

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
**Université A/Mira de Béjaïa**  
Faculté des Sciences Exactes  
Département d'Informatique



MÉMOIRE DE FIN DE CYCLE  
En vue de l'obtention de diplôme de master en informatique  
Option : Administration et Sécurité des Réseaux

## *Thème*

---

# Annotation sémantique des services web à base des données ouvertes liées

---

Présenté par :

- ✓ Mlle. BOUDJOU Wahiba.
- ✓ M. KHIMA Sofiane.

Encadré par :

- ✓ Mme. EL BOUHISSI Houda.

*Devant le jury composé de :*

- ✓ Président : M. MOKTEFI Mohand .
- ✓ Examineur : M. AMAD Mourad.
- ✓ Examinatrice : Mlle. BOUCHELAGHEM Siham.

Promotion 2016/2017

# Remerciements

*Tout d'abord nous tenons à remercier le bon Dieu de nous avoir donné la force, le courage et la volonté pour accomplir ce travail.*

*Nous remercions nos familles respectives et particulièrement nos parents pour leurs soutiens qu'ils nous ont accordés tout au long de notre chemin.*

*Nous remercions notre encadreur **Mme. EL BOUHISSI Houda** pour tout le temps qu'elle nous a consacré, pour ses précieux conseils et pour son aide et son appui tout au long de notre travail.*

*Nous remercions tout particulièrement les membres du jury qui ont accepté de juger notre modeste travail.*

*Nous remercions tout nos enseignants du département informatique de l'université **ABDRRAHMANE MIRA DE BEJAIA**.*

*Nous remercions nos chers ami(e)s qui sont toujours présents et fidèles.*

*Nous n'oublions pas de remercier tous ceux qui ont contribué, de près ou de loin à la réalisation de ce travail.*

# Dédicace

*À ma raison de vivre, l'affable, l'honorable, l'aimable : Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi. C'est pour toi MAMA,*

*À mon très cher père qui n'est jamais cessé de m'aider et m'encourager durant tout mon parcours et qui m'est permis d'être aujourd'hui ce que je suis,*

*Je vous dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, vous préserver et vous accorder santé, longue vie et bonheur,*

*À mes chers frères Adel et Massi,*

*À sœurs Ouardia et Nadjet,*

*À toute ma famille,*

*À tous mes amis et anciens collègues avec qui j'ai partagé les plus beaux et inoubliables moments de ma vie,*

*À toutes ces personnes, les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour vous.*

*Wahiba*

# Dédicace

*Je dédie ce mémoire à toute ma famille,*

*À mon très cher père Rachid,*

*À mon adorable mère Karima qui m'a beaucoup soutenue,*

*À mon frère sid ali et ma sœur léna.,*

*Je le dédie particulièrement à mes grands-parents et aussi à tous mes oncles et tantes, cousins et cousines,*

*Je ne saurai terminer sans citer mes amis,*

*Enfin je le dédie à tous mes amis et les personnes qui me connaissent, qu'ils trouvent à travers ce travail ma sincère reconnaissance.*

*Sofiane*

# Table des matières

<b>Table des matières</b>	<b>III</b>
<b>Liste des figures</b>	<b>V</b>
<b>Liste des tableaux</b>	<b>VI</b>
<b>Liste des abréviations</b>	<b>VII</b>
<b>Liste des abréviations</b>	<b>VII</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Les services Web</b>	<b>4</b>
1.1 Introduction . . . . .	4
1.2 Service web . . . . .	4
1.2.1 Définition et Concepts des services web . . . . .	4
1.2.2 Objectif des services web . . . . .	5
1.2.3 Architecture des services web . . . . .	6
1.2.4 Les standards utilisés pour le web service . . . . .	7
1.3 Web sémantique . . . . .	10
1.3.1 Principe général du web sémantique . . . . .	11
1.3.2 L'architecture du web sémantique . . . . .	11
1.3.3 Langages du Web sémantique . . . . .	13
1.3.4 Ontologies . . . . .	15
1.4 Service web sémantique . . . . .	17
1.4.1 Classification et présentation des approches . . . . .	17
1.5 Linked Open Data . . . . .	21
1.5.1 Définitions . . . . .	21
1.5.2 Les quatre principes des Linked Data . . . . .	22
1.5.3 Les cinq étoiles des Linked Open Data . . . . .	22
1.6 Conclusion . . . . .	23
<b>2 Annotation sémantique</b>	<b>24</b>
2.1 Introduction . . . . .	24

2.2	Définition de l'annotation sémantique et ses objectifs . . . . .	24
2.3	L'annotation et le Web Sémantique . . . . .	25
2.4	Exploitation des annotations sémantiques . . . . .	25
2.5	Méthodes et éléments de base pour l'annotation sémantique . . . . .	25
2.6	Les Langages de l'annotation sémantique . . . . .	26
2.7	Problèmes liées à l'annotation sémantique des services Web . . . . .	27
2.8	Les approches d'annotation des services Web . . . . .	27
2.8.1	Les approches à base Machine Learning . . . . .	28
2.8.2	Les approches à base Workflow . . . . .	29
2.8.3	Utilisation de "Ontology Matching" pour l'annotation semi-automatique des services Web . . . . .	29
2.9	Conclusion . . . . .	32
<b>3</b>	<b>Conception</b> . . . . .	<b>35</b>
3.1	Introduction . . . . .	35
3.2	Architecture de l'annotation sémantique des services web à partir des données ouvertes liées(LOD) . . . . .	35
3.3	Création des ontologies à travers les données ouvertes liées (Linked Open Data) . . . . .	37
3.3.1	Description des (dataset) utilisé pour la construction d'ontologie biomédicale . . . . .	38
3.3.2	Model conceptuel . . . . .	40
3.4	L'annotation sémantique des services web par l'ontologie construite . . . . .	41
3.4.1	Le système SAWS . . . . .	42
3.4.2	Similarité sémantique . . . . .	43
3.4.3	Le standard WSDL . . . . .	46
3.4.4	Les phases du processus d'annotation sémantique des services web par une ontologie construite à partir les données ouvertes liés(LOD) . . . . .	46
3.5	Conclusion . . . . .	48
<b>4</b>	<b>Implémentation</b> . . . . .	<b>49</b>
4.1	Introduction . . . . .	49
4.2	Outils d'implémentation . . . . .	49
4.2.1	Java . . . . .	49
4.2.2	NetBeans . . . . .	50
4.2.3	Jena . . . . .	50
4.2.4	L'API JDOM . . . . .	50
4.3	Plan de l' implémentation d'annotation sémantique de service web par l'ontologie construite . . . . .	50
4.4	Présentation de quelques fenêtres . . . . .	51
4.4.1	Fenêtre principale . . . . .	51
4.4.2	Panneau d'ontologie . . . . .	52
4.4.3	Panneau de service . . . . .	55

4.4.4	SAWSDL Namespace . . . . .	57
4.4.5	modelReference et SchemaMapping . . . . .	57
4.4.6	Le processus de "matching" . . . . .	61
4.5	Conclusion . . . . .	63
	<b>Conclusion générale</b>	<b>64</b>
	<b>Bibliographie</b>	<b>65</b>

# Liste des figures

1.1	Architecture des services web. . . . .	7
1.2	Le contenu de l'annuaire UDDI [1]. . . . .	9
1.3	Relation entre UDDI, WSDL et SOAP [8]. . . . .	10
1.4	Les couches du web sémantique [12]. . . . .	12
1.5	Niveau supérieur de l'ontologie d'OWL-S. . . . .	18
1.6	Les éléments fondamentaux de WSMO. . . . .	19
1.7	Les cinq étoiles des Linked Open Data. . . . .	22
3.1	Architecture générale de l'annotation sémantique des services web à partir des données ouvertes liées(LOD). . . . .	36
3.2	Interconnexion entre les "dataset" au domaine biomédical [49]. . . . .	39
3.3	schéma d'ontologie biomédicale. . . . .	40
3.4	Architecture générale de "mapping". . . . .	40
3.5	Schéma d'annotation sémantique des services web basé sur l'ontologie construite. . . . .	41
3.6	Deux sens et leur sens commun le plus spécifique dans une taxinomie. . . . .	43
4.1	Architecture global de l' annotation sémantique. . . . .	51
4.2	Fenêtre principale de l'outil "SAWS". . . . .	52
4.3	Barre d'outils de gestion des fichiers d'ontologie (Ouvrir, Recharger, Fermer). . . . .	52
4.4	Boîte de dialogue Open Ontology. . . . .	53
4.5	Passage entre plusieurs ontologies à l'aide de la liste déroulante. . . . .	53
4.6	Recherche d'une classe d'ontologie. . . . .	54
4.7	Sélection de la classe résultante. . . . .	54
4.8	Barre de recherche. . . . .	55
4.9	Barre d'outils du panneau de service. . . . .	55
4.10	Boîte de dialogue pour le fichier .xsd non trouvé. . . . .	56
4.11	Fichier xsd chargé sur un autre onglet. . . . .	56
4.12	Syntaxe XML correcte. . . . .	56
4.13	Syntaxe XML erronée. . . . .	56
4.14	Buttons d'ouverture WSDL et OWL. . . . .	56
4.15	L'ajout du bouton de l'espace de nom SAWSDL. . . . .	57
4.16	Boite de dialogue d'ajout d'un modèle de référence. . . . .	58
4.17	la boîte de dialogue d'ajout du mappage de schéma. . . . .	59

4.18 Ajout de l'URL du mappage de schéma. . . . .	59
4.19 Suppression d'un élément SAWSDL . . . . .	60
4.20 Button de matching. . . . .	61
4.21 Boite de Dialogue de recommandation. . . . .	62
4.22 Réussite des annotations recommandées ajoutées . . . . .	62

# Liste des tableaux

2.1	Tableau comparatif . . . . .	34
3.1	Interconnexion entre les "dataset" au domaine biomédical. . . . .	39

# Liste des abréviations

<b>API</b>	<b>A</b> pplication <b>P</b> rogramminig <b>I</b> nterface.
<b>CDDL</b>	<b>C</b> ommon <b>D</b> evelopment and <b>D</b> istribution <b>L</b> icence.
<b>CN</b>	<b>C</b> ompound <b>N</b> ouns.
<b>DAML</b>	<b>D</b> arpa <b>A</b> gent <b>M</b> arKup <b>L</b> anguage.
<b>DARPA</b>	<b>D</b> efense <b>A</b> dvanced <b>R</b> esearch <b>P</b> rojects <b>A</b> gency.
<b>HTTP</b>	<b>H</b> yper <b>T</b> ext <b>T</b> ransport <b>P</b> rotocol.
<b>IDE</b>	<b>I</b> ntegrated <b>D</b> evelopment <b>E</b> nvironment .
<b>IOL</b>	<b>O</b> ntology <b>I</b> nternational <b>L</b> anguage.
<b>ISO</b>	<b>I</b> nternational <b>S</b> tandard <b>O</b> rganisation.
<b>JDOM</b>	<b>J</b> ava <b>D</b> ocument <b>O</b> bjct <b>M</b> odel.
<b>LOD</b>	<b>L</b> inked <b>O</b> pen <b>D</b> ata.
<b>MWSAF</b>	<b>M</b> ETEOR- <b>S</b> <b>W</b> eb <b>S</b> ervice <b>A</b> nnotation <b>F</b> rame <b>W</b> ork.
<b>OIL</b>	<b>O</b> ntology <b>I</b> nterchange <b>L</b> anguage.
<b>OWL</b>	<b>O</b> ntology <b>W</b> eb <b>L</b> anguage.
<b>OWL-DL</b>	<b>O</b> ntology <b>W</b> eb <b>L</b> anguage - <b>D</b> escription <b>L</b> ogic.
<b>OWL-S</b>	<b>O</b> ntology <b>W</b> eb <b>L</b> anguage for <b>S</b> emantic <b>W</b> eb.
<b>RDF</b>	<b>R</b> essource <b>D</b> escription <b>F</b> ramework.
<b>RDF-S</b>	<b>R</b> essource <b>D</b> escription <b>F</b> ramework <b>S</b> chema.
<b>RI</b>	<b>R</b> echerche <b>I</b> nformation.
<b>RIF</b>	<b>R</b> ule <b>I</b> nterchange <b>F</b> ormat.
<b>SAWS</b>	<b>S</b> emantic <b>A</b> nnotation <b>W</b> eb <b>S</b> ervice .
<b>SAWSDL</b>	<b>S</b> emantic <b>A</b> nnotation for <b>W</b> eb <b>S</b> ervice <b>D</b> escription <b>L</b> anguage.
<b>SOAP</b>	<b>S</b> imple <b>O</b> bject <b>A</b> ccess <b>P</b> rotocol.

<b>SPARQL</b>	<b>Q</b> uery <b>L</b> anguage <b>F</b> or <b>R</b> DF.
<b>SQL</b>	<b>S</b> tructured <b>Q</b> uery <b>L</b> anguage.
<b>SWRL</b>	<b>S</b> emantic <b>W</b> eb <b>R</b> ule <b>L</b> anguage.
<b>UDDI</b>	<b>U</b> niversal <b>D</b> escription <b>D</b> iscovery and <b>I</b> ntegration.
<b>URI</b>	<b>U</b> niform <b>R</b> essource <b>I</b> dentifier.
<b>URL</b>	<b>U</b> niform <b>R</b> essource <b>L</b> ocator.
<b>W3C</b>	<b>W</b> orld <b>W</b> ide <b>W</b> eb <b>C</b> onsortium.
<b>WSML</b>	<b>W</b> eb <b>S</b> ervice <b>M</b> odeling <b>L</b> anguage.
<b>WSMO</b>	<b>W</b> eb <b>S</b> ervice <b>M</b> odeling <b>O</b> ntology.
<b>WWW</b>	<b>W</b> orld <b>W</b> ide <b>W</b> eb
<b>XML</b>	<b>X</b> Extensible <b>M</b> arkup <b>L</b> anguage.
<b>XML-S</b>	<b>X</b> Extensible <b>M</b> arkup <b>L</b> anguage <b>S</b> chema.
<b>XSD</b>	<b>X</b> ML <b>S</b> chema <b>D</b> ocument.

# *Introduction générale*

Avec l'évolution de l'Internet, de nouvelles technologies du Web ont vu le jour. Les services Web permettent le changement de la nature du Web, du document utilisé par les organisations pour la publication des informations, au Web orienté services. Utilisés par les humains afin d'accomplir une tâche donnée, ces services Web accordent aux serveurs d'applications, la communication entre eux. Ils sont suffisamment développés pour que les développeurs les utilisent maintenant dans tous les domaines de l'informatique, afin de récolter divers bénéfices de la technologie. La diversité de ces infrastructures et des organismes qui les déploient entraîne des hétérogénéités.

Les Services Web Sémantiques (SWS) ont eu un intérêt croissant comme une solution proposée pour faciliter et automatiser la découverte, la composition et le déploiement des Services Web, pour profiter pleinement des avantages de cette technologie. Malgré son importance, il n'y a que peu de travaux qui ont traité le problème du Web actuel, les machines ne sont pas capables d'interpréter l'information, et ne fournissent donc, que des outils de localisation, de transfert, de mise en forme et de présentation vers le Web sémantique, l'information aura une signification explicite, ce qui permettra aux machines de la traiter en temps réel.

