-requêtes n'a pas la capacité nécessaire pour traiter entièrement cette dernière, soit parce que les sous-requêtes comportent des dépendances entre elles.
- Au niveau de l'optimisation : le médiateur a rarement une vision sur la façon dont sont traitées les sous-requêtes au niveau des sources (placement des données, type de stockage, indexation, stratégie d'évaluation).

5.2. Couche d'Adaptateur

Un adaptateur prend en compte les dimensions Distribution, Autonomie et Interopérabilité. Son objectif est d'accéder à une source, d'extraire les données appropriées et de les présenter dans un format spécifié. Un adaptateur est donc un composant qui :

- Réalise la transformation entre le modèle de données (schéma local) dans lequel sont représentées les données de la source et le modèle choisi au niveau médiateur (schéma global).
- Effectue la transformation entre les expressions de requêtes du niveau global en expressions compréhensibles par les sources. Inversement, il traduit les réponses des sources au format du niveau global.
- Assure les traitements spécifiques non disponibles au niveau local.

5.3. Couche de source de données

La couche de source contient des sources de données externes. Ces sources locales sont en très grandes quantités et elles sont associées par leur propre ontologie (appelée ontologie locale) et permettent de représenter les connaissances d'une manière indépendante avec d'autres ontologies locales. Les sources de données sont sémantiquement hétérogènes par exemple : les substantifs de table relationnelle (pollution chimique et organique) et attributs (Cause et Conséquence). Et l'hétérogénéité de syntaxe Par exemple, le mot "mémoire" peut faire référence à des entités différentes: "support de stockage" et "mémoirc dc fin d'étudc". En outre, cette grande quantité de données est hétérogène dans la structure du schéma et le nombre de propriétés. [78]

6. Approches de médiation sémantique à base d'ontologie

Les ontologies peuvent contribuer à la résolution du problème de l'hétérogénéité sémantique. En effet, elles offrent une description formelle des concepts et de leurs relations existant dans certains domaines. Un utilisateur ou concepteur peut grouper et déclarer explicitement la sémantique des concepts et des données fournis par les sources.

Ainsi, en exploitant cette sémantique explicite et partagée, l'impact nuisible de l'hétérogénéité sémantique peut être réduit. Il s'agit dans ce cas d'une approche d'intégration de données fondée sur les ontologies. Une fois que les données des sources sont rendues conformes à l'ontologie, les requêtes peuvent être exprimées en termes de l'ontologie et les données

peuvent être interrogées via une seule interface de requêtes [60]. Dans un système de médiation, les requêtes sont réécrites en fonction des vues sur les sources et les extensions de ces vues sont ensuite intégrées et combinées pour pouvoir répondre à l'utilisateur.

Dans la littérature, nous distinguons trois approches en fonction de la façon dont les ontologies sont utilisées au sein d'une infrastructure d'intégration :

6.1. Approche à base d'une seule ontologie

Cette approche utilise une ontologie globale fournissant un vocabulaire partagé pour la spécification de la sémantique. Toutes les sources d'informations sont reliées à une ontologie globale. Une approche importante pour ce type d'intégration des ontologies est SIMS [65]. Un modèle indépendant de chaque source d'information doit être décrit pour ce système en reliant les objets de chaque source au modèle du domaine global.

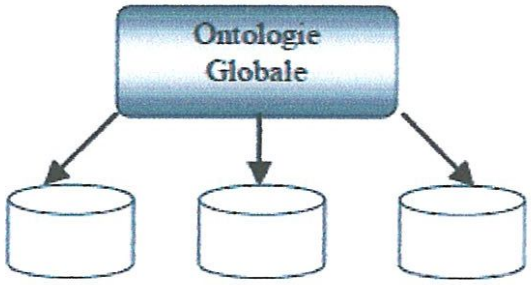

Figure 2.12: *Simple ontologie [66]*

6.2. Approche à base de multiples ontologies

Chaque source d'information est décrite par sa propre ontologie. L'avantage de cette approche est la possibilité de définir chaque ontologie source sans prendre en considération les autres sources ou autres ontologies [67]. Mais le manque d'un vocabulaire commun conduit à une difficulté extrême pour comparer différentes ontologies sources. Pour surmonter ce problème, un formalisme de représentation additionnelle définissant le mapping inter ontologie est fourni. Ce dernier identifie sémantiquement les termes correspondants des différentes ontologies sources. Ce mapping est difficile à définir à cause de divers problèmes d'hétérogénéité sémantique qui peuvent se produire. L'approche OBSERVER est utilisée pour ce type [67].

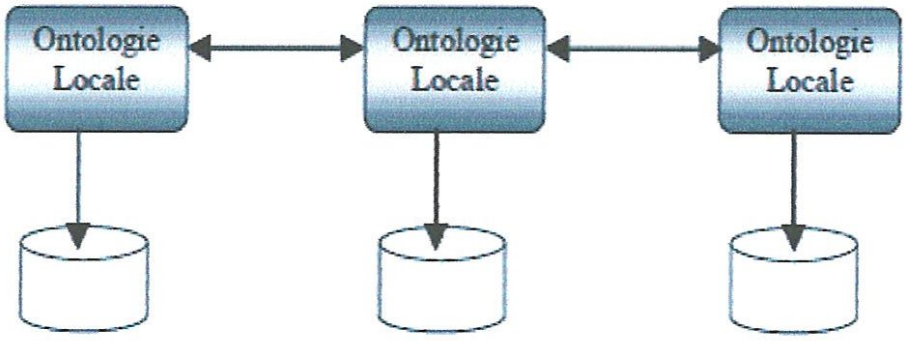

Figure 2.13: *Ontologie multiple [66]*

6.3. Approche hybride

Comme l'approche multiple, la sémantique de chaque source est décrite par sa propre ontologie. Mais pour rendre les sources d'ontologies comparables, elles sont construites sur un vocabulaire partagé global [68] [69]. Ce vocabulaire partagé comprend les termes de bases ou primitives d'un domaine. Afin de construire des termes complexes, ces primitives sont combinées par des opérateurs, et les termes seront comparés plus facilement qu'une approche multiple. Parfois, ce vocabulaire partagé est représenté sous forme d'ontologie [70]. Dans COIN [68], la description locale d'une information, qu'on appelle contexte, est simplement un vecteur de valeurs d'attributs. Dans METACOTA [69], chaque source d'information est annotée par un label qui indique la sémantique de l'information. Ce label combine les termes primitifs du vocabulaire partagé. Dans BUSTER [70], le vocabulaire partagé est une ontologie générale. Une source d'ontologie est un raffinement de l'ontologie générale. L'avantage de cette approche est que des nouvelles sources peuvent être ajoutées sans le besoin de la modification du vocabulaire partagé. Elle maintient encore l'acquisition et l'évolution des ontologies.

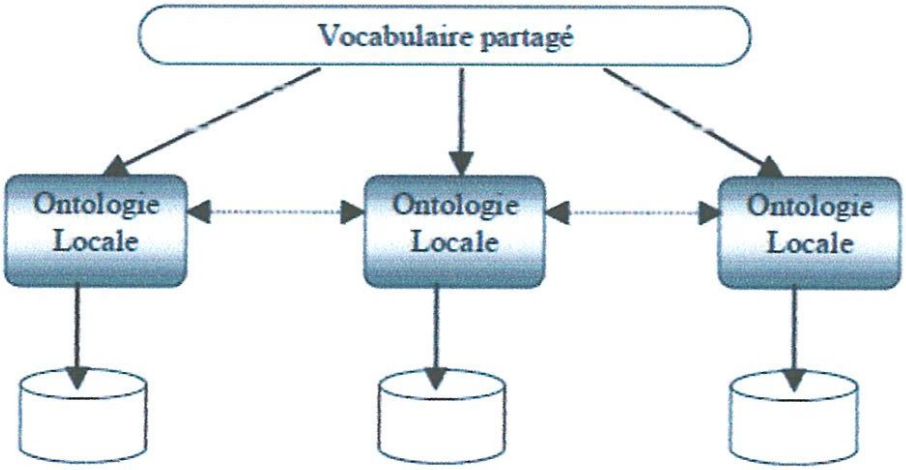

Figure 2.14: Ontologie hybride [66]

Pour notre système de médiation sémantique à base d'ontologie, nous utiliserons la première approche qui est celle à base d'une seule ontologie au niveau de médiateur.

7. Mesure de similarités entre les concepts d'une ontologie

Les approches de mapping heuristiques s'appuient sur des mesures de similarité sémantique pour établir les relations entre les concepts ; il existe dans la littérature plusieurs modèles de mesure de la similarité sémantique qui peuvent être regroupés selon la représentation des concepts qu'ils utilisent. Parmi les mesures de similarité sémantique on retrouve trois types principaux que nous allons maintenant décrire.

7.1. Approche basée sur les arcs

La mesure de similarité la plus intuitive des objets dans une ontologie est leurs distances [72]. Evidemment, un objet X est plus similaire à un objet Y qu'un objet Z. Cette similarité est évaluée par la distance qui sépare les objets dans l'ontologie.

Ces mesures se servent de la structure hiérarchique de l'ontologie pour déterminer la similarité sémantique entre les concepts. Le calcul des distances dans l'ontologie est basé sur

un graphe de spécialisation des objets. Dans chaque graphe, la distance de l'ontologie doit être caractérisée par le plus court chemin qui fait intervenir un ancêtre commun ou le plus petit généralisant, connectant potentiellement deux objets à travers des descendants communs. Parmi les travaux classifiés sous cette bannière on peut citer :

Similarité de « Wu & Palmer »

[74] a défini une mesure de similarité entre concepts pour la traduction automatique entre l'anglais et le chinois. Leur mesure s'applique à un domaine conceptuel qui correspond à un point de vue donné. La similarité est définie par rapport à la distance qui sépare deux concepts par rapport à leur PPG ainsi que la racine de la hiérarchie. La similarité entre C1 et C2 est :

$$ConSim(C1, C2) = \frac{2 * N3}{N1 + N2 + 2 * N3}$$

Plus formellement cette mesure devient :

$$ConSim(C1, C2) = \frac{2 * deph(C)}{deph_c(C1) + deph_c(C2)}$$

Où C est le PPG de C1 et C2 (en nombre d'arcs), depth(C) est le nombre d'arcs qui séparent C de la racine et depthc(Ci) avec i le nombre d'arcs qui séparent Ci de la racine en passant par C. Cette mesure a l'avantage d'être simple à implémenter et d'avoir d'aussi bonnes performances que les autres mesures de similarité [73].

7.2. Approche basée sur les nœuds

Ces approches adoptent une nouvelle mesure en termes de la mesure entropique de la théorie de l'information [73].

La probabilité P(.) pour l'identification de l'utilisation d'une classe ou de ses descendants dans un corpus désigne l'information de la classe. On définit l'entropie d'une classe par la formule suivante :

$$E(C) = -\log(P(C))$$

Similarité de « Resnik 1995 »

La notion du contenu informationnel (CI) a été initialement introduite par [74] qui a prouvé qu'un objet (mot) est défini par le nombre des classes spécifiées et que la similarité sémantique entre deux concepts est mesurée par la quantité de l'information qu'ils partagent. Pour évaluer la pertinence d'un objet il faut calculer le contenu informationnel. Le contenu informationnel est obtenu en calculant la fréquence de l'objet dans le corpus (Wordnet). La formule proposée par Resnik est définie par:

$$Sim(X, Y) = Max[E(CS(X, Y))] - Max[-\log(P(CS(X, Y)))]$$

Où CS(X, Y) représente le concept le plus spécifique (qui maximise la valeur de similarité) qui subsume (situé à un niveau hiérarchique plus élevé) les deux concepts X et Y dans l'ontologie. Cette mesure est un peu sommaire car elle ne dépend que du concept le plus spécifique.

Similarité de « Lin »

Lin a défini une mesure de similarité légèrement différente de celle de Resnik :

$$Siml(X,Y) = \frac{2 * \log(P(AC(X,Y)))}{\log(P(X)) + \log(P(Y))}$$

Cette mesure utilise une approche hybride qui combine deux sources de connaissances différentes (Thesaurus, corpus).
En plus, elle représente la similarité comme degré probabiliste de chevauchement des concepts descendants de X et Y. Les travaux de [76] ont évalué cette mesure à travers une expérience qui utilise des sujets humains pour évaluer la similarité entre 30 paires de noms, il en ressort que cette méthode offre une amélioration significative.

7.3. Approche hybride

Ces approches sont fondées sur un modèle qui combine entre les approches basées sur les arcs (distances) en plus du contenu informationnel qui est considéré comme facteur de décision.