La plupart de ces travaux n'ont pas traité les particularités des Services Web. Aussi la découverte des Services Web est l'un des challenges du paradigme émergent SOA (Service Oriented Architecture). La majorité des travaux existants proposent une recherche à base de mots clés. Cependant, ce type d'approche ne permet pas de récupérer certains Services Web susceptibles de satisfaire les besoins des utilisateurs.

Les technologies proposées n'exploitent pas les informations contenues dans les interfaces du service telles que le WSDL et exigent l'intervention d'un utilisateur humain et des experts du domaine pour l'exécution des tâches telles que la découverte, la sélection, l'invocation et la composition.

Toutes les approches de conception des Services Web Sémantiques souffrent plus ou moins du manque de méthodologies, du manque d'architectures flexibles pouvant décrire efficacement la sémantique liée aux applications, et aussi du manque de maturité ou parfois même de l'absence de mécanismes de description et de médiation efficaces.

Ce travail de recherche a pour objectif principal de présenter une nouvelle approche pour l'annotation sémantique des services Web, par une ontologie construite à partir des données ouvertes liées (Linked Open Data).

L'annotation sémantique consiste à enrichir et à compléter la description d'un service. Elle établit des correspondances entre des éléments de la description et les concepts d'un ensemble d'ontologies de référence. Une ontologie de référence permet de représenter un domaine par des structures interprétables par une machine.

Il s'agit de passer en revue toutes les technologies inhérentes, de faire un état de l'art détaillé sur l'annotation sémantique des services Web et d'implémenter un prototype basé sur l'approche SAWSDL (Semantic Annotations for Web Services Description Language and XML Schema). Tester sa validité, et voire même ses insuffisances afin de proposer de nouvelles issues pour l'améliorer.

Le standard SAWSDL présente un mécanisme permettant d'annoter sémantiquement les services décrits avec WSDL et leurs schémas XML (Extensible Markup Language) associés.

Notre contribution consiste à construire une ontologie de domaine à partir des données ouvertes liées et annoter la description WSDL d'un service Web existant avec cette ontologie.

L'approche se déroule en deux principales étapes, à savoir, une étape de construction d'une ontologie à partir des données ouvertes liées, et une autre étape d'annotation sémantique des services Web par l'ontologie construite. L'étape de construction d'ontologie concerne la création et l'enrichissement d'une ontologie du domaine biomédical (l'étude de cas que nous avons menée) afin d'enrichir les concepts d'ontologie en s'appuyant sur les instances disponibles dans le cloud « Linked Open Data. ».

L'étape d'annotation sémantique comprend le processus d'association de chaque concept d'ontologie aux concepts des services Web.

Ainsi, le présent mémoire est organisé en quatre chapitres répartis comme suit :

- ✓ Le premier chapitre intitulé « **Les services Web** » a pour objectif de présenter le domaine d'étude. Nous présentons les caractéristiques des services Web, l'architecture, et les notions liées à ce paradigme, ainsi que les standards. Par la suite, nous présentons le Web sémantique et ses différents langages. Enfin, nous décrivons le domaine des services Web sémantiques, étant donné qu'il constitue le noyau de notre travail, et introduisons certaines notions sur les données ouvertes liées.
- ✓ Le second chapitre intitulé « **Annotation sémantique** » est consacré à l'annotation sémantique des services Web, et les différents travaux relatifs à ce domaine. Ainsi, une étude

comparative des différentes approches sera présentée.

- ✓ Le troisième chapitre intitulé « **Conception** » présente en détail la contribution que nous avons utilisé au cours de nos recherches de construction d'ontologie à partir des données ouvertes liées (Linked Open Data) et l'annotation sémantique des services web par l'ontologie construite. Nous présenterons aussi les différentes étapes de notre approche.
- ✓ Dans le quatrième chapitre intitulé « **Implémentation** », nous introduisons les outils utilisés et exposons quelques interfaces de notre application.

Enfin, nous clôturons ce mémoire par une conclusion générale résumant les points essentiels de notre travail et dégagerons quelques perspectives envisagées pour la solution proposée.

## 1.1 Introduction

Les services web constituent un phénomène assez récent, il a eu un grand intérêt de la part des développeurs des systèmes d'information. De cela, il est considéré comme étant la solution idéale intégrant des fonctionnalités déjà implémentées afin de construire et développer des systèmes d'information. Malgré le succès de cette technologie, les services Web ont confronté le problème de l'interopérabilité, et pour s'en débarrasser, une jungle de standards a été élaborée, ce qui nous invite à aborder les technologies reliées telles qu'Extensible Markup Language (XML), Simple Object Acces Protocol (SOAP), Web Services Description Language (WSDL) et Universal Description Discovery and Integration (UDDI).

Nous introduirons dans ce chapitre les services web et les technologies adjacentes, les objectifs visés par les services web sémantiques ainsi que les différents langages qui sont utilisés pour expliciter la sémantique dans les descriptions des services web, nous aborderons aussi les concepts de données ouvertes liées.

## 1.2 Service web

Les services Web fournissent une nouvelle manière de développer des applications conformes aux besoins de l'Internet en vue de rendre le Web plus dynamique. Ils semblent être la solution la plus adaptée pour assurer l'interopérabilité permettant de transmettre les données entre les différentes applications d'une organisation. En effet, cette technologie permet de réaliser le traitement de ces données, et gérer les liaisons entre les différentes applications.

### 1.2.1 Définition et Concepts des services web

#### 1.2.1.1 Origines

On peut distinguer trois phases de développement dans l'histoire du Web :

- **Le Web du Document**, utilisé principalement par des organisations et des particuliers pour publier des informations.

- **Le Web Applicatif**, les entreprises ont commencé à utiliser le Web à des fins commerciales. Les sites Internet sont devenus plus interactifs, et plus complexes, gérés par des serveurs d'applications, capables de distribuer leur charge avec d'autres serveurs, en fonction des besoins.
- **Le Web des Services** est la phase émergente, dans laquelle les serveurs d'application précédents communiquent désormais entre eux. Cette évolution a été poussée par le désir de pouvoir réaliser des échanges interentreprises dans un environnement automatisé et ouvert tel qu'Internet. L'échange de données informatisées entre deux applications nécessite une normalisation des messages échangés.

### 1.2.1.2 Définition Des services web

Les services web représentent un mécanisme de communication entre applications distantes à travers le réseau internet indépendant de tout langage de programmation et de toute plate-forme d'exécution.

Les services web ont été proposés initialement par IBM et Microsoft, puis en partie standardisés par le W3C<sup>1</sup> (le consortium du World Wide Web).

Les services web sont la nouvelle vague des applications web. Ce sont des applications modulaires, auto-contenues et auto-descriptives qui peuvent être publiées, localisées et invoquées depuis le web. Les services web effectuent des actions allant de simples requêtes à des processus métiers complexes. Une fois qu'un service web est déployé, d'autres applications (y compris des services web) peuvent le découvrir et l'invoquer [1].

### 1.2.2 Objectif des services web

Les services Web fournissent un lien entre applications. Ainsi, des applications utilisant des technologies différentes peuvent envoyer et recevoir des données au travers de protocoles compréhensibles par tout le monde.

Les services Web sont normalisés, car ils utilisent les standards XML et HTTP pour transférer des données et ils sont compatibles avec de nombreux autres environnements de développement. Ils sont donc indépendants des plates-formes. C'est dans ce contexte-là qu'un intérêt très particulier a été attribué à la conception des services Web, puisqu'ils permettent aux entreprises d'offrir des applications accessibles à distance par d'autres entreprises. Cela s'explique par le fait que les services Web n'imposent pas de modèles de programmation spécifiques. En d'autres termes, les services Web ne sont pas concernés par la façon dont les messages sont produits ou consommés

---

1. World Wide Web Consortium, abrégé W3C, est un consortium fondé en octobre 1994, par Tim Berners-Lee, pour promouvoir la compatibilité des technologies du World Wide Web telles que HTML, XHTML, XML, CSS et SOAP.

par des programmes. Cela permet aux vendeurs d'outils de développement d'offrir différentes méthodes et interfaces de programmation au-dessus de n'importe quel langage de programmation, sans être contraints par des standards. Les services Web représentent donc la façon la plus efficace de partager des méthodes et des fonctionnalités. De plus, ils réduisent le temps de réalisation en permettant de tirer directement parti de services existants [2].

L'objectif principal des services Web est de faciliter le plus possible l'accès aux applications entre entités et ainsi de simplifier les échanges de données. Cette interopérabilité est due à l'utilisation de normes ouvertes. L'OSI et le W3C sont les comités de coordination responsables de l'architecture et de standardisation des services Web. Pour améliorer l'interopérabilité entre les réalisations de service Web, l'organisation WS-I a développé une série de profils pour faire évoluer les futures normes impliquées. L'aspect le plus important des Web Services est qu'ils reposent sur plusieurs standards qui permettent la structuration des architectures. Cette collection de normes et de protocoles est appelée Web Services Protocol Stack. Elle contient entre autres XML et SOAP pour le formatage des données, WSDL pour la description des services Web et UDDI pour la recherche des services Web nécessaire au bon fonctionnement des applications [3].

### 1.2.3 Architecture des services web

Les efforts de recherche et de développement récents autour des Web services ont conduit à un certain nombre de spécifications qui définissent aujourd'hui l'architecture de référence des Web services. L'architecture des services web consiste à acheminer une demande de service entre clients/serveur, qui sont, tous les deux, connectés au réseau. Le client est soit un être humain qui formule sa requête via un navigateur web, soit une demande de service web automatisée par une application.

Cette architecture vise trois objectifs importants : identification des composants fonctionnels, définition des relations entre ces composants et établissement d'un ensemble de contraintes sur chaque composant de manière à garantir les propriétés globales de l'architecture.

L'architecture de référence des Web services (cf. FIGURE 1.1) s'articule autour des trois rôles suivants :

- **Le fournisseur de service** : correspond au propriétaire du service. Techniquement, il est constitué par la plate-forme d'accueil du service.
- **Le client** : correspond au demandeur de service. Techniquement, il est constitué par l'application qui va rechercher et invoquer un service. L'application client peut être elle-même un Web service.
- **L'annuaire des services** : correspond à un registre de descriptions de services offrant des facilités de publication de services à l'intention des fournisseurs ainsi que des facilités de recherche de services à l'intention des clients.

Les interactions de base entre ces trois rôles incluent les opérations de publication, de recherche et de liens d'opérations (binding). Nous décrivons ci-dessous un scénario type d'utilisation de cette architecture. Le fournisseur de services définit la description de son service(en WSDL) et la publie dans un annuaire de service (ex.UDDI). Le client utilise les facilités de recherche disponibles(les fichiers WSDL) au niveau de l'annuaire pour retrouver et sélectionner un service donné.

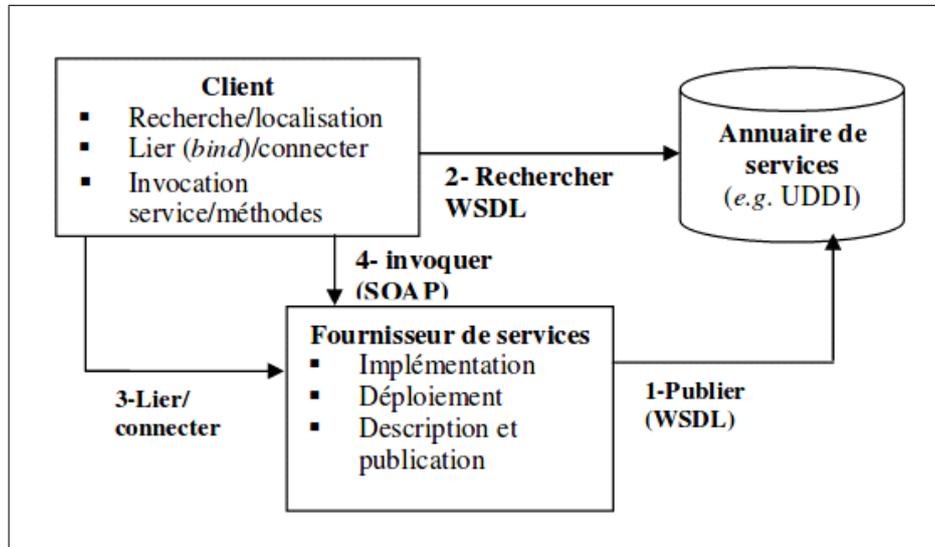


FIGURE 1.1 – Architecture des services web.

Il examine ensuite la description du service sélectionné pour récupérer les informations nécessaires lui permettant de se connecter au fournisseur du service et d'interagir avec l'implémentation du service considéré [4].

## 1.2.4 Les standards utilisés pour le web service

Les services Web reprennent la plupart des idées et des principes du Web (HTTP, XML), et les appliquent à des interactions entre machines. Comme pour le World Wide Web, les services Web sont construits autour de standards qui sont SOAP, WSDL et UDDI. Ils ont été conçus pour être réalisés sur de nombreux systèmes développés et déployés de façon indépendante.

### 1.2.4.1 Simple Object Access Protocol SOAP

#### Définition

SOAP [5] c'est un protocole de dialogue par appels de procédures à distance entre objets logiciels. Sa syntaxe d'utilisation est fondée sur XML et ses commandes sont envoyées sur Internet par l'intermédiaire du protocole HTTP, mais aussi SMTP et POP sous forme de texte structuré.

Il permet aux systèmes objets distribués de solliciter et d'obtenir des services rendus par d'autres objets, il est moins lourd à mettre en œuvre que d'autres protocoles et c'est pour cela qu'il est de

plus en plus adopté.

Le protocole SOAP est une note du Consortium W3C dont Microsoft fait partie, mais qui n'est pas spécifique à Microsoft et Windows. IBM a également participé à l'élaboration de ce protocole. De plus il existe des implémentations Java, et Borland vient déjà d'implémenter SOAP sous Windows dans Delphi 6 et sous Linux avec Kylix.

#### 1.2.4.2 Universal Description Discovery and Integration UDDI

UDDI<sup>2</sup> (Universal Description Discovery and Integration) [6] est proposé par le consortium OASIS<sup>3</sup> (Organization for the Advancement of Structured Information Standards) afin de publier et rechercher des services Web. Les spécifications UDDI sont les plus utilisées pour implémenter les registres internes aux organisations, et qu'elles sont souvent étendues par des travaux académiques qui ambitionnent de les appliquer à d'autres domaines tels que le Web sémantique.

UDDI a été conçu en 2000 à l'initiative d'un ensemble d'industriels (Ariba<sup>4</sup>, IBM<sup>5</sup>, Microsoft<sup>6</sup>), en vue de devenir le registre standard de la technologie des services Web. Pour convenir à la technologie des services Web, les services référencés dans UDDI sont accessibles par l'intermédiaire du protocole de communication SOAP, et la publication des informations concernant les fournisseurs et les services doit être spécifiée en XML afin que la recherche et l'utilisation soient faites de manière dynamique et automatique. UDDI constitue un méta-service possédant des fonctions de publication et de recherche. UDDI permet de stocker l'ensemble des services web sur un seul serveur, dont le contenu est dupliqué et synchronisé sur plusieurs sites miroirs.