Similarité de « Jiang et Conrath »

Pour remédier au problème présenté au niveau de la mesure de Resnik [75] a apporté une nouvelle formule qui consiste à combiner l'entropie (contenu informationnel) du concept spécifique à ceux des concepts dont on cherche la similarité (combine entre les techniques basées sur les arcs et les techniques basées sur les nœuds qui consistent à compter les arcs afin d'améliorer les résultats par des calculs basés sur les nœuds). La mesure adoptant cette méthode est basée sur la combinaison d'une source de connaissance riche (thesaurus) avec une source de connaissance pauvre (corpus). Notons que cette formule est définie par l'inverse de la distance sémantique.

$$Sim\ (X,Y) = \frac{1}{Distance(X,Y)}$$

Sachant que la distance entre X et Y est calculée par la formule suivante :

$$Distance\ (X,Y) = E(X) + E(Y) - (2.E(CS(X,Y)))$$

7.4. Approches basées sur l'espace vectoriel

Dans le domaine de la recherche de l'information, les modèles de l'espace vectoriel sont largement adoptés [77]. Ces approches utilisent un vecteur caractéristique, dans un espace dimensionnel, pour représenter chaque objet et calculent la similarité en se basant sur la mesure de cosine ou la distance euclidienne. Le modèle de l'espace vectoriel est employé pour un arrangement des objets complexes en les représentant comme des vecteurs de k-dimensions.
La définition de la similarité entre deux vecteurs d'objets est obtenue par leurs contenus internes. Parmi les approches citées dans la littérature on peut citer :

Similarité de « Jaccard »

La mesure de similarité de Jaccard est définie par le nombre des objets communs divisé par le nombre total des objets moins le nombre d'objets communs :

$$Sim\ j = \frac{X.Y}{||X||_2^2 + ||Y||_2^2 - X.Y}$$

Tels que x et y sont des vecteurs extraits à partir des concepts X et Y. ||x||=Φx i=1 désigne la norme du vecteur x et

$$||X||_2 = \sqrt{\sum_{i=1}^{n} |X_i|}^2$$

Similarité de « Cosinus »

Cette mesure utilise la représentation vectorielle complète, c'est-à-dire la fréquence des objets (mots).
Deux objets (documents) sont similaires si leurs vecteurs sont confondus. Si deux objets ne sont pas similaires, leurs vecteurs forment un angle (X, Y) dont le cosinus représente la valeur de la similarité. La formule est définie par le rapport du produit scalaire des vecteurs x et y et le produit de la norme de x et de y.

$$Simc(X,Y) = \cos(X,Y) = \frac{X.Y}{||X||_2.||Y||_2}$$

Conclusion

Dans ce chapitre nous avons présenté une étude sur les systèmes d'intégration de données hétérogènes qui empêchent une intégration souple des données, les problèmes du traitement des requêtes sur les systèmes basés sur le médiateur et la classification des systèmes d'intégrations.
L'intégration de données hétérogènes de différents types de structuration est donc un volet de recherche intéressant. En effet, la plupart des travaux sur l'intégration traitent des données homogènes, l'intégration des données de différents types structurées, semi-structurées et non structurées reste un sujet récent et assez peu exploré.

Dans le prochain chapitre, nous allons concevoir un système de médiation sémantique à base d'ontologie, intégrant des documents hétérogènes dans le domaine de l'écologie.

CHAPITRE 03

CONCEPTION DU SYSTEME DE MEDIATION SEMANTIQUE

Introduction

Nous avons présenté dans les chapitres précédents les concepts généraux de l'intégration des sources de données hétérogènes et particulièrement le rôle capital que jouent les ontologies dans cette intégration.

Dans ce chapitre, nous concevons un système de médiation sémantique à base d'ontologie dans le domaine de l'écologie, plus précisément la pollution. Pour cela, nous proposons une conception basée sur l'approche LAV pour définir les relations entre le schéma global et les sources de données.

Comme source de données, nous disposons d'un corpus de documents contenant entre autres des livres, des articles, des mémoires de fin d'étude et des cours; tous dans le domaine écologique.

1. Objectifs de notre Système

Nous avons comme objectif de concevoir un système permettant d'une part d'intégrer et de d'offrir un accès unifié aux sources de données hétérogènes, et d'autre part de traiter les conflits syntaxiques et sémantiques à travers une ontologie de domaine.

Pour assurer la flexibilité des sources de données, nous utilisons l'approche LAV qui consiste à définir les schémas des sources locales comme des vues du schéma global ; l'ontologie de domaine étant le schéma global.

Le choix du domaine écologique est surtout motivé par la disponibilité de nombreuses documentations qui présentent différents types d'hétérogénéité (structurelles, syntaxiques sémantiques…). Nous utilisons divers formats de fichiers (pdf, texte, doc…) dans lesquels nous avons décidé de ne traiter que le contenu textuel. Ainsi, un prétraitement sera fait afin d'éliminer les contenus de type multimédia et tout autre contenu différent du texte.

Les principaux objectifs de notre système se résument donc à:

- ✓ Offrir à l'utilisateur une interface lui permettant d'interroger les sources de données de manière uniforme ;
- ✓ Traiter les problèmes d'ambiguïté d'une requête utilisateur à travers l'utilisation de l'ontologie ;
- ✓ Traiter les requêtes au niveau adaptateur.

Pour ce faire, nous présentons dans ce qui suit, l'architecture conceptuelle du système de médiation sémantique.

2. Description des différents niveaux du système de médiation

L'architecture du système de médiation sémantique se compose de trois niveaux que nous décrivons comme suit :

2.1. Niveau Médiateur

Le niveau médiateur constitue le cœur de l'architecture et le point d'entrée du système. Il est représenté par une interface d'interrogation et des vues des sources à intégrer. Pour cela nous avons utilisé l'ontologie 'ONTEPRO' (Ontology for Ecological Problems).

Le médiateur doit offrir les fonctionnalités suivantes :

- Localiser les sources de données pertinentes.
- Accepter les requêtes utilisateurs.
- Réécrire (décomposer) et optimiser les requêtes.
- Envoyer les plans d'exécution à faire et les exécuter par les adaptateurs sur des différentes sources.
- Recomposer les résultats des adaptateurs.

Le médiateur comporte un moteur de requêtes et une base de connaissance spécifique au domaine du médiateur. Cette base se compose de l'ontologie 'ONTEPRO' et des descriptions du contenu des sources de données. La sortie de médiateur est un ensemble de plans de requêtes qui seront traduits par les adaptateurs afin d'obtenir la réponse à la requête de l'utilisateur.

Nous présentons dans la prochaine section la conception de l'ontologie 'ONTEPRO'.

2.2. Niveau Source de données

Ce niveau est constitué des sources qui doivent être intégrées. Nous disposons d'une source contenant des données hétérogènes selon la structure, la syntaxe et la sémantique.

Sémantique	Syntaxique	Structurelle
Deux mots peuvent s'écrire de manières différentes et avoir la même signification	Les abréviations et les erreurs de syntaxe doivent être prises en compte.	Les différentes structures que peuvent prendre les documents
Par exemple le terme « pollution » est désigné dans certains documents par « dégradation »	Par exemple si l'utilisateur entre atmospher, le système doit considérer atmosphérique à la place	Par exemple la structure d'un mémoire n'est pas la même que celle d'un article

Tableau 3.1: *Types d'hétérogénéités utilisées*

2.3. Niveau Adaptateur

Un adaptateur est placé au-dessus de la source afin d'y accéder, d'extraire les données pertinentes et de présenter les résultats au médiateur. Il représente l'interface de communication entre la source de données et le médiateur.
Il dispose des composants suivants :

- Module de traitement de requêtes (évaluation des sous-requêtes en termes de schéma de la source).
- Module de traitement sémantique à l'aide du dictionnaire Wordnet.

De plus, les adaptateurs transforment les réponses aux requêtes en des réponses conformes au schéma global du médiateur.

Il s'agit donc d'une interface permettant d'interroger un corpus de documents grâce à un langage normalisé.

L'adaptateur cache l'hétérogénéité au médiateur et il peut être intelligent et donc effectuer des optimisations spécifiques aux sources.

La figure suivante illustre l'architecture conceptuelle de notre système de médiation sémantique dédié au domaine de l'écologie.

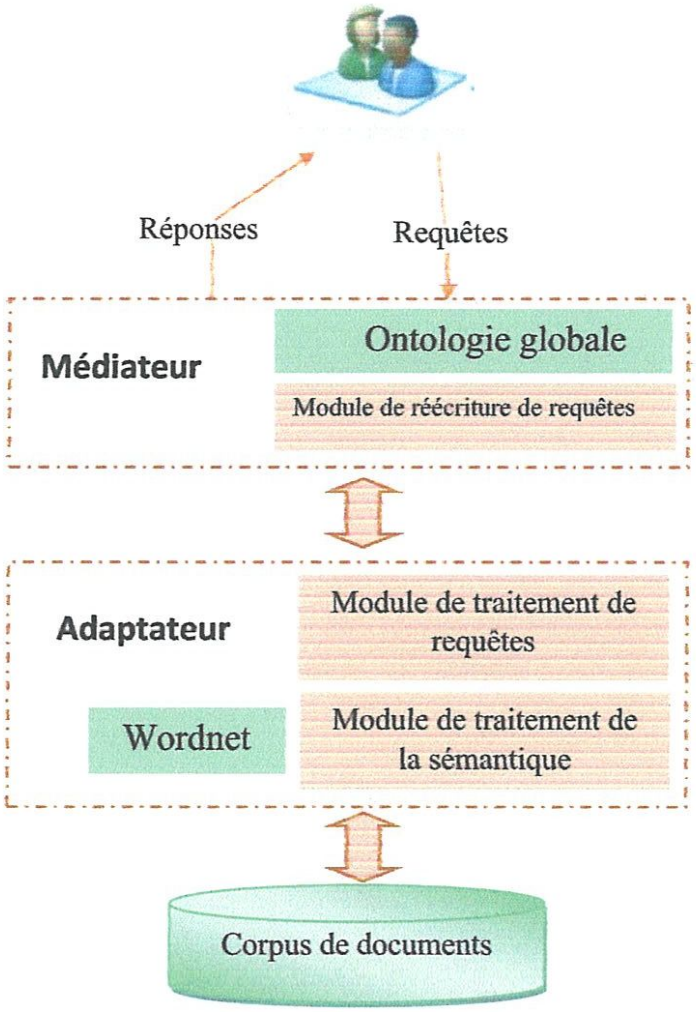

Figure 3.15: Architecture conceptuelle du système de médiation sémantique

3. Conception de l'ontologie globale

Comme nous l'avons indiqué dans les précédentes sections, notre choix de domaine s'est porté sur les problèmes liés à l'environnement. Avant de concevoir cette ontologie, il est important de faire une brève présentation du domaine.

3.1. Présentation du domaine des problèmes écologiques

La pollution est la dégradation d'un écosystème par l'introduction, généralement humaine, de substances ou de radiations altérant de manière plus ou moins importante le fonctionnement de cet écosystème. [79]

Aujourd'hui, nous savons que la pollution est une réalité très présente dans notre environnement. Cette pollution est une conséquence de notre mode de vie et elle existe sous plusieurs formes différentes. Elle peut être provoquée par certaines causes naturelles telle que les volcans, cependant les causes sont en grande partie d'origine humaine. En effet, les voitures émettent des gaz d'échappement, les incinérateurs polluent et les produits comme les pesticides ne respectent pas l'environnement.

Nous distinguons plusieurs types de pollution qui peuvent se résumer en trois grandes classes : pollution physique, pollution chimique, pollution biologique.

- La pollution physique :

Les pollutions d'ordre physique peuvent être divisées en : i) Pollution thermique due à l'augmentation de la température (production d'électricité) et ii) Pollution radioactive due à l'émission de rayonnements X provenant du soleil et de certaines industries.

- La pollution chimique :

Elle peut être entre autres : i) Atmosphérique caractérisé par une altération des niveaux de qualité et de pureté de l'air. Elle est généralement causée par les gaz toxiques des véhicules et des usines ; ii) Organique : Les matières organiques sont les principaux polluants des milieux aquatiques. Elles proviennent des déchets domestiques (ordures ménagères, excréments), agricoles (lisiers) ou industriels (papeterie, tanneries, abattoirs, laiteries, huileries, sucreries...), lorsque ceux-ci sont rejetés sans traitement préalable ; iii) hydrocarboné : c'est un type de pollution qui surtout les eaux de mer et les océans. Elle est due aux hydrocarbures provenant des industries et des activités d'extraction de pétrole.

- La pollution biologique :

Les pollutions biologiques sont en général : i) Agricole : elle est due aux mauvaises pratiques de fertilisation (engrais) ; ou ii) Génétique : c'est le phénomène d'introduction de gènes modifiés ou étrangers à une espèce ou à une variété dans une autre variété ou dans une population sauvage (Organismes génétiquement modifiés).

La pollution entraine des conséquences très néfastes sur l'homme ainsi que sur l'environnement :

- Impacts sur l'environnement :

Une grande diversité de polluants et de problèmes atmosphériques perturbent le fonctionnement naturel de l'écosystème.

- La qualité du sol et de l'eau est fortement touchée par les pluies acides et les polluants qui y contribuent.
- La santé de la faune est mise en danger par les polluants tel que le mercure qui détériorent la qualité des habitats et des aliments.
- Divers polluants, y compris l'ozone troposphérique, nuisent à la santé et à la productivité de la végétation.
- Ils peuvent grandement nous empêcher de voir clairement le paysage avoisinant, ce qui est important pour les parcs et les aires protégées.
- Les changements climatiques résultant du rejet des gaz à effet de serre peuvent modifier considérablement les écosystèmes globaux, les conditions atmosphériques, le niveau de l'eau et la vie sur Terre.
- La destruction de la couche d'ozone qui nous expose de plus en plus aux rayons ultra-violets du soleil.

- Les migrations de certaines espèces, voire leur extinction si elles ne peuvent s'adapter à leur nouvel environnement biophysique

➢ Impacts sur l'homme :

De nombreuses études permettent aujourd'hui d'affirmer que même à des niveaux faibles, la pollution a des effets néfastes sur notre santé. Selon l'Organisation Mondiale de la Santé, « trois millions de personnes meurent chaque année sous l'effet de la pollution atmosphérique, soit 5 % des 55 millions de décès annuels dans le monde ». En effet, les polluants peuvent entrainer diverses maladies chez l'homme (maladies respiratoires et cardio-vasculaires, fibrose des poumons, développement des cellules cancérogènes, accroissement de la sensibilité aux allergies,…).