#### Recherche de services Web avec UDDI

La recherche et la sélection dans UDDI reposent sur la publication préalablement décrite du service et de son fournisseur. Le futur client peut connaître par l'intermédiaire d'UDDI : les fournisseurs d'un service, les services proposés par un fournisseur donné, les moyens d'invoquer un service. Pour apporter aux clients la réponse à ces questions, UDDI organise l'ensemble des informations qu'il contient en trois parties, spécifiées en XML. Chacune d'elles peut être utilisée pour faire une recherche via UDDI. Ces parties sont les suivantes :

**Les pages blanches** : Cette catégorie inclut des informations générales au sujet d'une compagnie spécifique ; par exemple, nom commercial, description d'affaires, et adresse.

**Les Pages jaunes** : Cette catégorie inclut des données de classification générales pour la compagnie ou le service offert. Par exemple, ces données peuvent inclure l'industrie, le produit, ou les codes géographiques basés sur des taxonomies standard.

---

2. <http://uddi.xml.org/>

3. <http://www.oasis-open.org/>

4. <http://www.ariba.com/>

5. <http://www.ibm.com>

6. <http://research.microsoft.com>

**Pages vertes** : Cette catégorie inclut des informations techniques sur un service de Web (un indicateur selon des spécifications externes et une adresse pour appeler le service de Web) [1] .

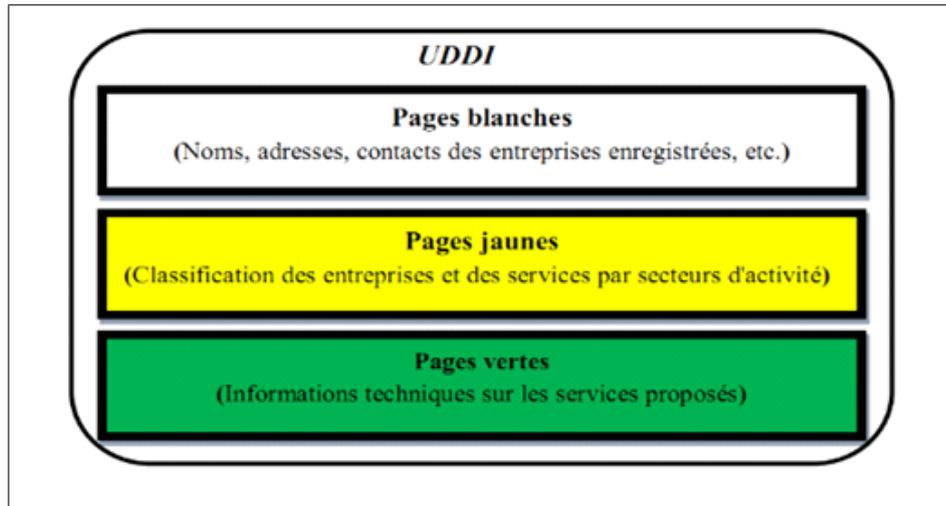


FIGURE 1.2 – Le contenu de l’annuaire UDDI [1]

### Description fonctionnelle et structurelle WSDL

WSDL (Web Services Description Language) est un langage de la famille XML permettant de décrire les types de données supportés et les fonctions offertes par un service Web. L’objectif est de fournir la description des services en XML, indépendamment de la plate-forme et du langage utilisé, et ainsi qu’elle soit sous une forme que des personnes ou des programmes peuvent interpréter. Les descriptions WSDL sont en fait l’équivalent des interfaces IDL (Interface Definition Language) de CORBA par exemple.

Dans le langage WSDL, un service est vu comme une collection de messages pour les échanges et une collection de points d’entrée. Un point d’entrée consiste en la description abstraite d’une interface et de son implantation. La description abstraite contient :(i) la définition des messages qui sont consommés et générés par le service (les entrées et les sorties), et (ii) la signature des opérations offertes par le service. Elle contient essentiellement l’indication du protocole utilisé pour échanger des messages avec le service (par exemple SOAP au-dessus de HTTP) et les associations entre la description de l’interface abstraite du service et les types de messages supportés par le protocole de communication sous-jacent (par exemple SOAP) [7].

Ces protocoles (SOAP, UDDI et WSDL) sont maintenant universellement acceptés et sont donc les standards qui constituent l’architecture des Services Web. Ensemble, ils résolvent les problèmes de l’hétérogénéité des systèmes pour l’intégration d’applications en ligne. La Figure 1.3 montre l’interaction des trois technologies.

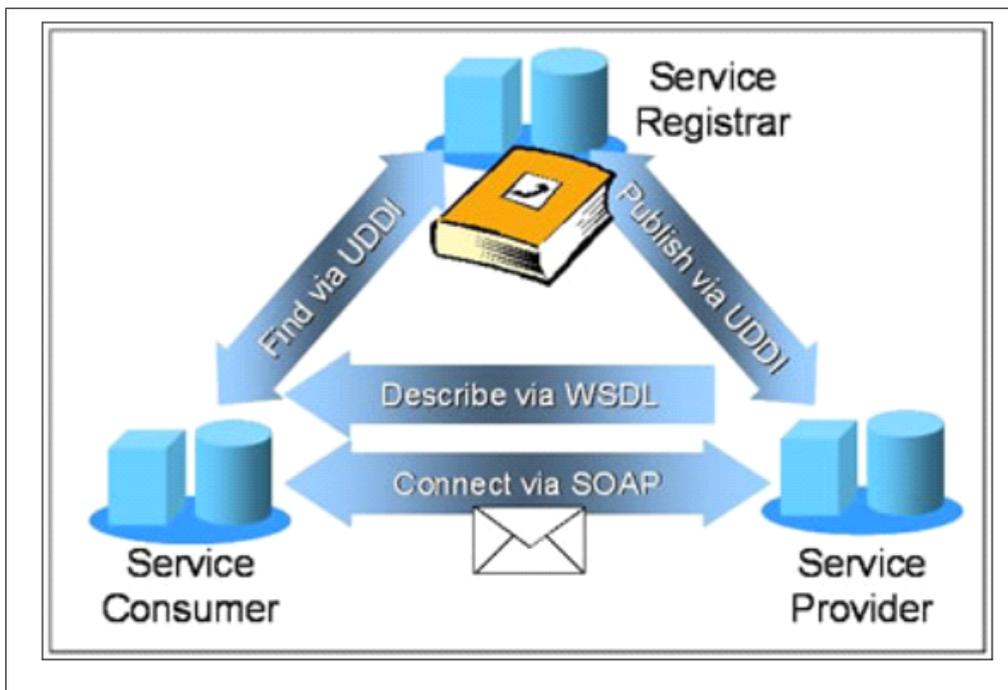


FIGURE 1.3 – Relation entre UDDI, WSDL et SOAP [8].

Pour comprendre cette image, imaginons qu'un informaticien veut développer un système d'informations boursières. Dans ce système, il veut montrer l'évolution de certaines actions. Tout ce qu'il doit faire c'est utiliser UDDI pour interroger un registre qui lui permet de trouver un fournisseur de tel service.

Quand la recherche est terminée, l'informaticien pourra connaître la structure des messages SOAP nécessaires pour l'utilisation du service grâce à la description WSDL donnée par le fournisseur.

Ces trois standards ont permis le développement initial des services web et ils ont incité beaucoup d'entreprises à leur utilisation. Cependant, à cause des problèmes de sécurité et de fiabilité, du besoin croissant de décrire et d'implémenter de nouveaux et plus complexes scénarios de business, plusieurs autres standards ont été proposés. Certains se sont affirmés, d'autres sont disparus et d'autres encore ont été réunis pour augmenter leur puissance [8].

### 1.3 Web sémantique

Le Web sémantique [9] désigne un ensemble de technologies visant à rendre le contenu des ressources du World Wide Web accessible et utilisable par les programmes et agents logiciels, et ce, grâce à un système de métadonnées formelles utilisant notamment la famille de langages développés par le W3C (World Wide Web Consortium). À sa création par Tim Berners Lee, au début des années 1990, le web était exclusivement destiné à partager des informations sous forme de pages HTML, affichables par un logiciel «navigateur web», et généralement destinées à être lues par un utilisateur humain. Très rapidement, on s'est rendu compte que cette conception du

web était bien trop limitée, et ne permettait pas un réel partage du savoir. Les limites du Web actuel sont nombreuses, nous mentionnons :

- ✓ L'hétérogénéité des formats d'information (Word, PDF ...).
- ✓ La nature dynamique et évolutive de l'information.
- ✓ L'absence de sémantique ou structure de l'information elle-même, etc.

Une des solutions existantes correspond à la vision du Tim Berners-Lee qui a défini le Web sémantique en (T. Berners-Lee, 2001) comme suit :

" Le Web sémantique est une extension du Web actuel (prolongation du Web actuel), dans laquelle l'information reçoit une signification bien définie, améliorant les possibilités de travail collaboratif entre les ordinateurs et les personnes".

Il s'agit d'arriver à un Web intelligent, où les informations ne seraient plus stockées, mais comprises par les ordinateurs afin d'apporter à l'utilisateur ce qu'il cherche vraiment.

D'après la définition de Tim Berners-Lee, le Web sémantique permettra (contrairement au Web actuel vu comme un Web syntaxique) de rendre le contenu sémantique des ressources Web interprétables, non seulement par l'homme, mais aussi par la machine[10].

### 1.3.1 Principe général du web sémantique

Le Web sémantique est entièrement fondé sur le Web et ne remet pas en cause ce dernier. Le Web sémantique s'appuie donc sur la fonction primaire du Web «classique» : un moyen de publier et de consulter des documents. Mais les documents traités par le Web sémantique contiennent non pas des textes en langage naturel (français, espagnol, chinois, etc.), mais des informations formalisées pour être traitées automatiquement. Ces documents sont générés, traités, échangés par des logiciels. Ces logiciels permettent souvent, sans connaissance informatique, de :

- ✓ Générer des données sémantiques à partir de la saisie d'information par les utilisateurs.
- ✓ Agréger des données sémantiques afin d'être publiées ou traitées.
- ✓ Publier des données sémantiques avec une mise en forme personnalisée ou spécialisée.
- ✓ Échanger automatiquement des données en fonction de leurs relations sémantiques.
- ✓ Générer des données sémantiques automatiquement, sans saisie humaine, à partir de règles d'inférences [11].

### 1.3.2 L'architecture du web sémantique

L'architecture du web sémantique [12] s'appuie sur une pyramide de langages proposée par Tim Berners-Lee pour représenter des connaissances sur le web en satisfaisant les critères de standardisation, interopérabilité et flexibilité. Cette architecture en couches (FIGURE 1.4) peut permettre une approche graduelle dans les processus de standardisation et d'acceptation par les utilisateurs.

Un langage de la couche haute doit être une extension du langage de la couche au-dessous. Jusqu'à aujourd'hui seulement les couches de basses sont relativement stabilisées.

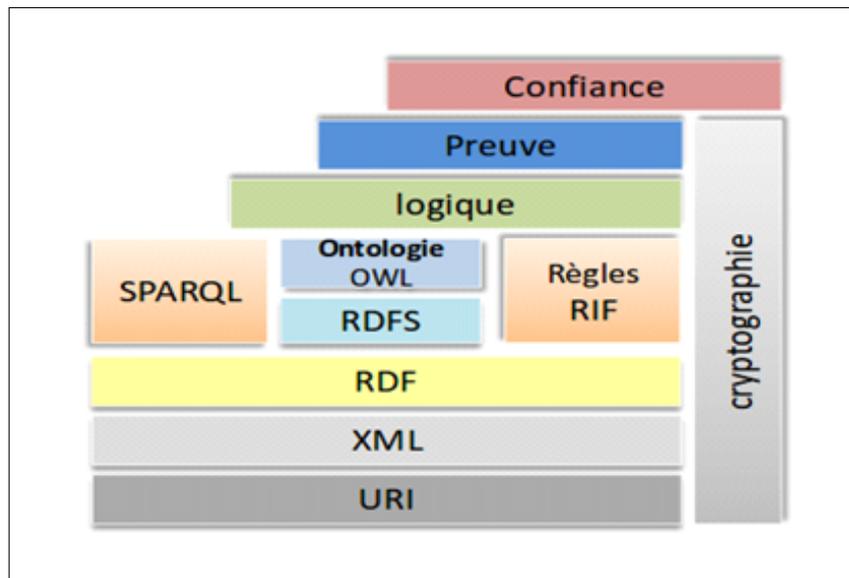


FIGURE 1.4 – Les couches du web sémantique [12].

La liste suivante introduit les structures et fonctions de chaque couche dans l'architecture du web sémantique :

- **XML** fournit une syntaxe pour décrire la structure du document, créer et manipuler des instances des documents. Il utilise l'espace de nommage (namespace) afin d'identifier les noms des balises (Tags) utilisés dans le document XML. Le schéma XML permet de définir les vocabulaires pour des documents XML valides.
- **Les couches RDF et RDF-Schéma** sont considérées comme les premières fondations de l'interopérabilité sémantique. Elles permettent de décrire les taxonomies des concepts et des propriétés (avec leur signature). RDF fournit un moyen d'insérer la sémantique dans un document. L'information est conservée principalement sous forme de déclarations RDF.

Le schéma RDFS décrit les hiérarchies des concepts et des relations entre les concepts, les propriétés et les restrictions domaine/co-domaine pour les propriétés.

- **La couche de l'ontologie OWL** elle décrit des sources d'information hétérogènes, distribuées et semi-structurées en définissant les consensus du domaine commun et partagées par plusieurs personnes et communautés (la signification de la connaissance). OWL permet de construire des relations de disjonctions ou de faire des unions. Les ontologies aident la machine et l'humain à communiquer avec concision en utilisant l'échange sémantique plutôt que syntaxique.
- **la couche du langage SPARQL** ce langage est désigné par le W3C comme le standard pour l'interrogation des graphes RDF et OWL. SPARQL est l'équivalent de SQL pour le Web

des données. La syntaxe SPARQL varie quelque peu de celle du SQL, de plus, il est nécessaire de déclarer les espaces de nom utilisés lors de la requête. SPARQL se base directement sur les métadonnées RDF. Cela permet aux machines ou aux humains d'interroger des bases de données sur le Web, sans forcément en connaître le schéma au préalable, ce qui permettrait un accès aux données sans intermédiaire.