La figure suivante illustre quelques causes de la pollution de l'environnement.

Figure 3.16: *Domaine de la pollution*

3.2. Construction de l'ontologie 'ONTEPRO'

Il existe une multitude de méthodes pour construire une ontologie. Cependant, il n'y a pas de méthode universelle sur les principes qui doivent guider la modélisation d'une ontologie. La plupart des méthodes utilisées consistent soit à réutiliser une ontologie déjà existante ou alors à la construire à nouveau en faisant appel à des experts du domaine ou une documentation approfondie.

Dans le cas présent, nous avons essayé de construire une nouvelle ontologie dans le domaine de l'écologie.

Pour ce faire, nous suivons la démarche suivante : faire la spécification des besoins et procéder à la conceptualisation de l'ontologie.

3.2.1. Spécification des besoins

Cette phase nous permet de déterminer la portée et les fonctionnalités de l'ontologie :
L'ontologie va couvrir le domaine des problèmes écologiques. Son utilisation doit permettre de traiter les hétérogénéités sémantiques dans un système d'intégration. Cette ontologie doit répondre aux requêtes posées par l'utilisateur en prenant en compte la sémantique de celles-ci. Elle doit pouvoir renseigner un utilisateur sur les questions concernant ce domaine et permettre à l'utilisateur de connaitre les différents types de pollutions, leurs causes ainsi que leurs impacts sur l'homme et sur l'environnement. Finalement, l'ontologie sera maintenue par des experts en environnement.
Dans ce qui suit nous procédons à la conceptualisation de la dite ontologie.

3.2.2. Conceptualisation

Cette phase nous a permis de construire notre ontologie en effectuant des entretiens avec les experts du domaine et de en s'aidant de documentation dans le domaine (articles, mémoires, livres, etc.). Nous avons choisi d'utiliser la langue anglaise pour construire l'ontologie par ce que plus tard nous utiliserons le dictionnaire Wordnet pour calculer les proximités entre les termes de l'ontologie et ceux du corpus. La version en anglais de ce dictionnaire étant plus riche et plus efficace que celle en français.

Etape 1 : Enumérer et identifier les termes importants de l'ontologie.
L'étude faite sur le domaine des problèmes écologiques nous a permis de dégager la liste de termes suivants : *Ecological problems* est le concept le plus général. *Types*, *Factors*, *Impacts* sont des concepts généraux de niveau supérieur.

Etape 2 : Définir les classes et la hiérarchie des classes
Si nous considérons par exemple les deux concepts de l'ontologie globale : *Types* et *Factors*

Concepts	Description	Sous concepts
Types	Décrit les grandes classes des différents types de pollution qui peuvent exister	-Biological_pollution -Chimical_pollution -Physical_pollution
Factors	Décrit les facteurs qui provoquent la pollution de l'environnement	-Naturals -Provok

Tableau 3.2: *Classes et hiérarchie des classes*

Etape 3 et 4 : Définir les propriétés des classes et leurs relations
Par exemple la description des propriétés de la classe de la super classe et la classe de la classe

Concepts	Propriétés	Type
Acid_rain	Consequences :'vast_scale_dieback_of_coniferous_forests'	String
	Origin : 'In_the_80s_some_stations_in_the_US_and_Europe_have_measured_a_sharp_increase_in_the_acidity_of_precipitation'	String
	Reason : 'Sulphur_dioxide_and_nitrogen_oxides'	String
Sterility	Name : 'female_infertility_and_male_infertility'	String
	Symptom : 'Disorders_implanting_and_uterine_abnormalities_and_endometriosis'	String
	Reason : 'weight_problem_and_smoking_and_stress'	String

Tableau 3.3: *Extrait des propriétés des classes*

En ce qui concerne les relations entre concepts (classes ou sous classes), nous avons utilisé les liens de généralisation/spécialisation qui sont les liens IS-A des modèles sémantiques auxquels sont associés un mécanisme d'héritage des propriétés de la classe d'objets génériques par la classe d'objets spécialisés.

Relation IS-A	Concept source	Concept cible	Description
Généralisation	Gas	Global_warming	Est une relation entre le polluant et l'impact qu'il peut avoir sur l'environnement
Spécialisation	Pesticide	Chemical_Fertilizer	Est une relation entre un polluant et le facteur qui le provoque

Tableau 3.4: *Relations IS-A entre les concepts de l'ontologie*

Etape 5 : créer les instances des classes dans la hiérarchie
Par exemple la classe *on_environment* de la super classe *Impacts* comporte les instances : *Acid_rain, Degradation_of_aquatic_fauna_and_flora, Degadation_of_the_ozone_layer, Global_warmimg*. Et la classe *Provok* de la super classe *Factors* comporte les instances : *Chimical_fertilizer, CO2, genetically_modified_organism, Hamful_gas, NO2, Waste.*

*Etape 6 :*C'est la dernière étape de la conceptualisation. Elle consiste à lier les informations des étapes précédentes sous un modèle conceptuel.

La figure suivante illustre la hiérarchie des concepts et les relations entre eux.

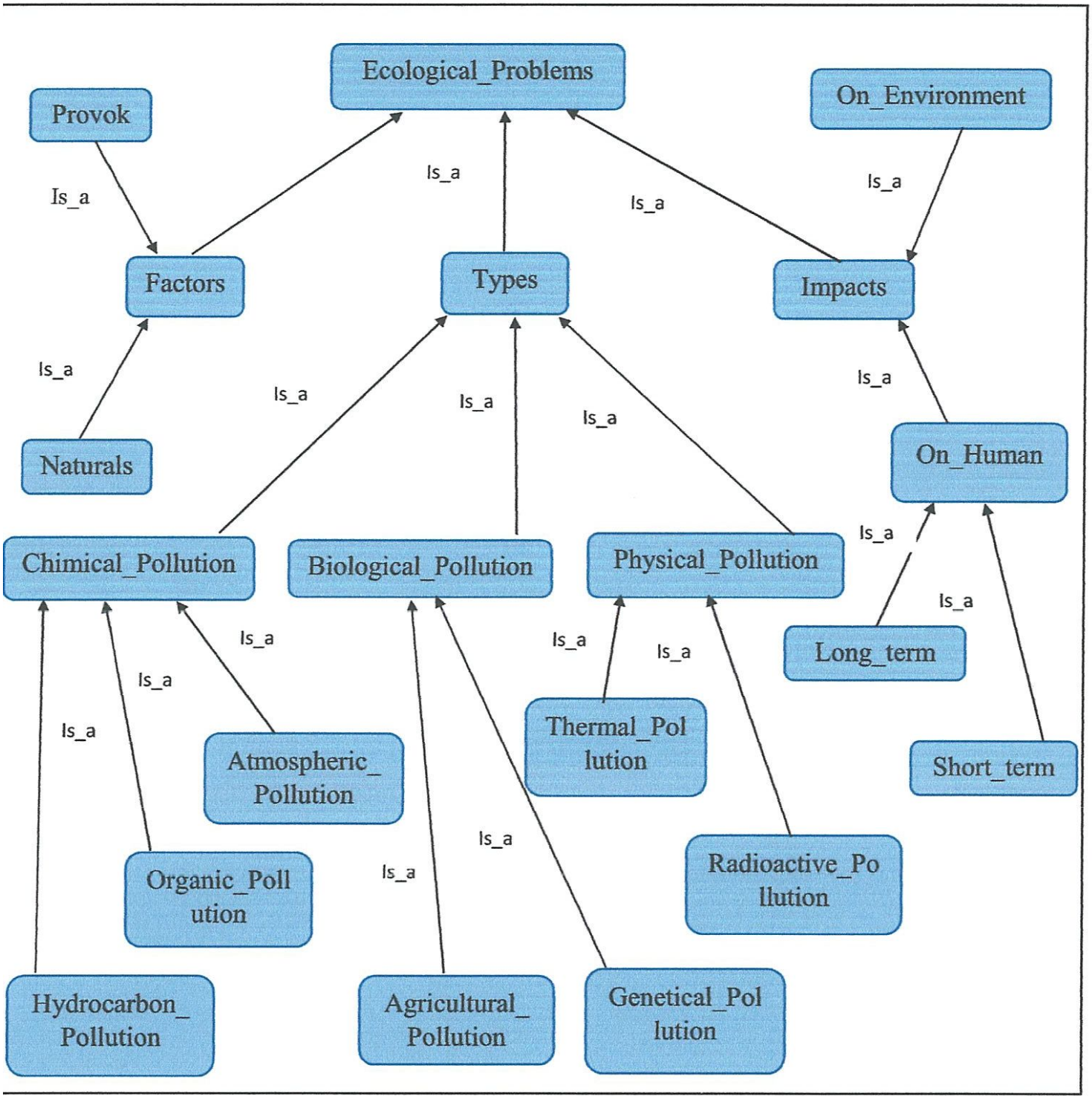

Figure 3.17: *Modèle conceptuel de l'ontologie 'ONTEPRO'*

4. Processus d'intégration sémantique

Dans cette section, nous expliquons comment se déroule le traitement d'une requête utilisateur par les différents niveaux de médiateur. Les réponses aux requêtes sont calculées par réécriture en termes de vues. Ces vues abstraites de données stockées dans les sources de données sont des requêtes exprimées en fonction de l'ontologie.

4.1. Formulation des requêtes et reconstruction des résultats

Comme nous l'avons expliqué en section 2 de ce chapitre, cette fonction est assurée par le médiateur qui formule les requêtes en termes de schéma global c'est-à-dire de l'ontologie

globale. Après traitement, c'est également par le biais du médiateur que les réponses à la requête envoyées par l'adaptateur sont reconstruites pour donner le résultat.
Les algorithmes Formulation et Reconstruction suivants illustrent respectivement les procédures de formulation de requêtes et de reconstruction de résultats.

Algorithme **Formulation ;**

Début
1. Q := Requête ;
2. D := Décomposer (Q) ;
3. Envoyer_a_adaptateur (D) ;

Fin.

Explication :

- **Etape1** : réception de la requête par le médiateur : dans cette étape, la requête a été écrite en langage d'ontologie SPARQL via une interface utile et utilisable.
- **Etape2** : décomposition de la requête en sous-requêtes : dans le cas d'une requête composée de plusieurs conditions, le médiateur décompose cette requête en un ensemble des requêtes atomiques (contient une seule condition). La décomposition est faite par élimination des opérateurs de conjonctions entre conditions.
- **Etape3** : envoi des sous-requêtes à l'adaptateur : le système est muni d'un adaptateur de corpus de documents hétérogènes. Son rôle est d'effectué l'interrogation de corpus après la réception des sous requêtes du médiateur. Dans la sous-section suivante, nous expliquons le processus de traitement de requêtes au niveau de l'adaptateur.

Algorithme **Reconstruction ;**
N : nombre de sous résultats ;
M : médiateur ;
S : Sous-résultats
R : résultat ;
Début
1. Pour i=1 à N faire
   M : envoyer (Si) ;
   Fin pour ;
2. R : reconstruire (Si) ;
3. Envoyer (R) ;

Fin.

Explication :

- **Etape1 :** l'adaptateur envoie les sous résultats au médiateur. Selon un nombre N de sous résultats, l'adaptateur les envoie au médiateur pour la reconstruction de résultat en terme de vocabulaire de schéma global (l'ontologie)

- **Etape2** : les sous résultats sont reconstitués selon le vocabulaire du schéma global du médiateur. La reconstruction est faite dans un premier temps par l'application des conjonctions entre sous résultats selon la requête initiale de l'utilisateur. Et dans un second temps, une liste de documents pertinents selon le degré d'apparition de termes de l'ontologie dans les documents.
- **Etape3** : le médiateur envoie le résultat final à l'utilisateur via l'interface unifiée sous forme d'une table de documents avec les différentes correspondances entre les concepts similaires de document et celles de l'ontologie.

4.2. Traitement des requêtes :

Cette étape est exécutée au niveau de l'adaptateur qui traduit les requêtes exprimées par le langage SPARQL en des termes à comparés avec les indexes de documents du corpus. La première phase du processus consiste alors à faire une indexation de tous les documents de la source de données. La procédure de l'indexation est bien détaillée dans le premier chapitre.
A l'issu de l'indexation, toute valeur, tout concept et tout terme de la source de données est représenté dans un ensemble de mots. Ensuite, il s'agit de supprimer tous les mots vides et d'appliquer une procédure de lemmatisation à tous les mots considérés dans la comparaison avec le contenu de l'ontologie.

- Mot vide : Un mot vide désigne tout élément d'un texte n'ayant pas de référence concrète ou notionnelle dans la réalité et dont le rôle est d'assurer la liaison ou la cohérence entre les principales parties d'un texte. Il s'agit des articles, des conjonctions, des adjectifs non qualificatifs, des prépositions et de la ponctuation.
- Lemmatisation : Il s'agit d'une procédure qui permet de ramener un mot portant des marques de flexion (par exemple, la forme conjuguée d'un verbe) à sa forme de référence (dite lemme).

Autrement dit, il s'agit, dans le cadre d'une analyse de texte par ordinateur ou d'un traitement automatique de la langue de manière générale, de reconnaître un mot par rapport à sa forme de base, et cela, quelle que soit la forme sous laquelle il apparaît dans un texte.
Exemple : lemmatisation (environnemental) = environnement.

Afin de mesurer la similarité sémantique entre les données de la source et le contenu de l'ontologie, nous utilisons une procédure d'appariement entre les concepts de l'ontologie et les indexes de documents. Cette procédure est fondée sur des critères sémantiques notamment les mesures de similarité sémantique présentées en Section 7 du chapitre 2.
Dans le cas présent, nous utiliserons la similarité de Wu et Palmer basée sur la profondeur des concepts dans la hiérarchie de l'ontologie globale. Le calcul de cette similarité s'effectue selon la formule suivante :

$$ConSim(C1, C2) = \frac{2 * deph(C)}{deph_c(C1) + deph_c(C2)}$$

Cette mesure présente l'avantage de la rapidité du temps d'exécution. Elle donne un score de similarité entre deux concepts, ainsi, un seuil est arbitrairement fixé à partir duquel les deux concepts peuvent être considérés comme similaires.
Il existe un package java permettant de calculer automatiquement la similarité de Wu et Palmer. Nous le présenterons dans le chapitre 4.

En plus des mesures de similarité, nous utiliserons le dictionnaire de synonyme Wordnet pour éliminer les conflits sémantiques entre les concepts durant le processus de l'intégration.

Finalement, les documents retournés sont tries par ordre décroissant d'appartenance de concepts de l'ontologie dans ces documents et les documents moins pertinents sont les documents qui contient des concepts similaires (synonymes) aux ceux de l'ontologie.

Conclusion

Ce chapitre a été consacré à la présentation de notre système de médiation sémantique à base d'ontologie de corpus de documents hétérogènes.

Après avoir dégagé quelques objectifs de notre système, nous en avons présenté l'architecture générale tout en expliquant le rôle de chaque niveau du système et nous avons conçu l'ontologie globale et présenter notre processus d'intégration sémantique.

Dans le prochain chapitre, nous allons mettre en œuvre l'implémentation de notre système de médiation sémantique basé sur une ontologie et dédié au domaine des problèmes écologiques.

CHAPITRE 04

EXPERIMENTATION ET EVALUATION DES RESULTATS

Introduction

Ce chapitre aborde la mise en œuvre du système de médiation sémantique dont nous avons effectué l'étude conceptuelle dans le chapitre précédant.

Pour mener à bien ce travail, nous avons utilisé le langage de programmation JAVA avec son IDE NetBeans ainsi que le logiciel Protégé 2000 pour la création et la manipulation de l'ontologie globale dédiée au domaine de la pollution.

Le présent chapitre décrit donc d'une part les outils logiciels utilisés pour mettre en œuvre le système et les détails de l'implémentation, et d'autre part une évaluation des performances du système ainsi qu'une étude comparative avec quelques travaux similaires.

1. Les outils logiciels utilisés

Pour l'implémentation d'un système de médiation sémantique basée sur une ontologie dédiée au domaine des problèmes écologiques, nous avons utilisé les outils logiciels suivants :

- NetBeans version 8.1 : un environnement de développement (IDE) basé sur les langages JAVA, C, PHP et bien d'autres
- Wordnet 2.1 : une ontologie linguistique et un dictionnaire de synonymes en anglais
- Protégé 2000 version 4.3 : un logiciel de développement d'ontologies
- Adobe PDF Illustrator : un logiciel graphique utilisé pour créer des images

1.1. NetBeans

NetBeans est un environnement de développement intégré (EDI), placé en *open source* par Sun Microsystems en juin 2000. En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C++, JavaScript, XML, Groovy, PHP et HTML de façon native ou avancée (comme Python ou Ruby). L'IDE NetBeans s'enrichit à l'aide de plugins et y comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web) [Web2]. NetBeans est disponible sous Windows, Linux, Solaris, Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Development Kit JDK est requis pour les développements en Java.

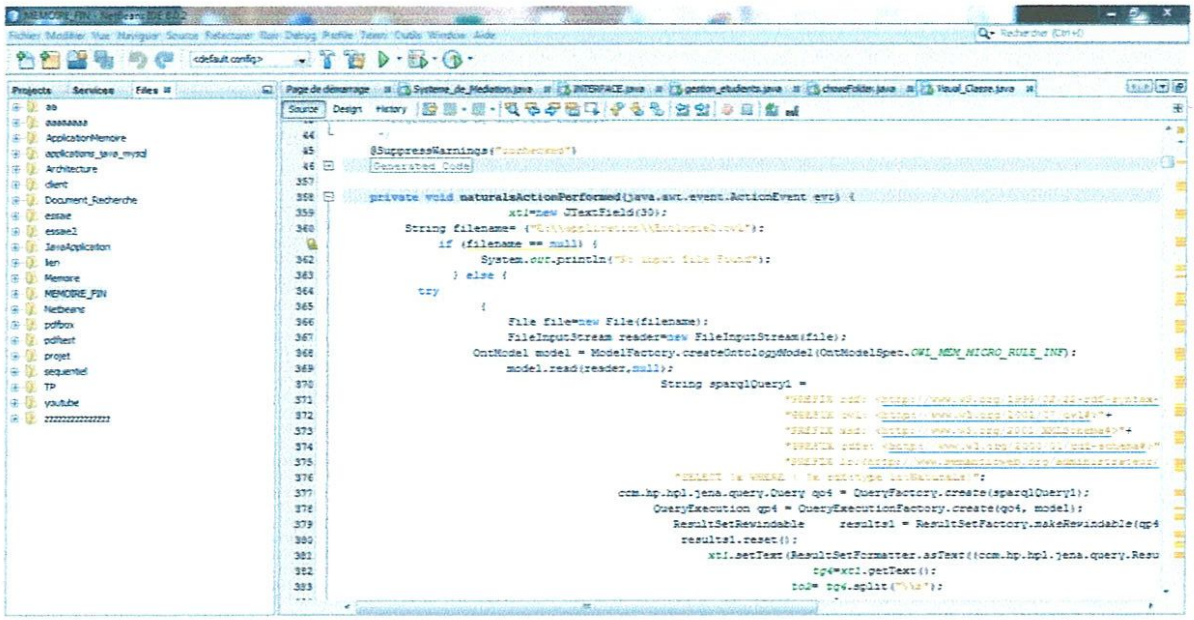

Figure 4.18: Interface de NetBeans

1.2. Protégé 2000

Protégé est un éditeur d'ontologies distribué en open source par l'université en informatique médicale de Stanford. Protégé n'est pas un outil spécialement dédié à OWL, mais un éditeur hautement extensible, capable de manipuler des formats très divers. Le support d'OWL, comme de nombreux autres formats, est possible dans protégé grâce à un plugin dédié.
Protégé est un outil employé par les développeurs et des experts de domaine pour développer des systèmes basés sur les connaissances (Ontologies).

Des applications développées avec Protégé sont employées dans la résolution des problèmes et la prise de décision dans un domaine particulier. Protégé est aussi une plate-forme extensible, grâce au système de plug-ins, qui permet de gérer des contenus multimédias, interroger, évaluer et fusionner des ontologies, etc. L'outil Protégé possède une interface utilisateur graphique (GUI) lui permettant de manipuler aisément tous les éléments d'une ontologie : classe, méta-classe, propriété, instance,...etc. Protégé peut être utilisé dans n'importe quel domaine où les concepts peuvent être modélisés en une hiérarchie des classes.

Protégé permet aussi de créer ou d'importer des ontologies écrites dans les différents langages d'ontologies tel que : RDF-Schéma, OWL, DAML, OIL, ...etc. Cela est rendu possible grâce à l'utilisation de plugins qui sont disponibles en téléchargement pour la plupart de ces langages.

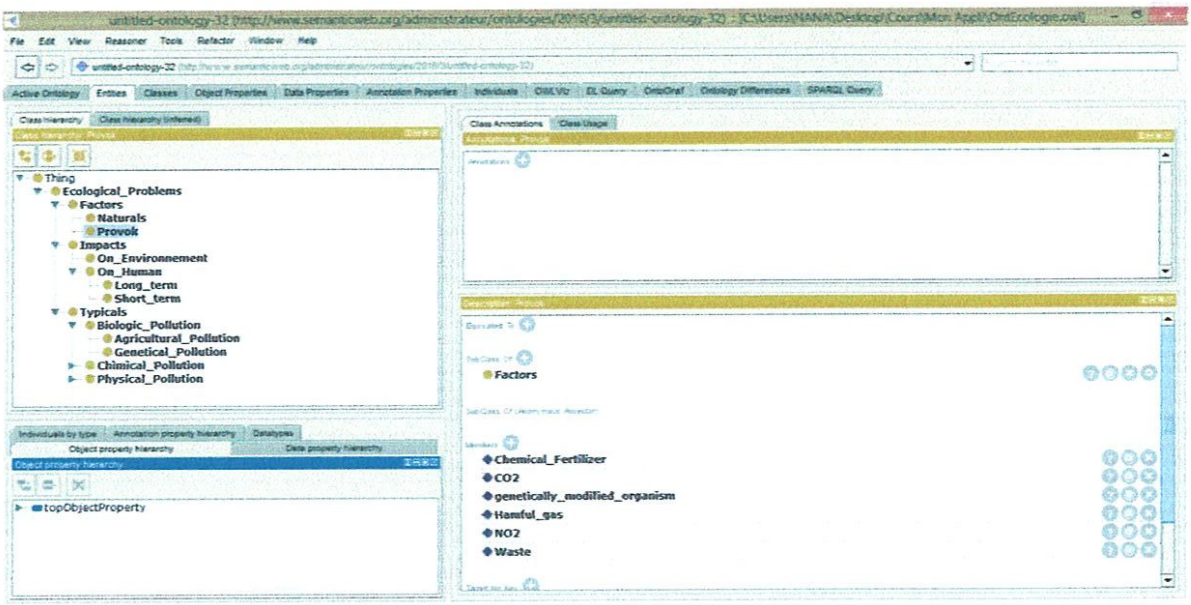

Figure 4.19: Interface de Protégé 2000

1.3. WordNet

WordNet est une base de données lexicale développée par des linguistes du laboratoire des sciences cognitives de l'université de Princeton [80]. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise. Des versions de WordNet pour d'autres langues existent, mais la version anglaise est cependant la plus complète à ce jour. La base de données ainsi que des outils sont disponibles gratuitement. Par rapport aux outils fournis, un développeur peut aussi accéder à la base de données à partir des interfaces disponibles pour de nombreux langages de programmation.

WordNet est distribué sous une licence libre, permettant de l'utiliser commercialement ou à des fins de recherche. La dernière version distribuée en avril 2013 est la 3.1, elle est par ailleurs consultable en ligne.
WordNet est cependant limité par rapport à certaines fonctionnalités [80]:

- Il est beaucoup moins élaboré pour les verbes, qui sont organisés en un système hiérarchique beaucoup plus « plat » avec moins de niveaux d'imbrication, où on passe très rapidement d'un concept spécialisé (le sens *operate, run* du verbe *running*, par exemple) à un concept très général (*control, command*).
- Il n'y a aucune catégorisation hiérarchique définie pour les embranchements des adjectifs et des adverbes.
- Il ne précise pas l'étymologie, la prononciation, les formes de verbes irréguliers et ne contient que des informations limitées sur l'usage des mots. La contrepartie de son importante couverture est que WordNet est très précis dans le sens des définitions, par exemple, le verbe « to give » (donner) n'a pas moins de 44 sens.

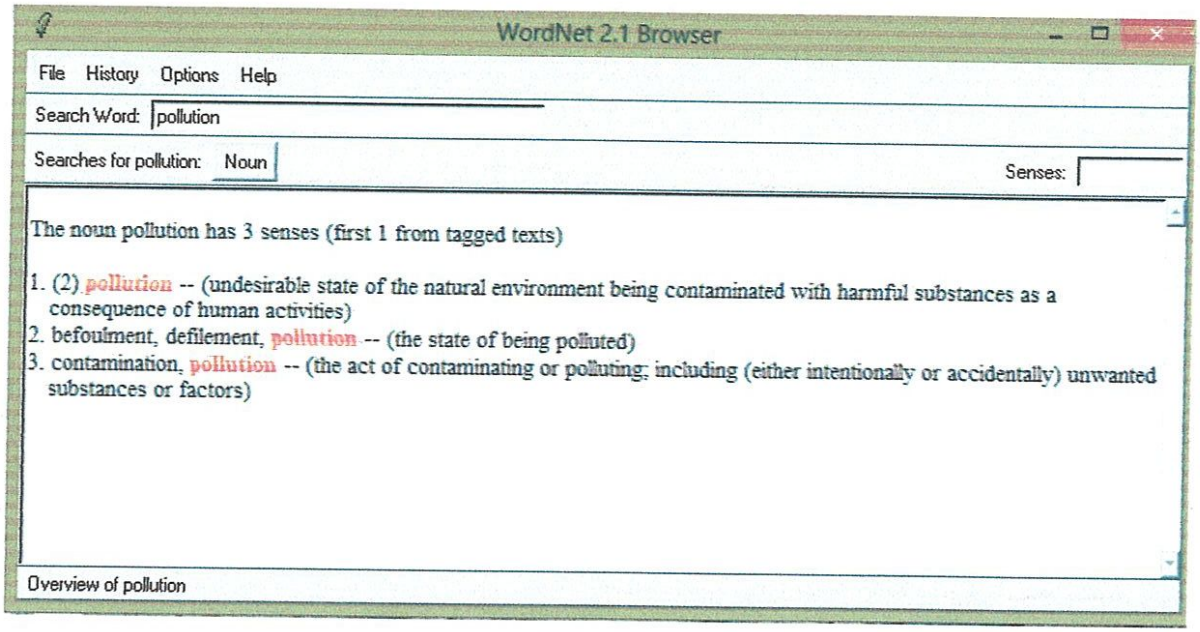

Figure 4.20: Interface de WordNet

1.4. Adobe PDF Illustrator

Adobe Illustrator permet à l'utilisateur de travailler avec une grande variété de types de fichiers pour créer des images utilisées généralement dans la publicité, manifestations artistiques ou de publications. Le logiciel est basé sur le vecteur graphique, un système qui utilise des représentations mathématiques pour définir les lignes et les ombres d'une image, plutôt que des fichiers d'images photographiques [Web 11].