- **La couche de règles** a pour objectif de normaliser la représentation des règles RDF. Elle comporte deux langages de règles : SWRL (Semantic Web Rule Language) et RIF (Rule Interchange Format). SWRL est une extension d'OWL. RIF ne repose pas directement sur RDF, mais sur XML et il permet de faciliter l'utilisation et l'échange de règles entre les formats déjà existants.
- **La couche logique** se trouve au-dessus de la couche ontologie. Certains considèrent ces deux couches comme étant au même niveau : c'est-à-dire comme des ontologies basées sur la logique et permettant des axiomes logiques. En appliquant la logique déductive, on peut inférer de nouvelles connaissances à partir d'une information explicitement représentée.
- **Les couches preuve et confiance** fournissent des éléments pour réaliser la vérification des déclarations effectuées dans le web sémantique. On s'oriente vers un environnement du web sémantique fiable et sécurisé dans lequel nous pouvons effectuer des tâches complexes en sûreté. D'autre part, la provenance des connaissances, des données, des ontologies ou des déductions est authentifiée et assurée par des signatures numériques. Dans le cas où la sécurité est importante ou le secret est nécessaire, le chiffrement est utilisé.
- **La couche Cryptographie** a pour but de s'assurer et de vérifier que les déclarations issues du web sémantique proviennent d'une source sûre, ce qui peut être réalisé par la signature numérique des déclarations RDF.

### 1.3.3 Langages du Web sémantique

Dans le contexte du Web sémantique, plusieurs langages ont été développés. La plupart de ces langages reposent sur XML ou utilisent XML comme syntaxe. Nous allons présenter brièvement certains langages principaux XML, XML Schéma, RDF(S), OWL, SPARQL, RDF et URI.

#### 1.3.3.1 URI ( Uniform Resource Identifier)

Le web repose sur la notion d'URI [13] (Uniform Resource Identifier) qui permet d'identifier de manière unique chaque ressource du Web, comme une page, une adresse e-mail, ou une image. Le point central des URIs est l'URL (Uniform Resource Locator) traditionnelle utilisée pour définir les liens du Web. Ces URLs sont utilisées pour référencer des fichiers Web à travers un protocole particulier, comme HTTP ou FTP.

#### 1.3.3.2 XML (S) (eXtensible Markup Language (Schema))

XML-S [14] est un langage de balise permettant de structurer des données et/ou des documents sur le web. Le langage XML est interopérable du point de vue syntaxique. Il sert à représenter

des données échangeables sur le web. Ainsi, tous les langages du web sémantique sont systématiquement exprimables et échangeables dans une syntaxe XML. Brièvement parlé, XML permet aux utilisateurs d'ajouter une structure arbitraire à leurs documents sans rien dire de la signification de la sémantique de ces structures.

### 1.3.3.3 XML (eXtensible Markup Language)

Le langage XML [15] est un langage de description et d'échange de documents structurés, issu de SGML (Standard Genralized Markup Language). En effet, XML permet de décrire la structure logique de documents à l'aide d'un système de balises permettant de marquer les éléments qui composent la structure et les relations entre ces éléments.

### 1.3.3.4 RDF (Resource Description Framework)

RDF [14] est un modèle de données pour les objets (ressources) et les relations entre eux fournissant des sémantiques simples pour ce modèle de données qui peuvent être représentés en XML. RDF permet de représenter des métadonnées à propos des ressources (identifiées par des URI) du web.

### 1.3.3.5 RDF-S (Resource Description Framework Schema)

[16] est un méta modèle recommandé par le W3C permettant la définition de schéma/modèle décrivant l'univers sémantique des déclarations RDF., au même titre que le XML-S pour le langage XML. RDFS fournit ainsi un système de typage pour les déclarations RDF. Il permet la définition des classes et des sous-classes (`rdfs :Class`, `rdfs :subClassOf`) décrivant les ressources à annoter et donnant un sens aux propriétés associées aux ressources. Il permet aussi d'assigner des contraintes spécifiques sur la nature des documents et de fournir des informations sur l'interprétation des déclarations RDF. Dans le contexte du web sémantique, RDFS est utilisé pour formaliser les ontologies sur lesquelles vont se baser les annotations RDF.

### 1.3.3.6 OWL (Ontology Web Langage)

Le langage OWL (Ontology Web Langage) [17]est un composant de l'activité Web sémantique qui vise à rendre les ressources Web plus accessibles aux processus automatisés en ajoutant des informations qui décrivent le contenu Web.

Le langage d'ontologie Web OWL définit des ontologies Web. Une ontologie OWL peut contenir des descriptions de classes, de propriétés et de leurs instances. Pour une telle ontologie donnée, la sémantique formelle OWL indique comment déduire ses conséquences logiques, c'est-à-dire les faits qui ne sont pas littéralement présents dans l'ontologie, mais déduits par la sémantique.

### 1.3.3.7 SPARQL (Query Language for RDF)

SPARQL [18] est un langage de requête et un protocole qui permet de rechercher, d'ajouter, de modifier ou de supprimer des données RDF disponible sur le Web. On pourrait oser dire que SPARQL est l'équivalent du langage SQL (Structured Query Language), le langage informatique normalisé qui sert à effectuer des opérations sur des bases de données. En utilisant SQL, on accède aux données d'une base de données via ce langage de requête, alors qu'avec SPARQL, on accède aux données structurées du Web des données. Cela signifie qu'en théorie, on pourrait accéder à toutes les données du Web avec ce standard, grâce à l'interopérabilité.

### 1.3.4 Ontologies

Aujourd'hui, les ontologies apparaissent désormais comme une clé pour la manipulation automatique de l'information au niveau sémantique. Au fur et à mesure des recherches, des idées se dégagent autour du contenu des ontologies, des méthodes à utiliser pour les construire et des modèles et langages servant à leur représentation.

L'importance des ontologies est reconnue dans divers domaines de recherche comme la représentation des connaissances, l'ingénierie des connaissances, la conception de bases de données, l'intégration d'information, les systèmes d'information... Les ontologies sont aussi centrales pour le Web sémantique qui, d'une part, cherche à s'appuyer sur des modélisations de ressources du Web à partir de représentations conceptuelles des domaines concernés et, d'autre part, a pour objectif de permettre à des programmes de faire des inférences dessus.

#### 1.3.4.1 Composantes d'une ontologie

Une ontologie peut être vue comme un ensemble structuré de concepts et de relations entre ces concepts destinés à représenter les objets du monde sous une forme compréhensible aussi bien par les hommes que par les machines. Les composants d'une ontologie sont :

- **Concept** : ou classe, définissant un ensemble d'objets, abstrait ou concret, que l'on souhaite modéliser pour un domaine donné. Les connaissances portent sur des objets auxquels on se réfère à travers des concepts. Un concept peut représenter un objet matériel, une notion, une idée. Les concepts dans l'ontologie sont habituellement organisés dans des taxonomies.
- **Les instances** : ou individus, constituent la définition extensionnelle de l'ontologie (pour représenter les éléments spécifiques).
- **les relations** : Une relation permet de lier des instances de concepts ou des concepts génériques. Elles sont caractérisées par un terme ou plusieurs, et une signature qui précise le nombre d'instances de concepts que la relation lie, leurs types et l'ordre des concepts, c'est – à – dire la façon dont la relation doit être lue.
- **Les axiomes** : Une ontologie est en outre composée d'axiomes qui forment des contraintes sémantiques pour le raisonnement et donnent un acompte d'une conceptualisation. Ils prennent la forme d'une théorie logique [15].

### 1.3.4.2 Typologie des ontologies

Nous listons ci-dessous les différents types d'ontologies les plus utilisées [16] :

#### Les ontologies de représentation

N'appartiennent à aucun domaine, mais définissent et organisent les primitives de la théorie logique pour permettre la représentation des ontologies. L'exemple le plus représentatif de ce genre d'ontologie est la Frame Ontologie<sup>7</sup>, qui définit d'une manière formelle, les primitives de représentation (classes, sous classes, attributs, valeurs, relations et axiomes) dans un environnement implémentant les langages de Frame.

#### Les ontologies génériques

Sont aussi appelées Ontologie de haut niveau ou ontologie Top, elles décrivent des concepts généraux, indépendants d'un domaine ou d'un problème particulier. Elles permettent par exemple de formaliser les aspects temporels ou spatiaux des objets du monde réel. Cyc<sup>8</sup> est un exemple d'une ontologie générique portant sur des concepts de haut niveau. Ces dernières décrivent des notions générales comme les notions d'objet, de propriété, d'état, de valeur, de moment, d'évènement, d'action, de cause et d'effet.

#### Les ontologies de domaine

Elles sont construites sur un domaine particulier de la connaissance. Les ontologies de domaine fournissent des vocabulaires au sujet des concepts dans un domaine et leurs relations au sujet des activités qui ont lieu dans ce domaine, et au sujet des théories et des principes élémentaires régissant ce domaine. Plusieurs ontologies de domaines existent déjà, telle que MENELAS<sup>9</sup> dans le domaine médical. Entreprise<sup>10</sup> est un autre exemple décrivant le domaine de l'entreprise.

#### Les ontologies de tâches

L'ontologie de tâche décrit les connaissances portant sur tâches et/ou des activités particulières. Ces ontologies fournissent un ensemble de termes au moyen desquels on peut décrire au niveau générique comment résoudre un type de problème. Elles incluent des noms génériques (objectif, contrainte...), des verbes génériques (classer, sélectionner...), des adjectifs génériques (assigné..) et autres dans les descriptions de tâches.

#### Les ontologies d'application

Aussi appelée ontologie de domaine-tache : Ce sont des ontologies les plus spécifiques, elles contiennent les connaissances requises pour une application particulière permettant ainsi de modéliser une activité spécifique dans un domaine donné.

---

7. <http://www-ksl.stanford.edu/people/brauch/demo/frame-ontology/>

8. <http://www.cyc.com/cycdoc/upperont-diagram.html>

9. <http://estime.spim.jussieu.fr/Menelas/Ontologie/html/>

10. <http://www.aiai.es.ac.uk/project/entreprise>

## 1.4 Service web sémantique

Le Web sémantique constitue une prolongation, et une sorte de révolution de fond du Web actuel qui permet une définition non ambiguë de l'information, pour favoriser une meilleure coopération entre humain et machine. Il permet de s'ouvrir à de nouvelles possibilités d'automatisation d'une grande quantité d'information sur le Web.

Le Web sémantique constitue le point de départ pour le développement des services Web intelligents. Le terme service Web sémantique se trouve à la convergence de deux domaines de recherche importants concernant les technologies de l'Internet ; le Web sémantique et les services Web.

Cette tâche de convergence est accomplie en rendant les services Web auto-exploitable par machines, et de réaliser l'interopérabilité entre les applications via le Web en vue de rendre le Web plus dynamique.

L'objectif visé par la notion de services Web sémantiques est de faciliter les tâches liées à leur utilisation, telles que la découverte, la sélection, l'orchestration et l'invocation, par le biais de leurs descriptions qui rendent la sémantique explicite et compréhensible par les machines. Ainsi, le domaine des services Web sémantiques se situe au croisement du Web sémantique et des services Web. De nombreux langages et architectures sont proposés afin de décrire les services Web sémantiques. Nous proposons ci-dessous une classification des approches existantes, qui structure notre présentation des approches de description sémantique pour les services Web [10].

### 1.4.1 Classification et présentation des approches

La réalisation des conditions qui élèvent les services Web au rang de services Web sémantiques peut suivre deux approches :

- La première approche consiste à développer un langage complet qui décrit les services Web ainsi que leur sémantique d'un seul bloc
- La deuxième approche consiste à annoter les langages existants avec de l'information sémantique. L'avantage principal de ce genre de solutions réside dans la facilité pour les fournisseurs de services d'adapter leurs descriptions existantes aux annotations proposées.

Nous classons donc ces approches de la manière suivante : dans un premier temps, nous étudions les langages de description sémantique, puis dans un second temps nous détaillons les annotations de langages existants.

### 1.4.1.1 Langages de description sémantique

#### Ontology Web Language for Services (OWL-S)

OWL-S [19] désigné par DAML-S<sup>11</sup> (DARPA Agent Markup Language Service) dans les versions antérieures est une ontologie de haute niveau de description des services web, dont les objectifs sont de résoudre les ambiguïtés et de rendre la description d'un service compréhensible par une machine. Les auteurs présentent une ontologie pour les services web dans le but d'automatiser la découverte, l'invocation, la composition et la surveillance de l'exécution des services. Ces auteurs reprennent la notion de classes d'OWL et proposent l'ontologie OWL-S. OWL-S fournit trois connaissances essentielles sur les services Web, La Figure 1.5 représente ces trois concepts :

- Le « **service profile** » a pour objectif d'automatiser la découverte et la sélection, il fournit une description de haut niveau d'un service et de son fournisseur.
- Le « **service model** » définit le flux de contrôle et données d'une composition de services.
- Le « **service grounding** » décrit les moyens d'accès au service en spécifiant le protocole de communication, le format des messages, l'encodage des paramètres.

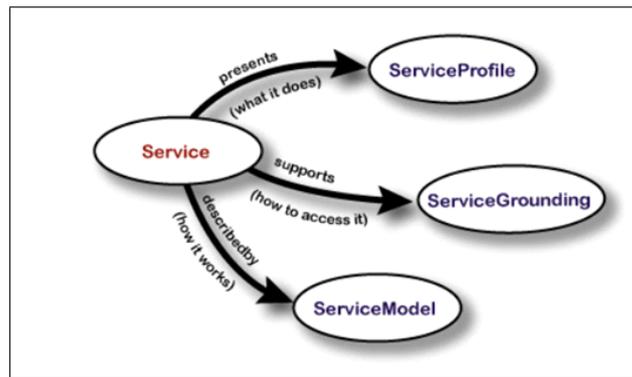


FIGURE 1.5 – Niveau supérieur de l'ontologie d'OWL-S.

#### Web Service Modeling Ontology (WSMO)

L'architecture Web Service Modeling Ontology (WSMO) [20], proposée par le laboratoire DERI, est une architecture conceptuelle, ou métamodèle, visant à expliciter la sémantique des services Web. Elle est organisée autour de quatre éléments principaux :

- **Les services Web** fournissent une fonctionnalité. Une description est associée à chaque service, dans le but de décrire sa fonctionnalité, son interface, et ses détails internes.
- **Les Objectifs** servent à décrire les souhaits des utilisateurs en termes de fonctionnalités requises. Les objectifs sont une vue orientée utilisateur du processus d'utilisation des services Web, ils sont une entité à part entière dans le modèle WSMO. Un objectif décrit la fonctionnalité, les entrées/sorties, les préconditions et posts conditions d'un service Web.

11. <http://www.daml.org/services/>

- **Les Médiateurs** sont utilisés pour résoudre de nombreux problèmes d'incompatibilité, telles que les incompatibilités de données dans le cas où les services Web utilisent différentes terminologies, les incompatibilités de processus dans le cas de la combinaison de services Web, et les incompatibilités de protocoles lors de l'établissement des communications.
- **Les Ontologies** fournissent la terminologie de référence aux autres éléments de WSMO, afin de spécifier le vocabulaire du domaine de connaissance d'une manière interprétable par les machines.

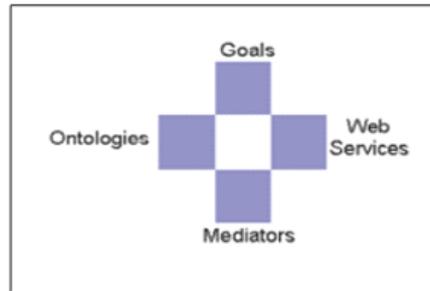


FIGURE 1.6 – Les éléments fondamentaux de WSMO.

#### 1.4.1.2 Annotation des langages existants

L'annotation sémantique consiste à enrichir et à compléter la description d'un service. Elle établit des correspondances entre des éléments de la description et des concepts d'un ensemble d'ontologies de référence. Une ontologie de référence permet de représenter un domaine par des structures interprétables par une machine. Trois modèles principaux suivent l'approche d'annotation sémantique, à savoir WSDL-S, SAWSDL et METEOR-S. Les deux premiers modèles permettent d'annoter manuellement une description WSDL avec des éléments faisant référence à des ontologies tandis que le dernier permet de générer les annotations de l'interface d'un service à partir des annotations du code source de son implémentation.

#### L'approche WSDL-S

WSDL-S (Web Service Description Language Semantic) [21] est une spécification commune à IBM et au laboratoire LSDIS. Elle a été soumise au W3C en 2005. Le WSDL-S permet de fournir un processus d'annotation sémantique compatible avec les technologies existantes. À cette fin, elle décrit un mécanisme pour lier ce modèle conceptuel avec une description fonctionnelle syntaxique capturée par le WSDL. En utilisant les éléments d'extensibilité du WSDL.

WSDL-S distingue quatre modèles sémantiques à savoir :

- InputSemantic** : Le sens des paramètres d'entrée ;
- OutputSemantic** : Le sens des paramètres de sortie ;

**Precondition** : Un ensemble d'états sémantiques qui doivent être vrais afin d'invoquer une opération avec succès ;

**Effect** : Un ensemble d'états sémantiques qui doivent être vrais après qu'une opération réalise son exécution.

### L'approche SAWSDL

L'initiative du groupe de travail d'annotations sémantiques pour WSDL (Semantic Annotations for WSDL Working Group<sup>12</sup>), SAWSDL (Semantic Annotations for WSDL and XML Schema) est une recommandation du W3C depuis août 2007. Ce groupe de travail prend part aux activités du W3C portant sur les services Web. L'objectif de SAWSDL est d'ajouter de la sémantique à la description WSDL des services et des schémas XML.

SAWSDL est la suite de WSDL-S (Web Service Description Language – Semantic). SAWSDL définit un mécanisme d'annotation en vue de spécifier les éléments de WSDL à l'aide d'ontologie(s). Cette annotation repose sur la définition d'attributs étendant le standard de description. Les annotations sémantiques référencent des ontologies pré existantes. Le mécanisme d'annotation de SAWSDL est indépendant de tout langage de représentation d'ontologies.

SAWSDL propose deux sortes d'annotations sémantiques : une première pour identifier le concept sémantique (représentée par l'attribut `model Reference`) et une seconde pour faire le lien entre le concept et le document WSDL (représentée par les attributs `lifting Scheme Mapping` et `lowering Scheme Mapping`).

SAWSDL permet d'ajouter facilement une description sémantique aux descriptions WSDL déjà existantes. L'avantage de cette proposition est qu'elle permet de modifier des descriptions WSDL existantes sans trop alourdir le fichier de description [22].

### L'approche METEOR-S

Le projet METEOR-S [23] initié par le laboratoire LSDIS vise l'intégration des technologies de services avec celles du web sémantique. Il ne fournit pas un nouveau modèle de représentation des services, mais propose de générer des annotations sémantiques pour les fichiers de type WSDL.

METEOR-S est basé sur le projet METEOR qui signifie "Managing End To End Operations". Ce dernier traite de la gestion des workflows de processus à grande échelle dans des environnements hétérogènes. L'idée principale de METEOR-S est de générer une annotation sémantique de l'interface d'un service à travers des références à une ontologie ajoutée manuellement dans le code source de l'implémentation du service.

---

12. <http://www.w3.org/2002/ws/sawSDL/>

## 1.5 Linked Open Data

Le web sémantique peut faciliter les démarches d'apprentissage en aidant à faire face à la multiplicité et à la complexité des données à traiter. Il offre une extension au web actuel, pour que l'accès aux données pertinentes soit facilité par des automatismes. Il organise et structure l'énorme quantité d'informations contenues dans le web.

LOD est de plus en plus important dans les domaines de la gestion state-of-the-art informations et des données. Il est déjà utilisé par de nombreux organismes bien connus, produits et services pour créer des portails, plates-formes, des services et applications Internet. LOD est indépendante du domaine et pénètre divers domaines et domaines, prouvant ainsi son avantage sur la gestion traditionnelle de données.

### 1.5.1 Définitions

Les Linked Open Data (LOD) sont des données ouvertes et liées, publiées dans le respect des normes du web sémantique. Le terme Linked Open Data résulte de la fusion de deux concepts distincts : Linked Data et Open Data. Premièrement, Linked Data est un terme créé en 2006 par Tim Berners Lee dans le contexte du web sémantique.

Berners-Lee a présenté le Web sémantique pour la première fois en 2001 [24]. Il voulait créer un web qui permet aux ordinateurs et aux logiciels de travailler en coopération avec les personnes et de développer la connaissance humaine, au moyen de l'interprétation automatique de données structurée. Trop complexe et utopique, l'initiative n'a pas tout de suite obtenu le succès escompté. Le phénomène prit réellement de l'ampleur en 2006, lorsque le célèbre informaticien apposa une nouvelle étiquette au web sémantique, les Linked Data, dans le but de populariser le concept en le présentant de manière plus pragmatique.

Créer des Linked Data consiste à utiliser les standards du web sémantique pour établir, entre plusieurs jeux de données, des liens définis, c'est-à-dire des liens ayant une signification. Par exemple, un lien défini décrira l'entité Berne comme étant la capitale de l'entité Suisse d'une manière reconnue universellement, y compris par des machines, ce qui lui donne un potentiel énorme. A l'inverse, dans le web traditionnel, la nature d'un lien hypertexte entre deux documents est à déduire par le lecteur humain. Basées sur le standard RDF, les Linked Data ont pour but de fournir des données interopérables pouvant être traitées automatiquement par des ordinateurs.

Le second concept est l'Open Data, ou données ouvertes en français. Il est défini ainsi par l'Open Knowledge Foundation (2012) : « Une donnée ouverte est une donnée qui peut être librement utilisé, le but des Open Data est d'arriver à une meilleure transparence. » [25]

L'Open Data s'inscrit dans une mouvance plus générale d'ouverture des connaissances, incluant

entre autres l'Open Source (pour les logiciels), l'Open Innovation (pour la recherche) ou encore l'Open Access (pour les ressources d'information). Le concept est par ailleurs étroitement lié à celui de l'Open Government Data, qui concerne les données produites par des institutions de droit public. L'interopérabilité technique et la puissance sémantique des Linked Data, couplées à la libre utilisation juridique et la gratuité de l'Open Data, forment la puissance du concept de Linked Open Data. L'aspect ouvert des données permet ainsi au web sémantique de se populariser et facilite la création de données liées en donnant accès à des jeux de données externes de manière gratuite [26].

### 1.5.2 Les quatre principes des Linked Data

Berners-Lee introduit le terme Linked Data dans un article publié en 2006. Il en définit quatre principes de base pour relier les données :

- Utiliser des URIs pour désigner les ressources.
- Utiliser des HTTP URIs pour rendre les ressources référençables.
- Fournir les données utiles pour chaque URI en utilisant des standards (RDF, SPARQL).
- Inclure des liens vers d'autres URIs pour favoriser la découverte.

Avant 2006, de nombreux jeux de données n'étaient pas liés, car ils ne respectaient pas l'un ou l'autre de ces principes. Le but de l'auteur est donc de communiquer une base technique sur laquelle chacun devrait s'appuyer pour créer des Linked Data interopérables, fonctionnelles et utilisables de manière optimale [27].

### 1.5.3 Les cinq étoiles des Linked Open Data

En 2010, Berners-Lee a complété ses quatre principes en y ajoutant le modèle des cinq étoiles des LOD .

★	disponibles sur le web (quel que soit le format), mais sous licence ouverte, pour de l'Open Data
★★	structurées pour le traitement automatique par des ordinateurs (un tableau Excel plutôt qu'un tableau scanné)
★★★	disponibles sous un format non-propriétaire (CSV plutôt qu'Excel)
★★★★	décrites selon les standards ouverts du W3C (RDF et SPARQL), pour que les gens puissent s'y référer
★★★★★	reliées à des données externes afin d'être contextualisées

FIGURE 1.7 – Les cinq étoiles des Linked Open Data.

L'interopérabilité technique des Linked Data, décrite dans les quatre principes, est ainsi complétée par l'aspect d'ouverture des données. Ce schéma vise notamment à encourager les gouvernements à publier leurs données (Open Government Data) [27].

## 1.6 Conclusion

Pour conclure ce chapitre, nous avons présenté dans celui-ci les différentes étapes d'évolution du web allant des applications web, en passant par le web sémantique et les services web, jusqu'à à l'arrivée les services web sémantique.

Comme nous le constatons, dans le présent travail, les services Web sémantiques s'intéressent à l'automatisation de l'utilisation des services Web, en ajoutant des annotations à ces services afin d'améliorer l'expression sémantique de leurs fonctionnalités. Pour cela, dans le chapitre suivant, nous présentons ce concept d'annotation sémantique des services web.

# *Annotation sémantique*

## 2.1 Introduction

Pour de nombreux domaines, le Web et ses technologies associées sont devenues la plus grande source d'information actuelle. Mais la spécificité de telles sources d'informations les rend difficilement exploitables, et leur évolution constante rend complexe les techniques de recherche d'informations. La raison principale à ça est que les documents sont fragmentés, dispersés, hétérogènes et sont souvent très peu structurés. Il est donc, nécessaire de proposer des méthodes et des outils permettant de partager, manipuler et rechercher au sein de tels documents. L'annotation sémantique est actuellement la méthode la plus pertinente et la plus prometteuse pour pallier aux problèmes de volatilité et d'hétérogénéité des documents sur le Web sémantique. En faisant appel à une ontologie, on assure un meilleur partage et une meilleure interprétation de ces annotations.

Alors, dans ce chapitre, nous présentons l'annotation sémantique son processus ainsi que ses langages. Nous abordons aussi les approches d'annotation sémantique basée sur l'utilisation d'une ontologie.

## 2.2 Définition de l'annotation sémantique et ses objectifs

L'annotation sémantique enrichit le texte en explicitant une partie du contenu suivant un besoin précis de description. C'est une compréhension automatique partielle du contenu textuel qui crée des méta-données exploitables par des traitements ultérieurs. Si l'être humain utilise la langue naturelle pour transmettre des connaissances en produisant des textes, l'annotation sémantique automatique a pour vocation d'accéder au moins à une partie de ces connaissances selon le prisme d'une catégorisation prédéfinie à défaut de pouvoir comprendre le texte entièrement.

L'annotation sémantique a pour objectif de rendre possible de nouvelles méthodes d'accéder aux informations, telles que l'indexation et la recherche d'informations, la catégorisation, ou la génération de méta-données avancées.

l'annotation sémantique cherche à identifier des éléments textuels pertinents par rapport à des critères spécifiques qui sont liés à des catégories recherchées, ou aux points de vue. Aussi, d'un point

de vue théorique, nous pouvons considérer que l'annotation sémantique est un cas très particulier de la recherche d'informations, où la requête correspond à la catégorie qui est à annoter, et les résultats sont marqués et étiquetés dans le texte au lieu d'être extraits.

L'annotation sémantique, qu'elle soit manuelle ou automatique, se base sur un ensemble de catégories prédéfinies ou des ontologies. L'annotation sémantique s'inscrit dans un processus de gestion de connaissances qui doit gérer trois types de données : ontologies, documents et annotations [28].

## 2.3 L'annotation et le Web Sémantique

Dans le cadre du Web sémantique, l'objectif est de décrire le contenu des ressources en les annotant avec des informations non ambiguës afin de favoriser l'exploitation de ces ressources par des agents logiciels. Or, les données actuelles du Web sont encore trop souvent écrites en langage naturel, car destinées aux humains. Le langage naturel étant par essence trop ambigu, des alternatives formelles et sémantiquement explicites doivent être mises en place pour lever les ambiguïtés du langage naturel, aussi bien dans le contenu des ressources que dans leurs annotations. La tâche d'annotation pour le Web sémantique consiste donc à prendre en entrée une ressource documentaire et fournir en sortie le même contenu enrichi par des annotations sémantiques basées sur des représentations de la connaissance plus ou moins formelles [29].

## 2.4 Exploitation des annotations sémantiques

Les annotations sémantiques sont en effet exploitables pour améliorer la gestion des documents annotés. Elles peuvent être utilisées dans le cadre de la recherche et la classification de documents, le résumé automatique et généralement, pour favoriser l'interopérabilité.

L'annotation sémantique peut être applicable pour toute sorte de texte : page web, documents réguliers, champs texte dans des bases de données, etc. Aussi, l'acquisition des connaissances peut se baser sur l'extraction des dépendances plus complexes et l'analyse des relations entre entités, événements et descriptions de situations. Les deux grandes tâches pour lesquelles les annotations sémantiques sont utilisées sont la recherche d'informations et la composition de services et de documents [30].

## 2.5 Méthodes et éléments de base pour l'annotation sémantique

Les standards existants forment l'ossature technologique du web, pouvant servir dans le cadre de l'annotation sémantique et qui est :

- Le protocole HTTP : pour la transmission.

- Les langages HTML/XML et les feuilles de style : pour la présentation des résultats.
- Le langage RDF (Resource Description Framework) : pour l'expression et l'échange des méta-données, sous la forme de triplets, même si ceux-ci peuvent également être stockés dans des bases de données pour une gestion plus efficace.

Le stockage des annotations au début était dans les documents eux-mêmes (RDF), dans les URL (Uniform Resource Locator) ou dans des bases sur des serveurs centralisés et les annotations étaient typées, plus ou moins partagées et souvent textuelles. Actuellement, il existe un ensemble d'éléments qui constituent, tous à la fois, un référentiel aussi bien technique que méthodologique sur lequel l'annotation sémantique de documents web peut s'appuyer. Il s'agit de :

- Méthodes de conception de schémas de méta-donnée (thésaurus, ontologies).
- Outils et principes d'utilisation de méta-données, d'annotation et de présentation celles-ci.
- Architectures de stockage, de requêtes et de diffusion d'annotation.
- Méthodes, outils et techniques d'extraction d'information des documents (fouille de données textuelles, méthodes statistiques, propositions aux utilisateurs, etc.) [31].