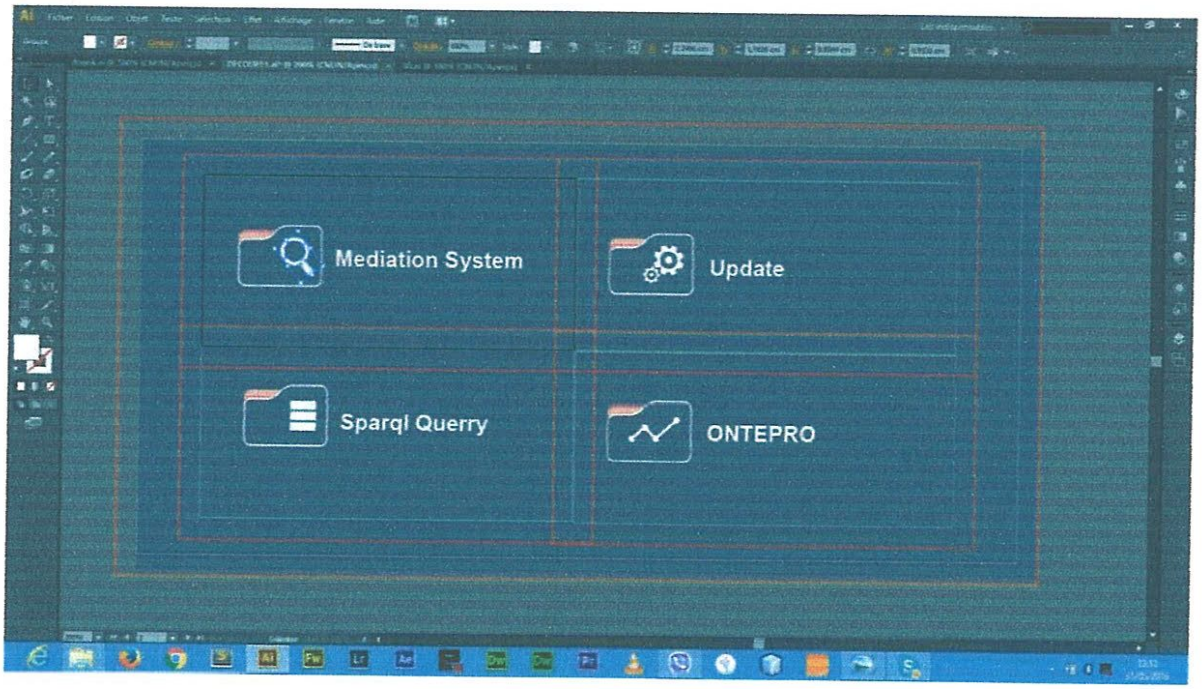

Figure 4.21: Interface de l'Adobe Illustrator

2. Les package utilisés

Un package est une archive contenant des données et/ou des programmes ainsi que les informations nécessaires à une installation correcte de ceux-ci sur le système. Un package est

constitué d'un et un seul fichier, ce n'est pas un exécutable, il est pris en charge par un programme dédié, le gestionnaire de package. Il permet de fournir un mécanisme pour l'extension de la plateforme et ainsi fournir la possibilité de développer des fonctionnalités qui ne sont pas fournies en standard dans les outils de développement.

Dans ce cadre, nous avons utilisé les packages suivants :

- OWLViz : une librairie intégrée dans protégé 2000 pour afficher le graphe de l'ontologie.
- Jena : une librairie intégrée à java pour manipuler l'ontologie via l'IDE
- WS4J et JawJaw : qui sont des librairies intégrés aussi à java pour calculer la similarité sémantique
- Lucene : bibliothèque intégrée à java pour indexer les fichiers PDFs.

2.1. OWLViz

OWLViz est conçu pour être utilisé avec l'éditeur Protégé-OWL . Elle permet des hiérarchies de classes dans une ontologie OWL à visualiser et progressivement navigué, permettant une comparaison de la hiérarchie des classes affirmée et la hiérarchie des classes inférée. OWLViz intègre avec l'éditeur Protégé-OWL, en utilisant le même schéma de couleurs de sorte que les classes primitives et définies peuvent être distinguées, les changements calculés à la hiérarchie des classes peuvent être clairement visibles [Web 12].

OWLViz permet de visualiser la structure de la classe d'une ontologie comme graphe orienté mettant en évidence les relations parent-enfant (is-a).

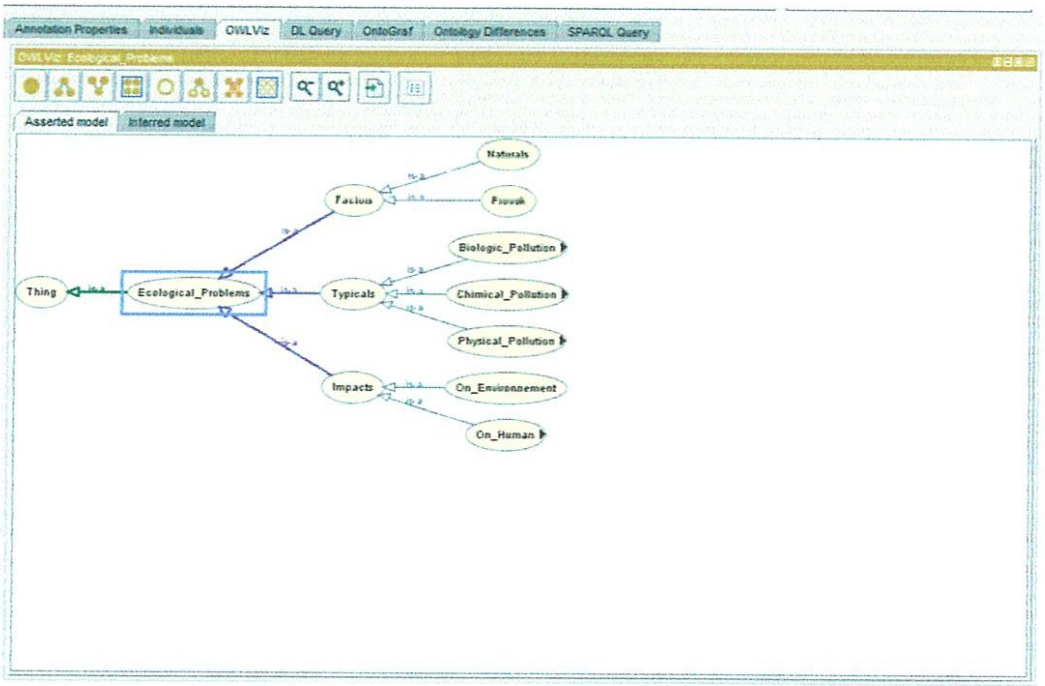

Figure 4.22: Interface d'OWLViz

2.2. Jena

Jena est une open source Web sémantique cadre pour Java . Il fournit une API pour extraire des données et écrire à RDF graphiques. Les graphiques sont représentés comme un «modèle»

abstrait. Un modèle peut provenir des données à partir de fichiers, bases de données, des URL ou une combinaison de ceux-ci. Un modèle peut également être interrogé par SPARQL 1.1.

Jena peut être téléchargé à partir du site : http://jena.apache.org (Apache Jena binary distribution). Tous les fichiers Jar nécessaires à la compilation et à l'utilisation de Jena sont dans le fichier Lib. Ils doivent tous être présents dans le CLASSPATH pour la compilation ou l'exécution de programmes utilisant Jena [Web 6].

2.3. WS4J

WS4J (Wordnet similarity for Java) fournit des API pour plusieurs algorithmes de similarité sémantique. En théorie, toute instance de WordNet peut être utilisée pour calculer le score d'apparentement tant qu'il implémente une interface de base de données lexicale [Web 13].

2.4. Lucene

Lucene est une bibliothèque open source écrite en Java qui permet d'indexer et de chercher du texte. Il est utilisé dans certains moteurs de recherche [Web 14]. C'est un projet de la fondation Apache mis à disposition sous licence Apache. Il est également disponible pour les langages Ruby, Perl, C++, PHP, C#.

3. Les interfaces de l'application

Dans cette section, nous présentons quelques interfaces de notre système de médiation sémantique afin de montrer le bon fonctionnement du système avec la possibilité d'effectuer des manipulations avancées telle que la mise à jour de corpus hétérogènes, l'exploration de schéma de médiateur et l'exécution des requêtes écrites en langage SPARQL. L'architecture de ce système est l'architecture médiateur représentée par trois niveaux (médiateur, adaptateurs et sources) comme nous avons décrit dans le chapitre précédent (section 2).

3.1. Interface principale

La fenêtre principale de l'application s'affiche au lancement de l'écran de démarrage de système de médiation sémantique. Elle offre à l'utilisateur le choix d'effectuer l'intégration de données hétérogènes via le bouton Mediation System. En vue de créer un corpus de documents hétérogènes provenant du Web, nous avons choisi les documents PDF comme format standard, sécurisés, et très utilisables par les internautes par rapport aux autres formats tels que doc, txt.

Par ailleurs, notre système offre des manipulations avancées suivantes :

- Le bouton ONTEPRO permet à l'utilisateur d'observer les classes de l'ontologie globale et leurs instances afin de pouvoir mieux interroger le système de médiation sans ambigüité
- Le bouton Update qui reste inactif jusqu'à la connexion d'un administrateur, permet de mettre à jour le corpus de documents. Chaque nouveau document ajouté est automatiquement indexé.
- Le bouton Sparql Query qui permet à l'utilisateur d'interroger l'ontologie globale à travers des requêtes Sparql.

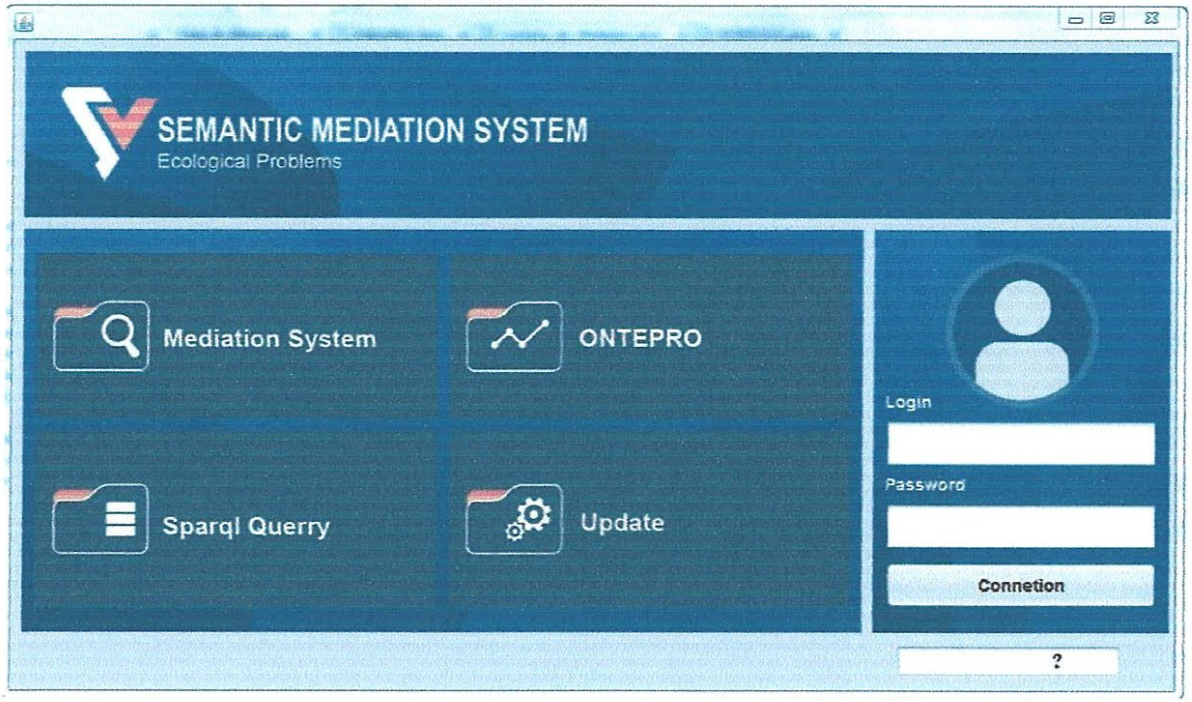

Figure 4.23: Interface principale de système

3.2. Interface d'intégration de documents hétérogènes

Cette interface permet de faire des interrogations sur le corpus de documents à travers le médiateur qui comporte l'ontologie globale. L'utilisateur pose sa requête dans le terme du schéma du médiateur à travers des composants graphiques facilitants la création de requêtes. Dans la figure suivante, nous voulons trouver les documents qui parlent sur le facteur CO2. Les documents trouvés sont obtenus par l'application du processus d'intégration sémantique et ils sont ordonnés par ordre décroissant des scores de similarité.

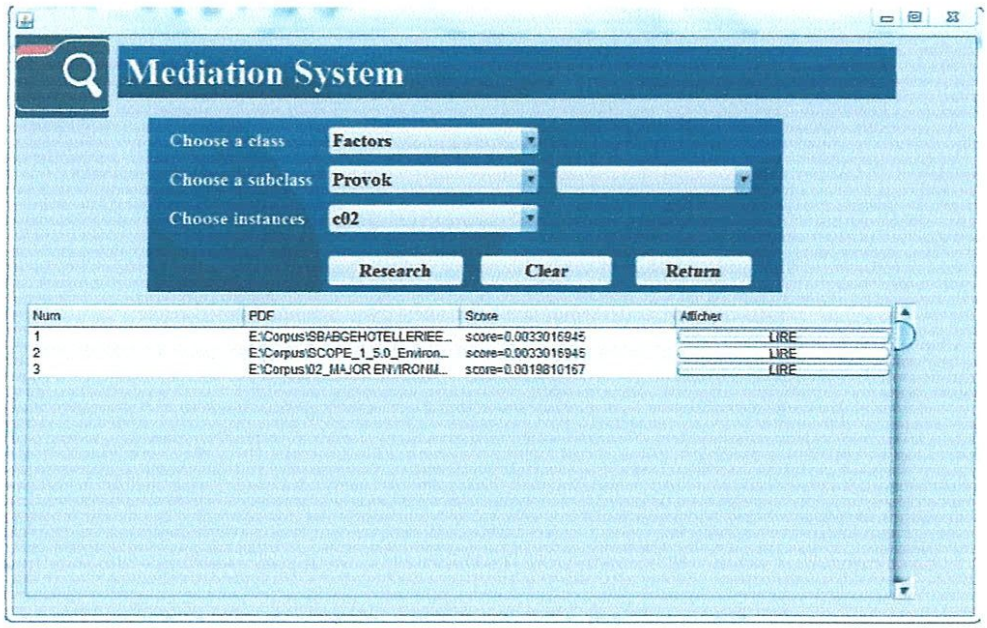

Figure 4.24: Fenêtre d'interrogation du système de médiation

Le traitement de l'hétérogéneité est effectué en mésurant les similarités de Wu et Palmer entre les concepts de la requête et ceux de la source en utilisant le WordNet comme outil d'aide pour la désambiguation.

3.3. Interface d'interrogation avancée de schéma du médiateur

Cette interface représente une manipulation avancée qui permet à un utilisateur d'interroger le schéma du médiateur qui est l'ontologie globale à travers des requêtes Sparql à condition de connaitre la syntaxe de ce langage.

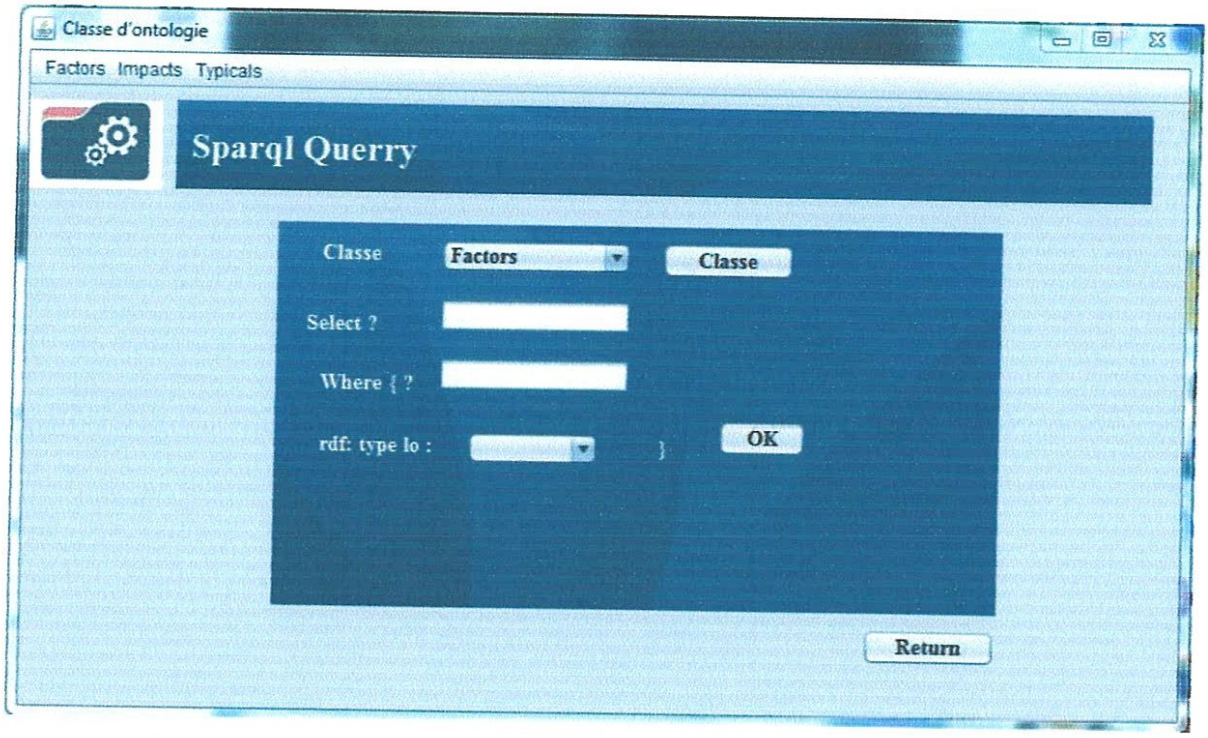

Figure 4.25: *Interface d'interrogation avancée de schéma du médiateur*

De plus, le système de médiation sémantique offre la possibilité de visualiser le contenu de l'ontologie et la hiérarchie des classes à travers l'interface suivante.

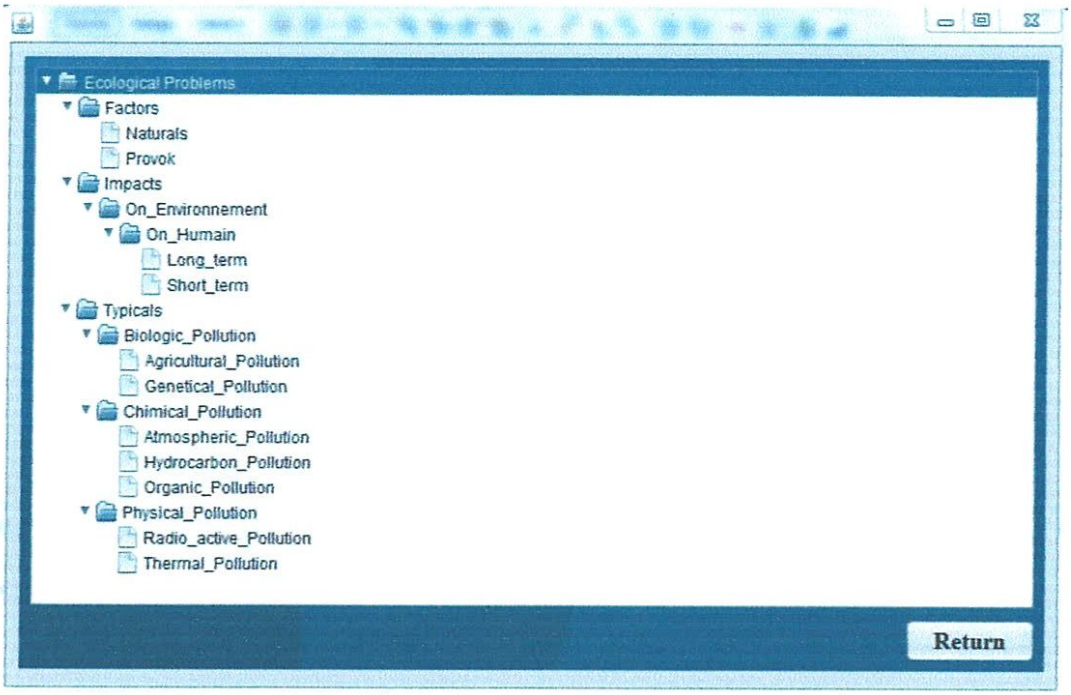

Figure 4.26: *Interface de visualisation de l'ontologie*

3.4. Interface de manipulation avancée de source de données

Afin d'assurer la souplesse et l'extensibilité de la source de données, cette interface permet à un utilisateur d'étendre autant qu'il veut le corpus. Le bouton *Browser* permet de parcourir l'ordinateur et d'ajouter de nouveaux documents au corpus. Nous nous focalisons seulement sur l'ajout d'un nouveau document car cette opération implique l'ajout d'un adaptateur propre pour ce document afin d'être interrogé par l'utilisateur. Les nouveaux documents sont automatiquement indexés à l'actualisation de l'application. Cependant, l'accès à cette interface est réservé aux utilisateurs possédant le droit d'administrateur, pour cela nous disposons d'une base de données dans laquelle sont stockées les noms d'utilisateurs et les mots de passe des administrateurs. La fenêtre de connexion est accessible à partir de l'interface principale.

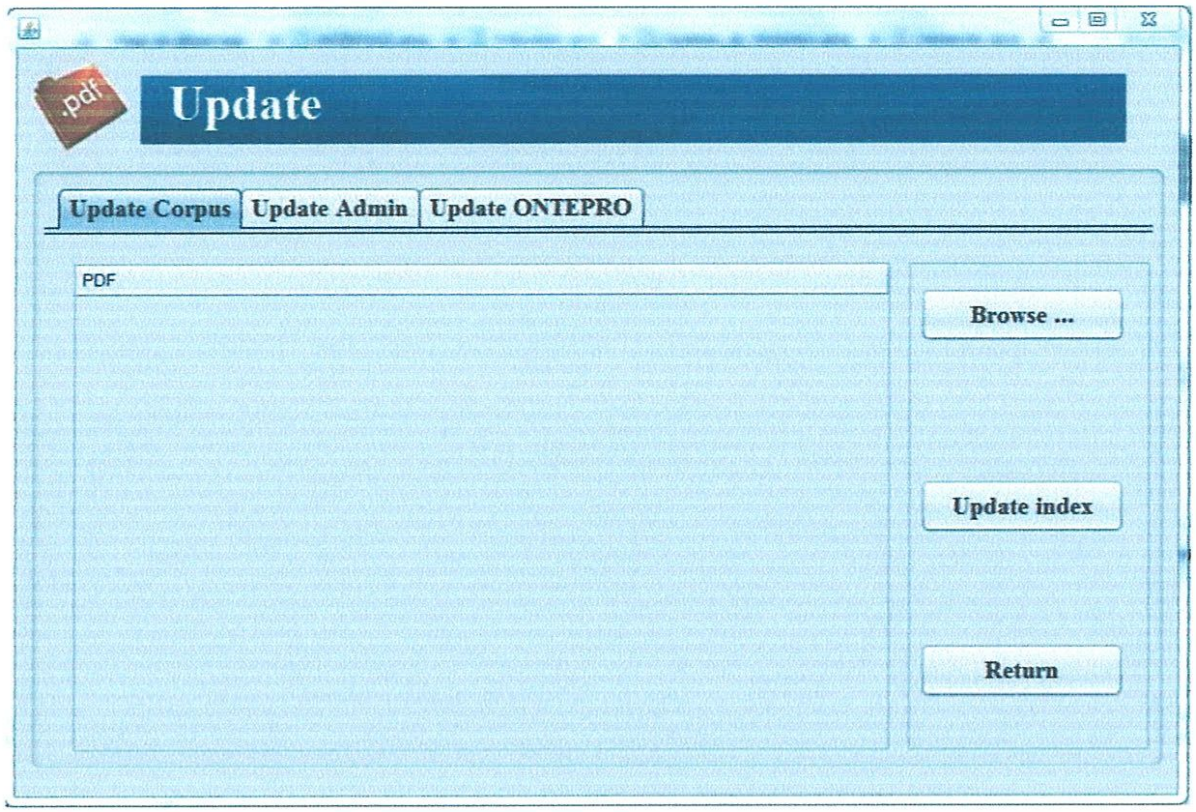

Figure 4.27: *Interface de mise à jour de la source de données*

4. Evaluation de performances

Pour évaluer et comparer les systèmes informatiques, on utilise les critères de performances habituels, c'est-à-dire les mesures du temps de réponse et d'espace mémoire utilisé : plus le temps de réponse est court et l'espace occupé par le système est faible, meilleur est le système. Cependant, d'autres mesures de performance ont été introduites pour évaluer l'efficacité d'un système. Parmi elles, nous pouvons citer la facilité d'utilisation du système et surtout la capacité du système à trouver des documents répondant aux besoins de l'utilisateur (documents pertinents). De façon générale, une mesure d'évaluation appropriée doit estimer la faculté des systèmes à retrouver des documents pertinents et rejeter tous les documents non pertinents. Les mesures de rappel et de précision sont les mesures les plus simples et les plus utilisées. Le détail de calcul de ces deux mesures est présenté dans le chapitre 01, section 5.

Pour cette raison, nous exécutons dix requêtes numéroté de 1 jusqu'à 10 et nous mesurons la Précision et le rappel selon les jugements de l'administrateur qui juge la pertinence de document de corpus. Le résultat est illustré dans le tableau suivant :

Requête	Précision	Rappel
1	0.033	1
2	0.2	0.75
3	0.2	0.857
4	0.333	0.909
5	0.066	1
6	0.333	0.833
7	0.367	1
8	0.2	0.857
9	0.533	0.842
10	0.2	0.857

Tableau 4. 1: *Evaluation de performances*

Un système de recherche parfait aurait fourni des réponses dont la précision et le rappel sont égaux à 1 (c'est-à-dire que l'algorithme trouve la totalité des documents pertinents - rappel - et ne fait aucune erreur - précision). Dans la réalité, les algorithmes de recherche sont plus ou moins précis, et plus ou moins pertinents. Il est possible d'obtenir un système très précis (par exemple un score de précision de 0,99), mais peu performant (par exemple avec un rappel de 0,10, qui signifiera qu'il n'a trouvé que 10 % des réponses possibles).