## 2.6 Les Langages de l'annotation sémantique

Plusieurs langages de représentation de la connaissance, et d'ontologies ont été définis par les chercheurs du domaine de l'Intelligence artificielle. Citons notamment OntoLingua, LOOM, OCML et F-Logic. Ces langages ont été conçus à partir de différents modèles issus de théories de la logique comme les logiques de description, les prédicats du premier et second ordre, les frames, etc. Mais ils ne peuvent être utilisés tels quels dans le Web sémantique sans une certaine adaptation, notamment syntaxique. Outre HTML, de nouveaux langages ont été créés pour le Web comme XML et RDF. Puis, avec la perspective naissante du Web sémantique, ces deux ensembles ont été fusionnés pour donner naissance à de nouveaux langages pour la spécification d'ontologies orientées Web, comme RDF Schéma, DAML+OIL<sup>1</sup> et OWL. Mais il est évident que le Web sémantique ne peut exister sans une standardisation de ses langages, tel que ce fut le cas pour HTML dans le Web actuel. C'est pourquoi le World Wide Web Consortium (W3C) a participé activement à l'élaboration de ces nouveaux métalangages pour le Web par la publication des recommandations autour d'XML, RDF(S) et OWL. D'autres organismes, comme le Defense Advanced Research Projects Agency (DARPA) ou l'International Standard Organisation (ISO) ont également sponsorisé des projets d'envergure autour de ces métalangages, respectivement DAML et XML Topic Maps. Nous allons étudier quelques-uns de ces langages dans la perspective de l'annotation sémantique, à commencer par les précurseurs [32].

---

1. DAML+OIL Le langage de définition d'ontologies (DARPA Agent Markup Language) + (Ontology Interchange Language) est crée de la fusion de deux langages. Il a été introduit dans le but de surmonter les limites de RDFS en matière d'inférences. Ce langage est dédié aux définitions de classes et de types de propriétés de manière plus expressive.

## 2.7 Problèmes liées à l'annotation sémantique des services Web

Les services Web sont actuellement annotés par des moyens manuels. Annotation manuelle étant une tâche difficile, les erreurs et la consommation du temps se manifestent pour les trois raisons qui suivent :

1. Le grand nombre d'ontologies de domaines potentiels qui peuvent être utilisés pour l'annotation. Au fil du temps il est prévu que plusieurs ontologies de domaine seront disponibles pour les développeurs SWS obligeant le promoteur à rechercher manuellement pour la plupart des ontologies appropriées.
2. La grande taille des ontologies potentielles. Les ontologies peuvent contenir des centaines ou peuvent être des milliers d'entités. L'utilisation de ces ontologies de poids lourds pour l'annotation nécessite un développeur pour parcourir leurs descriptions de trouver des entités qui répondent aux différents éléments de service.
3. La grande taille des candidats de services Web. De nombreux services ont un grand nombre des éléments qui devraient être annotés.

En tenant compte de ces trois problèmes, il y a besoin pressant dans l'arène SWS des mécanismes efficaces et automatiques d'annotation des services Web. Quelques approches et outils ont été développés visant à automatiser la tâche d'annotation [33].

## 2.8 Les approches d'annotation des services Web

Plusieurs approches ont été proposées pour faciliter l'annotation sémantique des Services Web. La catégorisation des approches est peut-être formulée selon beaucoup de critères. Dans ce qui suit, nous allons diviser ces approches en : à base l'apprentissage (learning-based), à base matching (matching-based) et à base workflow (workflow definition-based) , ces catégories sont brièvement décrites comme suit :

**Les approches basées sur l'apprentissage (learning-based)** utilisent des techniques d'apprentissage pour construire automatiquement des ontologies pour l'annotation. Ces techniques fournissent des ontologies de mauvaise qualité, car ils manquent de nombreuses constructions importantes. En outre, les ontologies résultantes sont des représentations de services individuels plutôt que d'être des modèles de domaine précis et partagés.

**Les approches basées sur "Matching"** nécessitent la construction manuelle des ontologies d'application pour capturer la sémantique des candidats de services. La construction Manuel des ontologies d'application est difficile, car elle nécessite beaucoup de domaines et des techniques des connaissances. En outre, les techniques de "matching" employés ne peuvent pas effectuer un réglage précis lorsque les étiquettes des services candidats et entités ontologiques sont composées

de plusieurs mots.

L'approche basée sur "workflow" utilise "trié et éprouvé" workflows et services annotés pour tirer annotation pour de nouveaux services. Les services existants et annotés "trié et éprouvé" workflows sont difficiles à trouver dans les milieux pratiques [34].

### 2.8.1 Les approches à base Machine Learning

Heß et Kushmerick en 2003 [35], ont développé une approche pour créer automatiquement des métadonnées à partir de données de formation afin de décrire sémantiquement un service Web. Les données de formation viennent des pages HTML documentant le service et le fichier WSDL du service. Trois différents types interdépendants de métadonnées sont créés.

Le premier type est la taxonomie de catégorie qui classe les services Web. Le deuxième type est la taxonomie de domaine, qui décrit la fonctionnalité d'une opération de service spécifique comme «la recherche d'un livre» ou «interroger un calendrier aérien». La troisième taxonomie décrit les catégories sémantiques de données de "input" et "output" comme «titre du livre 'ou' aéroport de destination». Cette approche considère les annotations "inputs", "outputs", la catégorie et la fonctionnalité seulement.

Lerman et autres en 2006 [36], proposent une approche qui utilise les connaissances qui existe dans les modèles de domaine pour former la sémantique des données de service Web représentées dans les documents WSDL. Le système démarre en interrogeant un modèle de domaine pour le peupler avec les instances de tous les types sémantiques. Deux classificateurs sont ensuite utilisés. Le premier classificateur, qui est basé sur les méta-données, prévoit les types des inputs à l'aide des concepts tirés de documents WSDL de données. Alors que le second classificateur, qui est basé sur le contenu, prédit la sémantique des données d'output après invocation de service Web avec des données des inputs correctes.

Cette approche a une étape de vérification pour garantir une prédiction correcte de données d'entrée (input) et génération de données de sortie (output), mais présente certains inconvénients. Premièrement, le système ne trouve pas de métadonnées sémantiques appropriées pour une entrée (input) si un type sémantique appropriée n'existe pas dans le modèle de domaine sélectionné. Dans ce cas, l'entrée (input) au-dessous de considération est laissée sans annotation. Par conséquent, un service annoté en utilisant cette approche est susceptible d'avoir de nombreux éléments qui ne sont pas annotés. Ce dernier problème est appelé le «problème de faible pourcentage". Deuxièmement, la recherche de métadonnées sémantiques appropriée devient chère quand le service Web a plus de deux entrées (input).

Emil et autres en 2010 [37], proposent un framework utilisant les techniques de machine learning basé sur un prolongé modèle Hierarchical Self-Organizing Maps. Cette approche représente

non supervisé "top-down" réseaux de neurone adopté par le modèle pour la classification de "datatype" de service, les inputs et outputs des messages des services web sont codés d'une façon vecteurs sac de mots, en tenant compte de ces mots. Ces vecteurs sont employés comme donnée d'entrée (input) pour "unsupervised neural network" ils ont expérimenté ce modèle d'annotation sémantique automatique avec SAWSDL-TC test collection de service, un ensemble de données utilisé benchmarks pour évaluer la performance de SAWSDL des algorithmes service de matchmaking.

**LI Yuan-jie et CAO Jian en 2012** [38], proposent un framework de classification des services Web basés sur l'annotation sémantique automatique et l'ensemble Learning, après filtrage de fichiers WSDL de l'Internet, ils sont automatiquement annotés sémantiquement par l'outil qu'ils ont développé appelé AutoWSDLAnnotation. Il contient 3 modules : d'abord, analyser les fichiers de WSDL et obtenir les termes des interfaces et des variables, deuxièmement obtenir d'annotation sémantique de l'information par le traitement de langage naturel. Troisièmement, obtenir l'annotation sémantique standard par mapping les termes aux concepts de l'ontologie. Les données (data) après annotation ont été écrites dans la base de données, et classifiées par l'algorithme ensemble Learning par WEKA en utilisant trois méthodes de classification : Naïve Bayes, SVM et REPTree.

## 2.8.2 Les approches à base Workflow

**Belhajjame et autres en 2008** [39], proposent un Framework pour annoter automatiquement des services de Web basés sur workflow « tried-and-tested ». Des contraintes sur l'annotation des paramètres d'opération de service Web sont impliquées, ils ont basé sur leurs liens à d'autres paramètres d'opération annotés dans workflow. Les auteurs affirment que leur approche est efficace pour détecter les erreurs dans les annotations existantes. Si un workflow produit des résultats corrects alors les paramètres des opérations liées sont sémantiquement compatibles. Préexistante les services Web annotés et workflow « tried-and-tested », qui ne sont pas toujours disponibles, sont des conditions préalables pour que cette approche fonctionne.

## 2.8.3 Utilisation de "Ontology Matching" pour l'annotation semi-automatique des services Web

Cette catégorie des techniques d'annotation utilise l'existantes d'ontologies de domaine partagé pour l'annotation plutôt que de développer de nouveau. Exécution de la semi-automatisation de l'annotation désirée n'exige que les deux processus suivants :

1. Les fichiers WSDL des services avec leurs XSD doivent être représentés en utilisant des ontologies d'application.
2. Des ontologies de domaine partagé et les ontologies d'application sont appariées en utilisant des techniques de "matching" de l'ontologie. En raison du rôle central des techniques en automatisant le processus d'annotation correspondant.

### 2.8.3.1 Ontology Matching

Les ontologies peuvent être hétérogènes au syntaxique ainsi que les niveaux sémantiques. Les différences syntaxiques résultent de l'utilisation des langues différentes. Ce problème peut être résolu en utilisant la même langue ou la traduction d'une ontologie d'une langue dans une autre. Alors que mal "matching" sémantiques se produisent en raison de différences de terminologie, l'interprétation ou la conceptualisation entre ontologies développées pour le même univers de discours.

Hétérogénéité d'ontologie empêche les systèmes qui utilisent des ontologies différentes de communiquer. La solution à ce problème est "ontology matching".

L'"ontology matching" permet de relier le vocabulaire des deux ontologies qui partagent le même domaine de telle manière que la structure mathématique de signatures ontologiques et leurs interprétations projetées. Le processus des techniques de mesure de similarité est utilisé pour effectuer une correspondance entre les deux ontologies. Les résultats des techniques individuelles doivent être combinés de manière appropriée pour donner un score global de "matching" des entités considérées [40].

### 2.8.3.2 Les approches d'annotation semi-automatique basée sur "matching"

Le Framework d'annotation de service web METEOR-S (MWSAF) qui fait partie du projet METEOR-S<sup>2</sup> est développé par **Patil et autres en 2004** [41] pour ajouter la sémantique aux documents WSDL de services Web. MWSAF utilise des techniques de "ontology matching" afin d'annoter semi-automatique les documents WSDL avec des concepts appropriés à partir des ontologies de domaine. Le Framework suggère une transformation des ontologies de domaine et le schéma XML de documents WSDL dans une représentation commune appelée "SchemaGraph" afin de permettre "structural matching". Une fois une représentation commune est atteinte, chaque concept dans le graphique WSDL est comparé à chaque concept dans le graphique domaine de l'ontologie en utilisant deux techniques de "matching". Les deux techniques sont :

Technique de "matching" basée sur les chaînes de caractères qui utilise "N-Gram" comme un mécanisme basé sur une chaîne, et technique de "matching" basé sur la structure qui utilise la similarité comme un mécanisme linguistique. Ce Framework souffre de quelques limitations :

- XSD et OWL transformation graphique de schéma est effectuée par des moyens manuels qui rend cette transformation difficile à réaliser.
- Cette approche peut mesurer les similarités entre les étiquettes contenant des CN ( Compound Nouns ) en utilisant des mécanismes de similarité de base qui ignorent la structure linguistique de NC ( Compound Nouns ).
- Ce Framework devient coûteux en calcul lorsque le nombre de candidats d'ontologies augmente.

---

2. <http://lsdis.cs.uga.edu/projects/meteor-s/>

Le processus de génération d'OWL-S commence par traduire manuellement le schéma XML d'une description WSDL dans une ontologie OWL intermédiaire. Cette transformation utilise des règles telles que :

1. Les éléments complexes et simples de XSD sont convertis en Owl : Class.
2. L'élément local de type simple est traduit en Owl : DatatypeProperty. Ensuite, l'ontologie intermédiaire est mappée à des ontologies de domaine existant en utilisant des mesures basées sur le nom et la similarité structurelle.

Le résultat de mapping est utilisé pour générer la description OWL-S souhaiter dans le fichier WSDL. Les règles de mappage de WSDL à OWL-S sont données comme suit :

- Un type de port WSDL devient un modèle de processus en OWL-S.
- Une opération dans WSDL est mappée à un processus atomique en OWL-S.
- Les inputs et outputs des messages d'une opération correspondent aux inputs et outputs d'un processus atomique OWL-S.
- Un message WSDL "part type" devient un paramètre OWL-S pour cette partie du message.

Il est à noter que l'approche a quelques inconvénients :

1. L'implémentation basée sur les noms mis en œuvre utilise "Levenshtein Distance" pour mesurer les similarités entre les étiquettes contenant des termes simples seulement. Bien que relativement efficace, cette approche ne peut pas mesurer les similarités entre les étiquettes qui ont CNs.
2. L'approche nécessite construction d'ontologie manuelle qui est une tâche difficile.
3. De nombreux éléments de service pourraient se retrouver sans annotation puisque cette approche utilise un ensemble limité d'ontologies et n'utilise pas un mécanisme efficace d'extension de l'ontologie.

**Salomie et autres en 2008** [42], ont proposé de combiner l'algorithme de "sting matching" et le concept de "Levenshtein distance" pour atteindre un niveau élevé de justesse dans la classification des concepts des ontologies. L'utilisation du processus d'annotation semi-automatique contribue également à améliorer le choix du concept matching le plus approprié.

L'outil d'annotation SAWS reçoit en entrée un document WSDL syntaxiquement décrit un service Web et un ensemble d'ontologies qui sont utilisés pour le processus d'annotation, le processus d'annotation dans SAWS se compose de trois étapes : Dans la première étape, le document WSDL est annoté avec les références à l'ontologie d'entrée et au langage de Description SAWS. Dans la deuxième et troisième étapes, les éléments WSDL, input, output et les opérations sont annotés avec des concepts de l'ontologie. Le résultat du processus d'annotation est un fichier WSDL enrichi avec les nouvelles balises et les attributs. L'outil de SAWS se compose de deux éléments principaux : "Manual Matcher" et "Automatic Matcher", "Automatic Matcher" effectue la correspondance entre les éléments de WSDL et les concepts ontologiques. "Manual Matcher" est utilisé

pour fournir une flexibilité relative à l'utilisateur, en lui permettant de sélectionner l'une des annotations fournies par l'"Automatic Matcher".

**Ting-Xin et autres en 2008** [43], utilisent l'algorithme de "matching" pour implémenter semi-automatique et annotation automatique basées sur SAWSDL, et ils utilisent "Service Template" pour archiver le calcul du degré de "matching", et de choisir le meilleur services Web selon les résultats de l'évaluation.

Ils développent un Framework de services Web auto composition, son idée principale est : annoter le document WSDL selon SAWSDL spécification, c'est-à-dire établir des relations de mapping entre l'ontologie de domaine et les éléments du document WSDL, et ajoutant quelques étiquettes de concepts d'ontologie dans le document WSDL. Ils utilisent un Framework NWSAF API, ils entrent un document annoté de SAWSDL et son fichier d'ontologie correspondant, puis calculent le degré de matching entre le concept d'ontologie et Schéma de WSDL avec l'algorithme matching .