Nous pouvons déduire au regard du tableau ci-dessus que :

- Les mesures absolues de performances ont une portée limitée. Cette limitation est due, d'une part, à l'impossibilité de définir précisément la notion de pertinence, et d'autre part, à l'impossibilité d'obtenir des corpus de grande taille totalement et correctement étiquetés.
- L'exécution de ces différentes requêtes par le système retourne 89,05% des documents pertinents.

5. Comparaison avec les travaux similaires

Dans cette section, nous effectuons une comparaison de notre système de médiation avec quelques travaux similaires réalisés antérieurement. Le premier travail est celui de Kamar MARRAKCHI qui a soutenu sa thèse de Doctorat en Bio-informatique le 19 Décembre 2012 à l'université Abdelmalek ASSAADI sous l'intitulé : « Une approche hybride pour une intégration sémantique des données biologiques de Pseudomonas». Le deuxième travail est de Fatiha SAÏS qui a soutenu sa Thèse de doctorat en informatique le 07 Décembre 2007 à l'université Paris-Sud de France sous le thème : « Intégration sémantique de données guidée par une ontologie». Et le travail de HOUDA Ahmad qui a soutenu sa Thèse de doctorat en informatique à l'université Joseph Fourier de Grenoble1 sous l'intitulé : « Une approche matérialisée basée sur les vues pour l'intégration de documents XML ». Le choix de ces travaux est basé principalement sur l'utilisation de l'ontologie dans le système d'intégration pour le traitement de l'hétérogénéité de sources de données. Le tableau suivant illustre cette comparaison selon un ensemble de critères.

Critères	Notre travail	Travail de Kamal	Travail de Fatiha SAIS	Travail de HOUDA Ahmad
Domaine	Problèmes écologiques	Données Biologiques de	Risque Alimentaire	Domaine médical
Schéma global	Ontologie de domaine	Ontologie de domaine	Ontologie de domaine	Utilisation de vues
Sources locales	Corpus de documents	Bases de données biologiques accessibles via le Web	Des tableaux de XML	Documents XML
Types d'hétérogénéité	- Des mots qui s'écrivent de manière différentes mais ont le même sens - Les abréviations et erreurs de syntaxe - Les structures des différents documents	- Diversités syntaxiques - Diversité des schémas - Informations au niveau des instances - Diversités des langages de requêtes - Diversités des services	- Informations incomplètes et erronées - Informations ambiguës - Informations non identifiables	- Hétérogénéité schématique - Hétérogénéité sémantique - Hétérogénéité des modèles de données
Dictionnaire sémantique	Wordnet	SD-Core	- Créer son propre dictionnaire - SML (Semantic Markup Language)	Pas précisé
Approche de médiation	Une seule ontologie	Une seule ontologie	Une seule ontologie	Une seule ontologie
Système d'intégration	Approche de médiation sémantique	Approche entrepôts de données	Entrepôts de données	Entrepôts de données
Approche d'intégration	LAV	GAV	Rien à mentionner	GAV
Architecture logicielle	- Module interface - Module médiateur - Module adaptateur - Module sources de données - Mise à jour des sources	- Module schéma des sources - Modules services de données - Module schéma intégrateur - Correspondances sémantiques	Rien à mentionner	- Correspondance entre schémas - Transformation, Extractions, Classement - Stockage - Traitement de requêtes, Interrogation, Mise à jour d'un document XML
Mesure de similarité	Wu et Palmer	Aucun	Jaro-Winkler, utilisation des clusters	Optimisation de requêtes par un P-types
Langage de programmation	Java	Java, Servlet API, XHTML	Delphi	Java

Tableau 4.2: *Comparaison avec trois travaux similaires*

6. Quelques extraits du code source

Dans cette section, nous présentons quelques extraits de codes sources de notre application.

Le premier extrait de code permet d'établir la connexion entre l'ontologie et l'IDE NetBeans, puis exécute une requête Sparql :

```
private void long_termActionPerformed(java.awt.event.ActionEvent evt) {
            xt1=new JTextField(30);
      String filename= ("E:\\application\\Ecologie2.owl");
          if (filename == null) {
                System.out.println("No input file found");
            } else {
        try
                {
          File file=new File(filename);
          FileInputStream reader=new FileInputStream(file);
          OntModel model = ModelFactory.createOntologyModel(OntModelSpec.OWL_MEM_MICRO_RULE_INF);
          model.read(reader,null);
          String sparqlQuery1 =
          "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
           "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
           "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"+
           "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"+
           "PREFIX lo:<http://www.semanticweb.org/administrateur/ontologies/2016/3/untitled-ontology-32#>"+
           "SELECT ?x WHERE { ?x rdf:type lo:Long_term}";
            com.hp.hpl.jena.query.Query qo4 = QueryFactory.create(sparqlQuery1);
            QueryExecution qp4 = QueryExecutionFactory.create(qo4, model);
            ResultSetRewindable results1 = ResultSetFactory.makeRewindable(qp4.execSelect());
            results1.reset();
             xt1.setText(ResultSetFormatter.asText((com.hp.hpl.jena.query.ResultSet) results1));
            tg4=xt1.getText();
```

Le deuxième extrait de code source pour calculer la similarité de Wu et Palmer entre deux concepts :

```
import edu.cmu.lti.ws4j.impl.WuPalmer;
import edu.cmu.lti.ws4j.util.WS4JConfiguration;

public class SimilariteSemantique {
    private static ILexicalDatabase db = new NictWordNet();
    private static double compute(String word1, String word2) {
        WS4JConfiguration.getInstance().setMFS(true);
        double s = new WuPalmer(db).calcRelatednessOfWords(word1, word2);
        return s;
    }
    public static void main(String[] args) {
        String[] words = {"pollution","degradation"};

        for(int i=0; i<words.length-1; i++){
            for(int j=i+1; j<words.length; j++){
                double distance = compute(words[i], words[j]);
                System.out.println(words[i] +" and  " +  words[j] + " = " + distance);
            }
        }
```

Le troisième extrait de code source montre l'indexation des documents du corpus avec Lucene

```
            writer.addDocument(doc);
            System.out.println("Added: " + f);
          } catch (Exception e) {
            System.out.println("Could not add: " + f);
          } finally {
            fr.close();
          }
        }
        int newNumDocs = writer.numDocs();
        System.out.println("");
        System.out.println((newNumDocs - originalNumDocs) + " documents added.");
        queue.clear();
      }
      private void addFiles(File file) {
        if (!file.exists()) {
          System.out.println(file + " does not exist.");
        }
        if (file.isDirectory()) {
          for (File f : file.listFiles()) {
            addFiles(f);
          }
        } else {
          String filename = file.getName().toLowerCase();
          if (filename.endsWith(".pdf")){
            queue.add(file);
          } else {
            System.out.println("Skipped " + filename);
          }
```

Conclusion

Nous avons présenté dans ce chapitre la mise en œuvre de notre système de médiation sémantique à base d'ontologie validant l'étude conceptuelle que nous avons présenté dans le chapitre précédent. De plus, nous avons effectué une évaluation de performance et une étude comparative qui montrent que nous avons réussi d'implémenter un système de médiation sémantique à base d'ontologie dédié au domaine écologique. Ces expérimentations méritent donc d'être poursuivies pour améliorer notre système.

CONCLUSION GENERALE

Conclusion générale

L'intégration sémantique de données consiste à offrir une interrogation unifiée et homogène des données hétérogènes afin d'éliminer les conflits entre ces données. Actuellement, les ontologies sont utilisées dans divers domaines pour représenter explicitement les connaissances d'un domaine particulier. Dans les systèmes d'intégration, une ontologie joue le rôle de schéma global sur lequel l'utilisateur pose ses requêtes, nos propositions sont basées essentiellement sur l'utilisation des ontologies afin de pouvoir résoudre les hétérogénéités entre les sources de données.

Nous avons décrit dans la première partie de ce mémoire un état de l'art sur d'une part la recherche d'information et particulièrement la recherche sémantique d'information ainsi que le rôle des ontologies dans cette recherche. Et de l'autre part, les systèmes d'intégration existants, les différentes mesures de similarité sémantique entre concepts avec les différents types d'hétérogénéités qui peuvent exister entre les sources de données.

La deuxième partie du mémoire décrit notre contribution qui consiste à développer un système de médiation sémantique à base d'ontologie et d'effectuer des manipulations avancées. Ce système permet d'accéder à des données hétérogènes sans avoir à les migrer vers un schéma global, pour cela choisir cette solution est préférable dans des systèmes où l'information évolue rapidement. Nous avons implémenté ce système pour le domaine des problèmes écologiques.

Enfin, pour évaluer notre système, nous avons réalisé une évaluation de performance par le calcul de mesure de rappel et précision et une étude comparative qui montrent le bon fonctionnement de système d'intégration sémantique.

Comme perspectives nous envisageons d'enrichir cette architecture par d'autres sources couvrant tout le domaine de l'environnement, nous souhaitons également traiter le contenu multimédia des documents et distribués dans le Cloud ainsi que le Big Data.

Références

Bibliographie

[1] ABBASSI MEFTAH « Un modèle de reformulation des requêtes pour la recherche d'information sur le web », Mémoire de Magister, 2013.

[2] ABBAS Nacira « Vers une extension sémantique de l'analyse formelle de concepts : application à la recherche d'informations », Mémoire de Magister, 2014.

[3] Peter Ingwersen, « Information retrieval interaction », Taylor Graham Publishing, London, UK, 1992.

[4] L.Maisonnasse « Les supports de vocabulaire pour les systèmes de recherche d'information orientés précision : application aux graphes pour la recherche d'information médicale », Thèse de doctorat en informatique, Université Joseph Fourier-Grenoble I, France, 2008.

[5] T.D.Cao. « Exploitation du web sémantique pour la veille technologique », Thèse de doctorat en informatique, Université de Nice-Sophia Antipolis-UFR, Sciences, 2006.

[6] M.K.Khelif. « Web sémantique et mémoire d'expérience pour l'analyse du transcriptome », Thèse de doctorat en informatique, Université de Nice-Sophia Antipolis-UFR, Sciences, 2006.

[7] V.Uren, P.Cimiano, J.Iria, S.Handschuh, M. Vargas-Vera, E.Motta, F.Ciravegna. « Semantic annotation for knowledge management: requierements and a survey of the state of the art », In web semantics, Volume 4, Issue 1, pp 14-28, 2006.

[8] K.Ottens « Un système multi-agents adaptatif pour la construction d'ontologie à partir de textes », Page 14, Octobre 2007.

[9] R.Neches, R and P.Velardi. « An analysis of ontology-based query expansion strategies », Proceeding of the workchop on adaptive Text Extraction and Mining, Croatia.

[10] T. Gruber «A translation approach to portable ontology specifications: Knowledge Acquisition », pages 199–220, 1993.

[11] M. Charhad « Modèles de Documents Vidéo basés sur le Formalisme des Graphes Conceptuels pour l'Indexation et la Recherche par le Contenu Sémantique », Pages 24-25, Novembre 2005.

[12] R. Baeza-Yates and R. A. Ribeiro-Neto «Modern Information Retrieval», New York: ACM Press; Harlow England: Addison-Wesley, cop., 1999.

[13] G. Salton «A comparison between manual and automatic indexing methods. Journal of American Documentation», pages 61–71, 1971.

[14] G. Salton and M. McGill « Introduction to Modern Information Retrieval ». McGraw-Hill, New York, 1983.

[15] S. Robertson and K. Sparck Jones « Relevance weighting for search terms », Journal of The American Society for Information Science, pages 129–146, 1976.

[16] M. E. Maron and J. L. Kuhns « On relevance, probabilistic indexing and information retrieval », J. ACM, pages 216–244, 1960.

[17] T. Berners-Lee, J. Hendler, O.Lassila « The semantic web », Scientific American, 2001.

[18] N. Guarino, P. Giaretti « Ontologies and knowledge bases: Towards a terminological clarification », In Towards Very Large Knowledge Bases. N. J. I. Mars, Ed., IOS Press: 25-32, 1995.

[19] N. Aussenac-Gilles, B. Biébow, N. Szulman « Revisiting Ontology Design: a method based on corpus analysis », Proc of KAW'2000, Juan-Les-Pins (F), Oct 2000, Lecture Notes in Artificial Intelligence Vol 1937, Springer Verlag, pages 172-188, 2000.

[20] G. Van Heijst, A. Schreiber, B. Wielinga « Using explicit ontologies in KBS development », Int. J. of Human-Computer Studies, pages183–292, 1997.

[21] A. Gomez-Perez, M. Fernandez-Lopez, O. Corcho « Ontological Engineering », Springer Verlag, 2003.

[22] M. Kiffer, G. Laussen, J. Wu « Logic Foundations of Object- Oriented and Framebased Languages », J. ACM, pages 741-843, 1995.

[23] J.F. Sowa « Conceptual Graphs: Information Processing », In Mind and Machine. Reading, Addison Wesley, 1984.

[24] Fouzia Amourache « construction d'une ontologie pour l'annotation des CVs/offres d'emploi », Mémoire de Magister, Université Mentouri-Constantine, 2008.

[25] D. Fensel, I. Horrocks, F. Van Harmelen, S. Decker, M. Erdmann, M. Klein « OIL in a nutshell. In Proceedings of European Knowledge Acquisition Workshop », EKAW 2000, Springer-Verlag LNAI 1937, pages 1-16, Juan-les-Pins, France, 2000.

[26] J. Hendler et D.L. McGuinness « The DARPA Agent Markup Language», IEEE Intelligent Systems- Trends and Controversies, pages 67-73, Janvier 2000.