**Zhen et autres en 2013** [44], proposent une approche de l'annotation sémantique pour les services Web en utilisant les ressources Dbpedia. Les principales étapes de l'approche proposée sont :

- La première étape de l'approche proposée implique l'analyse des services Web (parsing web service). Cette étape comprend deux tâches :
  1. La validation des fichiers WSDL qui sont significatifs pour les utilisateurs.
  2. L'acquisition des éléments dans trois niveaux de chaque service Web comme décrit dans le modèle (Niveau de service, Niveau de l'interface, Niveau de paramétrage).
- L'étape suivante porte sur les paramètres, y compris le paramètre de raffinement et de nettoyage. À partir du modèle mentionné ci-dessus, le niveau de paramétrage est fixé sémantiquement avec les instances DBpedia.
- Les prochaines étapes impliquent l'instance matching et ontology mapping. Ces processus consistent à prendre chaque paramètre WSDL de service Web et en l'associant au concept approprié dans le DBpedia ontologie, à l'aide du Spotlight DBpedia. Pendant tout processus, le vocabulaire d'annotation offre l'annotation record préalable qui peuvent être utiles lorsque le traitement avec le paramètre que nous avons annoté avant.

## 2.9 Conclusion

Dans ce chapitre, nous avons présenté les principales composantes du web sémantique à savoir les ontologies et les annotations. Sous cette optique les ontologies se présentent comme un pilier du web sémantique.

SAWSDL fournit un mécanisme d'annotation des capacités et des besoins des services Web avec des concepts sémantiques référencés à partir d'un modèle sémantique. Ce processus d'annotation consiste en deux phases, à savoir, la catégorisation et le matching : la catégorisation permet d'attribuer chaque service Web à son domaine correspondant ; le matching permet de relier chaque élément WSDL avec son concept correspondant dans l'ontologie de domaine.

Dans le chapitre qui suit nous allons présenter les annotations des services Web en utilisant des ontologies de domaine créé à partir des données ouvertes liées (Linked Open Data).

Approches	Technique	Categories	Les éléments consideres	Outils	Avantages	Inconvenient
[Heß and Kushmerick, 2003]	Naive Bayes, SVM and HyperPipes.	Machine Learning.	Données de formation (formulaire de HTML), operation et input, output.	Weka.	* Créer des métadonnées. * Classification des services web .	*
[Lerman et al., 2006]	Naive Bayes Logistic Regression.	Machine Learning.	Operation, input, output et dataType.	*	Classification des services web.	Prévoit les types des inputs à l'aide des concepts tirés de documents WSDL de données.
[Emil et al.,2010]	Self-Organizing Map.	Maching Learning.	*Input et output. * datatype.	SAWSDL-TC.	Classification des services web.	*
[ LI Yuanjie et CAO Jian,2012]	Naïve Bayes, SVM, REPTree et ensemble learning(bootstrap).	Machine Learning.	Input, output.	AutoWSDL Annotation Tool : WEKA.	Classification des services web : Naïve Bayes, SVM and REPTree and ensemble learning ( Boosting).	*
[Patil et al., 2004]	N-Gram , structural matching, String matching.	Ontology Matching.	schéma XML.	MWSAF.	Classification des service web.	XSD et OWL transformation graphique de schéma est effectuée par des moyens manuels. Ce Framework devient coûteux en calcul lorsque le nombre de candidats d'ontologies augmente.
[Duo et al. , 2005]	Structural matching, String matching.	Ontology Matching.	Eléments complexes et simples de XSD.	*	Annotation de service web.	Transformation manuellement de schéma XML d'une description WSDL dans une ontologie OWL.
[Salomie et al., 2008]	Algorithme "string matching" - "Levenshtein distance".	Ontology Matching.	Les balises <i>&lt; Input &gt;</i> , <i>&lt; output &gt;</i> , <i>&lt; element &gt;</i> et <i>&lt; operation &gt;</i> .	Framework "SAWS".	* Classification des services web et annotation les éléments, input, ouput en utilisant similarité string matching et distance de levenshtein. * opération en utilisant distance de levenshtein.	*
[Ting-Xin et al.,2008]	Algorithme "service matching".	Ontology Matching.	Input, ouput et Operation.	NWSAF API.	Calculer le degré de matching entre le concept d'ontologie et le schéma de WSDL.	*
[Zhen Zhang, et al 2013]	La similarité sémantique à base DBpedia knowledge.	Ontology Matching.	XSD data types, interface, opérations et messages.	SAWSDL generator.	L'annotation.	*

TABLE 2.1 – Tableau comparatif

## *Conception*

### 3.1 Introduction

Le Web sémantique a vu une augmentation de la disponibilité et de l'utilisation de bases de connaissances au cours des dernières années, en particulier dans l'initiative de Linked Open Data. Il se base sur deux composants principaux pour pouvoir évoluer (ontologies et annotations sémantiques) afin de s'adapter à l'environnement distribué et dynamique en perpétuel changement. L'ontologie et l'annotation sémantique jouent un rôle important dans un Web sémantique. Les ontologies sont nécessaires pour les représentations sémantiques nécessitant un certain consensus de communauté. Les annotations sémantiques consistent à ajouter des méta-données structurées aux ressources documentaires du Web. Cependant, ces deux composants sont souvent modifiés et évoluent afin de s'adapter aux nouveaux besoins ou exigences.

Actuellement, la plupart des travaux d'annotation se basent sur des ontologies de domaine existantes avec différentes techniques d'annotation. Nous abordons dans ce chapitre la création des ontologies de domaine à partir des données ouvertes liées (Linked Open Data), ensuite l'utilisation de ces ontologies pour annoter les services web en utilisant le mécanisme d'annotation du SAWSDL.

### 3.2 Architecture de l'annotation sémantique des services web à partir des données ouvertes liées(LOD)

Les services Web fournissent un modèle simple de programmation et de déploiement. La sémantique dans les services Web a permis pour certains services que l'on a adoptés dans leur description de faire évoluer leurs mécanismes de découverte, l'invocation et la composition au-delà des mécanismes syntaxiques habituels. C'est le cas pour WSDL. Le manque de sémantique des services Web présente la principale difficulté devant une utilisation efficace de ce standard de services Web pour la découverte et la composition automatiques.

SAWSDL reste une approche indépendante du langage de représentation sémantique grâce à la séparation entre les mécanismes d'annotation sémantique et la représentation de la description sémantique. Sans un tel mécanisme, les développeurs n'ont pas assez de flexibilité pour choisir

leurs langages de représentation sémantique favoris, ou pour annoter les descriptions avec diverses ontologies.

Les ontologies sont utilisées pour formaliser la connaissance dans le Web sémantique. Elles sont représentées par un langage qui permet de spécifier une conceptualisation, définie comme une version simplifiée d'un domaine que nous voulons représenter de façon formelle en utilisant des concepts et leurs relations. Un des objectifs des ontologies est la facilitation des échanges de connaissances entre les humains, entre humain et machine ou entre machines.

L'approche proposée par **Zhen et al en 2013** [44] permet l'annotation sémantique des services Web en utilisant les ressources "DBpedia". Cependant, nous proposons tout d'abord la création de l'ontologie de domaine à partir des données ouvertes liées (Linked Open Data), ensuite utiliser cette ontologie pour annoter les services web en utilisant le mécanisme d'annotation du SAWSDL.

La figure 3.1 illustre l'architecture du système proposé qui comprend deux étapes principales, une étape de construction d'une ontologie à partir des données ouvertes liées et une autre étape d'annotation sémantique de service web par l'ontologie construite. L'étape de construction d'ontologie concerne la création et l'enrichissement des ontologies "semi-automatique" afin d'enrichir les concepts d'ontologie schéma en s'appuyant sur les instances disponibles dans le nuage Linked Open Data. L'étape d'annotation sémantique comprend le processus d'associer chaque concept d'ontologie avec les concepts de service web, et ce processus se compose de quatre grandes phases, à partir de chargement de service web et l'ontologie jusqu'à l'annotation.

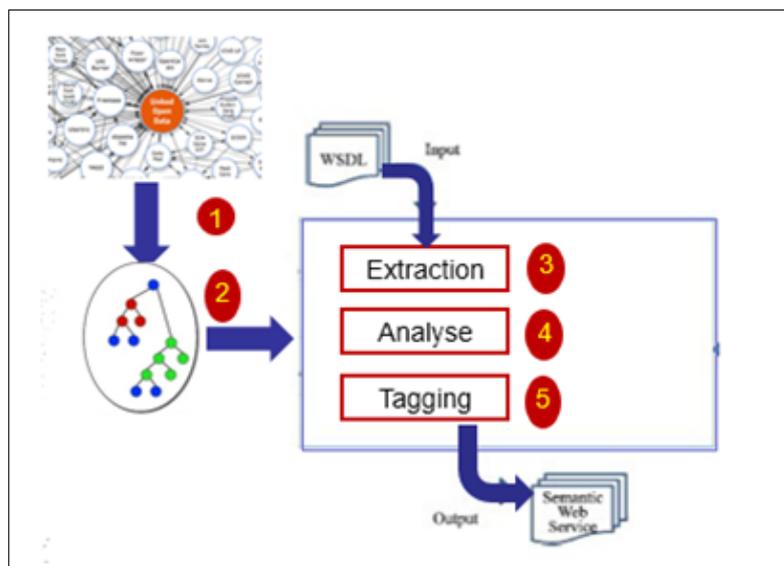


FIGURE 3.1 – Architecture générale de l'annotation sémantique des services web à partir des données ouvertes liées(LOD).

### 3.3 Création des ontologies à travers les données ouvertes liées (Linked Open Data)

De nombreuses ontologies ont été récemment créées dans le domaine des sciences biomédicales [45]. Beaucoup d'entre eux ont été inclus dans les catalogues de l'ontologie bien connus, tels qu'"OBO Foundry" [46] et "BioPortail" [47]. Cependant, la création d'une ontologie de domaine est une tâche difficile et fastidieuse.

La population automatique des ontologies de domaine par l'extraction de données à partir des documents textuelles est difficile. Nous supposons que la grande quantité de données distribuées disponibles dans "LOD" offre une grande occasion pour les experts du domaine pour créer de haute qualité, une nouvelle ontologie de domaine peut être créée par la définition de ses classes et propriétés, puis précisant comment obtenir des instances en termes des concepts et propriétés dans les ensembles de données (dataset) disponibles dans "LOD". Une ontologie créée de cette manière, non seulement favorise la réutilisation des concepts et propriétés existant, mais aussi augmente l'interconnexion de la nouvelle ontologie à partir des ensembles de données déjà existantes dans "LOD". Dans ce qui suit, nous introduisons la création d'ontologie biomédicale à partir des données ouvertes liées (Linked Open Data).

Dans cette partie, nous allons présenté un travail sur la création d'une ontologie spécifique au domaine à partir des données ouvertes liées (linked open data). Dans un premier temps, nous avons élaboré une méthodologie en travaillant à la création d'ontologie pour la représentation des termes biomédicals.

Notre corpus est composé d'une certaine de source de donnée au domaine biomédical et sa transformation en une ontologie manuellement serait une activité difficile qui nécessiterait des compétences à la fois dans la représentation des connaissances avec les ontologies et des connaissances sur les domaines couverts (pédagogie, histoire, mathématiques...etc.).

Nous proposons une méthode de création et d'enrichissement d'ontologie biomédicale "semi-automatique" afin d'enrichir les concepts d'ontologie en s'appuyant sur les instances disponibles dans le nuage Linked Open Data. Nous présentons d'une façon générale les données ouvertes liées (LINKED OPEN DATA). Puis nous donnons une description du corpus utilisé pour la création de notre ontologie. Et pour finir, nous décrivons la mise en place de notre solution.

### 3.3.1 Description des (dataset) utilisé pour la construction d'ontologie biomédicale

Les études sur les maladies sont parmi les domaines de recherche les plus critiques dans le domaine biomédical. Les informations utiles concernant divers types des maladies et les traitements peuvent être trouvés dans divers ensembles de données de "LOD", tels que Diseasome [48], LinkedCT [49], DrugBank [50], UniProt, et plusieurs autres. Néanmoins, pour le meilleur de notre connaissance, aucune de ces ontologies ne se concentre spécifiquement sur les traitements contre les maladies.

Toutes ces sources de données contiennent une grande quantité d'utiles informations relatives à une maladie et son traitement. Cependant rien qu'en utilisant les données c'est difficile. Par exemple "Diseasome" stocke des informations sur toutes sortes de maladies, tandis que "DrugBank" contient des informations sur tous les types des médicaments. Il est également difficile, car l'information pertinente dans les données de différentes sources ne sont généralement pas explicitement connectées, même s'ils sont évidemment sémantiquement liés. Par exemple, "UniProt" fournit une grande quantité d'informations sur les protéines et les gènes. D'un autre côté, "Diseasome" l'information génétique associée à chaque maladie. Toutefois, les médicaments dans "Diseasome" ne sont pas reliés explicitement à leurs équivalents dans "DrugBank".

L'expert peut choisir d'utiliser les noms de médicaments à partir de deux sources de données pour les relier et donc récupérer plus de connaissances sur cette relation. En outre, même si ce document se concentre sur les données sources disponibles dans "LOD", la façon dont chaque source de données a défini son ontologie est différente, et accéder au schéma de ces ontologies n'est pas toujours trivial. Alors que "UniProt" fournit un schéma clair et bien défini, "Diseasome", "DrugBank" et d'autres nécessitent plus d'efforts pour accéder à les données disponibles. Nous allons utiliser cet exemple que notre étude de cas pour illustrer l'utilité de notre cadre. Ceci sera décrit plus en détail plus tard.

Nom	Sujet	Description
Diseasome	Diseases/Genes .	Diseasome décrit les caractéristiques des troubles et des gènes de maladies liées.
DrugBank	Drugs.	Drugbank fournit des données avec la drogue cible de médicament complète (la séquence, la structure, et la voie) information (chimique, pharmacologique et pharmaceutique).
SIDER	Diseases/ Side Effects.	SIDER contient des informations sur les médicaments commercialisés et de leurs effets indésirables.

TABLE 3.1 – Interconnexion entre les "dataset" au domaine biomédical.

Et La figure 3.2 montre interconnexion entre les "dataset" au domaine biomédical.