[27] I. Horrocks, P. F. Patel-Schneider « Reducing OWL Entailment to Description Logic Satisfiability», In D. Fensel, K. Sycara et J. Mylopoulos, editors, 2nd International Semantic Web Conference (ISWC'03), volume (2870) of Lecure Notes in Computer Science, pages 17-29, Sanibel Island, Floride, USA, 20-23 Octobre 2003 Springer Verlag.

[28] V. Haarslev et R. Möller «RACER System Description», Proceedings of the First International Joint Conference on Automated Reasoning table of content,. Pages 701-706, 2001.

[29] S. Bechhofer, I. Horrocks, C. Goble and R. Stevens « OILEd: a Reason-able Ontology Editor for the Semantic Web », In Proceedings of KI2001, Joint German/Austrian conference on Artificial Intelligence, September 19-21, Vienna. Springer Verlag LNAI Vol. 2174, pages 396-408. 2001.

[30] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer and Wenke, D « OntoEdit: Collaborative Ontology Engineering for the Semantic Web », In Proceedings of the International Semantic Web Conference2002 (ISWC 2002), Sardinia, Italia, June 2002.

[31] Farquhar, R. Fikes and J. Rice « The Ontolingua Server: A Tool for Collaborative Ontology Construction », In Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop, Banff, Alberta, Canada, 44.1-44.19, 1996.

[32] B. Swartout, P. Ramesh, K. Knight and T. Russ « Toward Distributed Use of Large-Scale Ontologies », In Symposium on Ontological Engineering of AAAI, Stanford, California, March, 1997.

[33] Natalya F. Noy and Deborah L. McGuinness « Ontology Development 101: A Guide to Creating Your First Ontology », Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.

[34] Christophe Brouard and Jian-Yun Nie. Reliefs « Un système pour le filtrage adaptatif de documents textuels basé sur la notion de resonance », In INFORSID, pages 267-278. 2001.

[35] Boubekeur F « Contribution à la définition de modèles de recherche d'information flexibles basés sur les CP-Nets », thèse de doctorat en informatique, Université Paul Sabatier, 2008.

[36] Hernandez N « Ontologie de domaine pour la modélisation du contexte en recherche d'information », thèse de doctorat en informatique, Université Paul Sabatier, 2006.

[37] Daoud M « Accès personnalisé à l'information : approche basée sur l'utilisation d'un profil utilisateur sémantique dérivé d'une ontologie de domaines à travers l'historique des sessions de recherche », thèse de doctorat en informatique, Université Paul Sabatier, 2009.

[38] N.D.Y. Kompaoré « Fusion de systèmes et analyse des caractéristiques linguistiques des requêtes: vers un processus de RI adaptatif», Thèse de doctorat en informatique, Université Paul Sabatier de Toulouse, 2008.

[39] A. Y. Levy, A. Rajaraman, and J. J. Ordille « The World Wide Web as a collection of views: Query processing in the information manifold », Proceedings of the International Workshop on Materialized Views: Techniques and Applications (VIEW'1996), pages 43–55, June 1996.

[40] Lahmar Fatima « Une approche Hybride d'intégration de sources de données hétérogènes dans les datawarehouses », thèse de doctorat, université Mentouri de Constantine, 2011.

[41] Nguyen Xuan D « Intégration de bases de données hétérogènes par articulation à priori d'ontologies : application aux catalogues de composants industriels », Thèse de doctorat, ENSMA/Université de Poitiers, 2006.

[42] A. Levy « Logic-based techniques in data integration », In J. Minker, editor, Logic Based Artificial Intelligence, Kluwer Academic Publishers, 2000.

[43] Z. Ives « Efficient Query Processing for Data Integration », A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy University of Washington, 2002.

[44] N. Ashish, C. Knoblock « Semi-automatic Wrapper Generation for Internet Information Sources » In Second IFCIS International Conference on Cooperative Information Systems (CoopIS), Charleston, SC, 1997.

[45] G. Wiederhold « Mediators in the architecture of future information systems », IEEE Computer, vol.25, n°3, page 38-49, 1992.

[46] M.-S. Hacid and C. Reynaud « L'intégration de sources de données » La revue I3: Information - Interaction - Intelligence, Vol5 n°1, 2005.

[47] S. S. Chawathe, H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. D. Ullman, and J. Widom « The tsimmis project: Integration of heterogeneous information sources », Proceedings of the 10th Meeting of the Information Processing Society of Japan, pages 7–18, Mars 1994.

[48] P. Vassiliadis, A. Simitsis, P. Georgantas, M. Terrovitis, and S. Skiadopoulos « A generic and customizable framework for the design of etl scenarios » Information Systems, page: 492–525, 2005.

[49] B. Alexe L. Chiticariu R. J. Miller W. Tan. MUSE « Mapping Understanding and deSign by Example » Proceedings of International Conference on Data Engineering (ICDE) 2008.

[50] Carey, M. J, Haas, L. M., Schwarz, P. M,Arya, M., Cody, W.F., Fagin, R., Flickner, M., Luniewski, A., Niblack, W., Petkovic, D., Thomas, J., Williams, J.H., Wimmers, E.L., Towards « heterogeneous multimedia information systems: The Garlic approach », In Proc. of the 5th Int, Workshop on Research Issues in Data Engineering – Distributed Object Management (RIDE-DOM'95), pages 124–131, IEEE Computer Society Press, 1995.

[51] M. R. Genesereth, A. M. Keller, and O. M. Duschka. Infomaster « an information integration system », In ACM SIGMOD International Conference on Management of Data, pages 539–542, 1997.

[52] G.-L. G. Gomez « Construction automatisée de l'ontologie de systèmes médiateurs » Application à des systèmes intégrant des services standards accessibles via le Web, PhD thesis, Universit´e Paris XI Orsay, 2005.

[53] Friedman M. & Weld D. S. (1997) « Efficiently executing informationgathering plans » In 15th International Joint Conference on Artificial Intelligence, page 785-791, Nagoya. Japan.

[54] Etzioni O. & Weld D. (1994) « A Softbot-Based Interface to the Internet » Communications of the ACM. Vol. 37(7), pages 72-76.

[55] Mena E. & Kashyap V. & Sheth A. & Illarramendi « A OBSERVER: An approach for query processing in global information systems based on interoperation across pre-existing ontologies », In 4th Int. Conf. on Cooperative Information Systems, page 14-25. Bruessels. Belgium, 1996

[56] Koffina,I., Serfiotis, G., Christophides, V « Foundations for Information Integration: A State of the Art », Sixth Framework Programme, 2005.

[57] P. McBrien, A. Poulovassilis « Data integration by bi-directional schema transformation rules », In 19th International Conference on Data Engineering, ICDE'03, March 5 - March 8, 2003

[58] Xu, L., Embley, W « Combining the Best of Global-as-View and Local-as-View for Data Integration », In Information Systems Technologies and Its Applications - ISTA. 123–136, 2004.

[59] D. Beneventano, S. Bergamaschi, S. Castano, A. Corni, R. Guidetti, G. Malvezzi, M. Melchiori, and M. Vincini « Information integration: The MOMIS project demonstration », In The VLDB Journal, pages 611–614, 2000.

[60] H. Wache, T. V¨ogele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. H¨ubner « Ontology-based integration of information - a survey of existing approaches » Proceedings of the International Workshop on Ontologies and Information Sharing, pages 108–117, August, 2001.

[61] Stuckenschmidt H., Wache H., Vogele T., Visser U (2000) « Enabling technologies for interoperability », In Ubbo Visser and Hardy Pundt, editors, Workshop on the 14th International Symposium of Computer Science for Environmental Protection, pages 35–46, Bonn, Germany.

[62] Heflin J., Hendler J., Luke S (1999) « Shoe: A knowledge representation language for internet applications », Technical CS-TR-4078, Institute for Advanced Computer Studies, University of Maryland.

[63] Goasdoué F., Lattes V., Rousset M (1999) « The use of carin language and algorithms for information integration: The picsel project », International Journal of Cooperative Information Systems (IJCIS), 9(4):383 – 401.

[64] Jose Blakeley « Data access for masses through OLE DB », in proceeding of the ACM SIGMOD International Conference on Management of data, New York, 1996, vol. 2 of ACM SIGMOD Record, pp. 161-172, ACM Press.

[65] Yigal Arens, Chun-Nan Hsu, and Craig A « Knoblock. Query processing in the SIMS information mediator », In Advanced Planning Technology. AAAI Press, California, USA, 1996.

[66] Razika DRIOUCHE « Proposition d'une architecture d'intégration des applications d'entreprise basée sur l'interopérabilité sémantique de l'EbXML et la mobilité des agents », thèse de doctorat, université MENTOURI DE CONSTANTINE, 2007

[67] E. Mena, V. Kashyap, A. Sheth, and A. Illarramendi « Observer: An approach for query processing in global information systems based on interoperability between preexisting ontologies », In Proceedings 1st IFCIS International Conference on Cooperative Information Systems (CoopIS '96). Brussels, 1996.

[68] Cheng Hian Goh « Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Sources », Phd, MIT, 1997.

[69] H. Wache, Th. Scholz, H. Stieghahn, and B. K¨onig-Ries « An integration method for the specification of rule–oriented mediators », In Yahiko Kambayashi and Hiroki Takakura, editors, Proceedings of the International Symposium on Database Applications in Non-Traditional Environments (DANTE'99), pages 109–112, Kyoto, Japan, November, 28-30 1999.

[70] Heiner Stuckenschmidt, Holger Wache, Thomas V¨ogele, and Ubbo Visser « Enabling

technologies for interoperability », In Ubbo Visser and Hardy Pundt, editors, Workshop on the 14th International Symposium of Computer Science for Environmental Protection, pages 35–46, Bonn, Germany, 2000. TZI, University of Bremen.
[71] T.T.Dang NGOG « Fédération de données semi-structurées avec XML », Thèse soutenue à l'université de Versailles, Juin 2003.

[72] Rada R., Mili H., Bicknell E., & Blettner M. (1989) « Development and application of a metric on semanticnets », IEEE Transaction on Systems, Man, and Cybernetics, 19(1):17--30.

[73] D. Lin « An information-theoretic definition of similarity», In Proceedings of 15th International Conference On Machine Learning, 1998.

[74] Resnik P « Using information content to evaluate semantic similarity in a taxonomy», In Proceedings of the 14th International Joint Conference on Artificial Intelligence, Montreal, 1995.

[75] [Jiac 97] J. Jiang et D. Conrath « Semantic similarity based on corpus statistics and lexical taxonomy », In Proceedings of International Conference on Research in Computational Linguistics, Taiwan, 1997.

[76] [Mil 93] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller « Introduction to WordNet: An On-line Lexical Database », Cognitive Science Laboratory, Princeton University, Princeton, Technical Report, 1993.

[77] [Sal 83] G.Salton and M. J.McGill « Introduction to modern information retrieval », McGraw-Hill. New York, 1983.

[78] A .Aggoune, A. Bouramoul, M-K. Kholladi « Big Data Integration: A Semantic Mediation Architecture Using summary », In International Conference on Advanced Technologies for Signal and Image Processing, ATSIP'2016. Tunisia. IEEE Explore Tunisia, 2016

[79] Organisation de coopération et de développement économiques « principes et concepts environnementaux », Paris 1995.

[80] François-Régis Chaumartin, « Wordnet et son écosystème : un ensemble de ressources linguistiques de large couverture », Montréal, Canada, Avril 2007.

Webographie

[Web1]http://www-igm.univ mlv.fr/~dr/XPOSE2009/Le%20Web%203.0/concepts.html#vocabulaires (visité le 10/02/2016).

[Web 2] https://fr.wikipedia.org/wiki/Th%C3%A9saurus (visité le 10/02/2016).

[Web 3] http://blog.sparna.fr/2013/12/07/ontologie-thesaurus-taxonomie-web-de-donnees/ (visité le 09/02/2016).

[Web 4] http://theses.ulaval.ca/archimede/fichiers/24629/ch02.html (Visité le 29/02/2016).

[Web 5] https://fr.wikipedia.org/wiki/Pr%C3%A9cision_et_rappel (visité le 29/02/2016).

[Web 6] www.info.univ-angers.fr/pub/genest/fichiers/m1 ws/ws chap6 (visité le 02/06/2016).

[Web 7] B. McBride « RDF Vocabulary Description Language 1.0: RDF Schema », W3C Recommendation, 2004. http://www.w3.org/TR/rdf-schema/ (Visité le 29/01/2016).

[Web 8] OWL. Web Ontology Language Reference. W3C Recommendation, 2004. http://www.w3c.org/TR/owl-ref/. (Visité le 08/02/2016).

[Web 9] André Tricot. "La prise de conscience du besoin d'information: une compétance documentaire fantôme". Mise à jour le 08 Octobre 2005. In André Tricot. Site André Tricot pages perso Orange, [En ligne]. http://www.andre.tricot.pagesperso-orange.fr/tricot-besoinInformation.pdf (visité le 10/02/2016).

[Web 10] Jian-Yun Nie, "introduction à la RI", 2012. In cours IFT6255 A 2012 recherche d'information, site de l'université de Montréal [en ligne]. http://www.iro.umontreal.ca/nie/introduction.hml. (Visité le 22/02/2016).

[Web 11] http://coursz.com/cours-illustrator-pdf.html (Visité le 07/06/2016).

[Web 12] http://protegewiki.stanford.edu/wiki/OWLViz (Visité le 07/06/2016).

[Web 13] https://code.google.com/p/ws4j (Visité le 07/06/2016).

[Web 14] https://fr.wikipedia.org/wiki/Lucene (Visité le 05/06/2016).