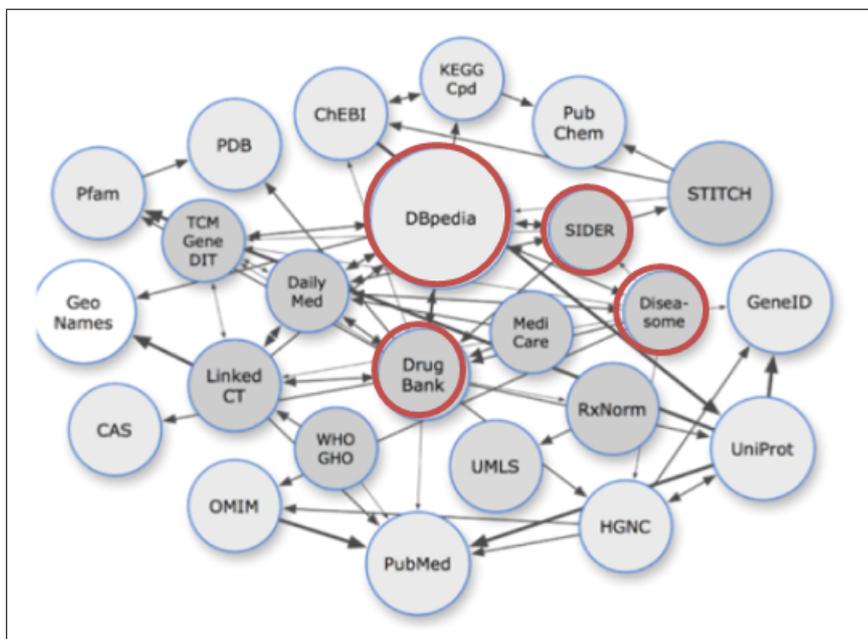


FIGURE 3.2 – Interconnexion entre les "dataset" au domaine biomédical [49].

La figure 3.3 montre un aperçu du schéma d'une telle ontologie avec les ensembles de données fournies et les données requises.

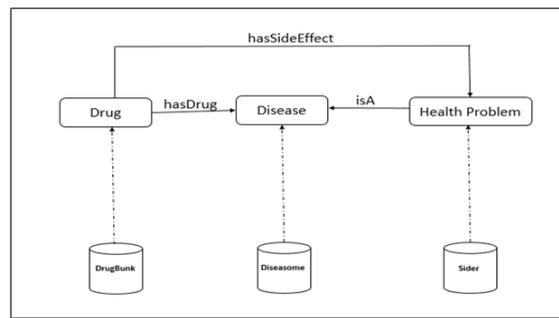


FIGURE 3.3 – schéma d'ontologie biomédicale.

### 3.3.2 Model conceptuel

On a présenté un modèle conceptuel pour la création d'ontologie spécifique au domaine à partir des ensembles de données (dataset) disponibles dans le nuage des Données ouvertes liées (linked open data) (figure3.4).

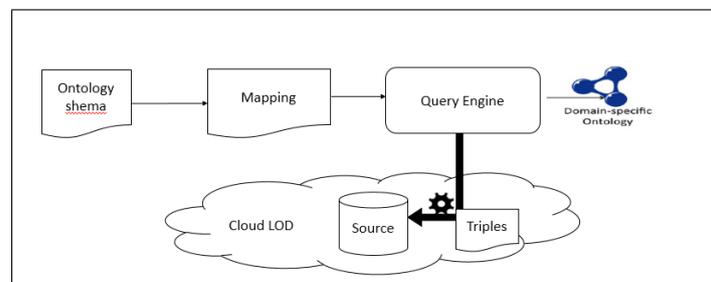


FIGURE 3.4 – Architecture générale de "mapping".

Au début, le processus de création de l'ontologie est subséquentement "mapping" avec un certain nombre des sources de données disponibles sur "LOD" qui ont été identifiées par l'expert en rapport avec le domaine de l'ontologie. Autrement dit, l'expert va créer un ensemble de "mapping" pour effectuer la population des classes et propriétés dans la nouvelle ontologie, en utilisant les instances disponibles dans les ensembles de sources de données sélectionnés de "LOD". Chaque "mapping" est une requête "SPARQL", qui est formulée pour effectuer les tâches suivantes :

- Interroger "SPARQL endpoint" pour acquérir les données d'un ensemble des sources de données de "LOD".
- Interroger la nouvelle ontologie (à savoir l'ontologie en construction) afin de connecter les données récupérées aux instances et propriétés déjà créées.
- les résultats de sortie sont une collection de triplets de la nouvelle ontologie construite.

Les requêtes de "mapping" peuvent être conçues, soit créer de nouveaux "URI" pour les entités (instances) ou réutiliser les "URI" à partir des sources de données d'origine. Autres requêtes de "mapping" peuvent étendre les descriptions des entités déjà existantes par propriétés de données

supplémentaires ou de création les objets des propriétés.

Nous avons expliqué la création d'une collection de "mapping". La nouvelle ontologie spécifique de domaine est progressivement peuplée et elle est stockée localement.

Le nombre de "mapping" nécessaire pour remplir la nouvelle ontologie dépend du nombre de classes, propriétés, et les ensembles de données sous-jacentes de "LOD", chaque "mapping" est défini par ensemble des sources des données " dataset" et prend la forme unique de requête "SPARQL" composée de trois éléments principaux décrits ci-dessous :

- La nouvelle ontologie est progressivement peuplée pour exécuter séquentiellement les "mapping". Par conséquent, après chaque "mapping" d'exécution, le nombre des instances de classe et propriété augmente et l'ontologie grandit. Basé sur le pattern utilisé pour le "mapping", nous pourrions interroger l'ontologie locale et utiliser les résultats pour interroger l'ensemble de données (dataset) à distance.
- Chaque "mapping" est conçu pour interroger un certain ensemble de données (dataset) pour récupérer les données nécessaires pour peupler une partie de l'ontologie locale. Ceci est réalisé en envoyant une requête à "SPARQL ENDPOINT" du jeu de données sélectionné. Le bloc sous service <endpointURI>est responsable de soumettre la requête à l'extrémité.
- Lorsque les données désirées sont obtenues, il devrait contribuer à peupler l'ontologie locale. Le graphique est revenu de la requête "SPARQL" inclus les triples qui devraient être affirmé dans la nouvelle ontologie. Bien sûr, les triplets devraient utiliser les vocabulaires de l'ontologie locale et d'autres espaces de noms, tels que rdf, rdfs, owl... etc. Les triplets construits sont ensuite ajoutés à l'ontologie construite.

### 3.4 L'annotation sémantique des services web par l'ontologie construite

Figure 3.5 illustre le processus d'annotation sémantique des services Web basés sur l'ontologie construite précédemment.

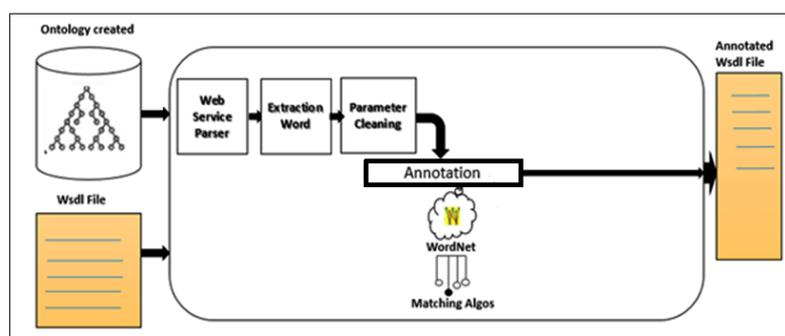


FIGURE 3.5 – Schéma d'annotation sémantique des services web basé sur l'ontologie construite.

Notre processus d'annotation sémantique des services web se déroule comme expliqué précédemment en deux grandes étapes : une étape de création d'ontologie de domaine à partir des données liées ouvertes et une étape d'annotation sémantique basée principalement sur le "matching" et qui consiste à associer chaque élément du fichier de description WSDL avec le concept correspondant dans l'ontologie de domaine.

Il faut noter que notre approche se concentre uniquement sur les services web de description WSDL, nous avons laissé en perspectives la deuxième génération des services web, à savoir les services web REST.

Dans ce qui suit, nous allons expliquer la démarche suivie et le fonctionnement du système d'annotation avec l'ontologie créée précédemment.

### 3.4.1 Le système SAWS

Notre proposition vise à l'annotation sémantique des Services Web avec l'ontologie construite à partir les données ouvertes liées(LOD).

La proposition Figure 3.5 que nous allons détailler en dessous, l'outil "SAWS" signifie Annotation sémantique des services Web. Il a été développé comme un outil semi-automatique qui permet l'annotation de services Web avec des métadonnées extraites d'ontologie de domaine construite à partir les données ouvertes liées (LINKED OPEN DATA). L'outil d'annotation "SAWS" reçoit en entrée un document WSDL syntaxiquement décrit un service Web et une ontologie qui est utilisée pour le processus d'annotation, le résultat généré par l'outil est un fichier WSDL enrichi avec les nouvelles balises et les attributs.

L'outil "SAWS" se compose de deux éléments principaux pendant l'annotation : "Manual Matcher" et "Automatic Matcher", l'outil SAWS effectue la correspondance (mapping) entre les éléments de WSDL et les concepts ontologiques en utilisant l'algorithme de similarité à base "WordNet". "Manual Matcher" est utilisé pour fournir une flexibilité relative à l'utilisateur, en lui permettant de sélectionner l'une des annotations fournies par l'"Automatic Matcher".

Les similarités entre les ensembles doivent être calculées en comparant l'ensemble des entités du fichier "Schema XML" avec toutes les entités de l'ontologie. Il est à noter que notre contribution s'articule sur le concept de calcul de la similarité sémantique qui permet d'enrichir sémantiquement les éléments de notre approche et résoudre le problème d'annotation. La section suivante fournit plus d'explications sur ce concept.

### 3.4.2 Similarité sémantique

La similarité joue un rôle très important, en particulier dans le processus de "matching". Elle se rapporte à la comparaison des éléments des documents. Elle renvoie une valeur numérique indiquant si les deux éléments de document ont un degré élevé ou bas de similarité. La notion de similarité sémantique est utilisée pour exprimer la ressemblance entre des concepts. Certaines mesures de similarité sémantique ont été proposées en utilisant les ressources sémantiques disponibles.

Une mesure de similarité vise à quantifier combien de concepts sont similaires. Pour calculer la similarité sémantique, nous avons utilisé des techniques qui se basent sur "WordNet". "WordNet" est une ressource lexicale de langue anglaise développée à l'Université de Princeton disponible en format électronique, il regroupe des termes (noms, verbes, adjectifs et adverbes) en ensembles de synonymes appelés "synsets". Un "synset" regroupe tous les termes dénotant un concept donné. Elle organise les noms et les verbes dans des concepts (synset) en hiérarchies de relations "is-a". Chaque concept est décrit par une brève glose. Les méthodes qui utilisent "WordNet" ou Thesaurus peuvent fournir des résultats excellents [51], puisque "WordNet" et thesaurus sont manuellement créés et les relations entre les mots dont les ressources sont bien explicites. Ces techniques peuvent être classifiées en trois catégories :

- Les mesures de similarité basées sur la longueur du chemin entre les concepts .

La Figure 3.6 représente la relation de deux sens quelconques  $S_1$  et  $S_2$  dans une taxinomie par rapport à leur sens commun le plus spécifique  $S_3$  et par rapport à la racine de la taxinomie ; cette figure servira à exprimer de manière homogène les formules des différentes mesures de similarité.

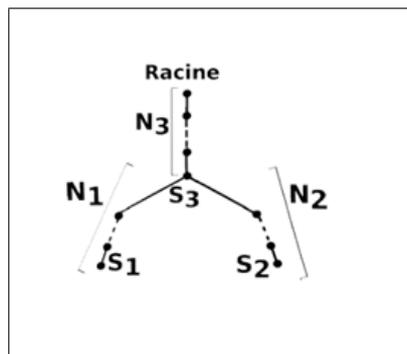


FIGURE 3.6 – Deux sens et leur sens commun le plus spécifique dans une taxinomie.

La mesure de **Rada** [52] est la première à utiliser la distance entre les noeuds correspondants aux deux sens sur les liens d'hyponymie et hyperonymie :

$$sim_{Rada}(s_1, s_2) = d(s_1, s_2) = N_1 + N_2$$

La mesure **WUP** [53] détermine la longueur du chemin vers la racine à partir du plus proche père commun (PPC) des deux concepts. Cette valeur est multipliée par la somme des deux longueurs des chemins à partir de chaque concept vers la racine, la mesure de **Wup** est définie par la formule suivante :

$$sim_{WuP} = \left( \frac{2.N_3}{N_1 + N_2 + 2.N_3} \right)$$

La mesure **LCH** [54] se base également sur la mesure de Rada, mais au lieu de normaliser par la profondeur relative de la taxinomie par rapport aux sens, ils choisissent une normalisation par rapport à la profondeur totale de la taxinomie D et normalisent avec un logarithme, elle détermine le plus court chemin entre deux concepts et multiplie cette valeur par le maximum des deux chemins là où ils apparaissent dans l'hierarchie "is-a".

$$sim_{LCH} = -\log\left(\frac{N_1 + N_2}{2.D}\right)$$

La mesure **PATH** est égale à l'inverse de la longueur du plus court chemin entre les deux concepts.

- **Les mesures basées sur le contenu d'information :**

La notion du contenu informationnel (CI) a été initialement introduite par **Resnik** [55] 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.

Les mesures **LIN** [56] et **JCN** [57] augmentent le contenu d'information du PPC des deux concepts avec la somme des contenus d'information de chaque concept. Alors que la mesure **LIN** multiplie le contenu d'information du PPC par cette somme, la mesure **JCN** fait la soustraction du contenu d'information du PPC de cette somme.

- **Les mesures basées sur le type des relations entre les concepts :**

La mesure **HSO** [58] adapte le concept de chaînes lexicales développées par Morris et Hirst en 1991 comme mesure de similarité sémantique en utilisant la structure de "WordNet". Cette mesure se base sur l'idée de Halliday et Hasan en 1976 que dans un texte, des mots ont une forte probabilité de référer à des mots antérieurs ou à d'autres concepts reliés, et que l'enchaînement de ces mots forme des chaînes cohésives. Par exemple, Navigli en 2009 Rome->ville->habitant et manger->plat->légume->aubergine, sont des chaînes lexicales. À chaque relation est associée une direction horizontale, ascendante ou descendante, qui marque respectivement une relation forte, très forte et moyennement forte (par exemple l'hyponymie est une relation ascendante, l'holonymie une relation descendante, et l'antonymie une relation horizontale). Les changements de direction constituent un élément de dissimilarité et la proximité dans la taxinomie un élément de similarité. Un changement de direction est défini comme le passage d'un élément de la taxinomie à un élément B par une relation d'un autre type que celle qui a permis d'arriver sur A. Notons que plus la distance entre les sens est grande, plus il y aura de changements de direction potentiels.

Même si l'algorithme proposé est un algorithme global, il est possible d'utiliser la fonction d'évaluation des chaînes lexicales en tant que mesure de similarité. Soient C et k deux constantes et soit la fonction virages (s1, s2) qui retourne le nombre de changements de direction entre les sens s1 et s2.

Il existe également d'autres mesures exploitant la structure taxonomique, mais en pondérant les arcs avec des valeurs de contenu informationnel, notion que nous allons maintenant définir.

La mesure **LESK** [59] calcule la similarité en déterminant et enregistrant le chevauchement entre les gloses des deux concepts, ainsi que les concepts ayant un lien direct avec eux dans WordNet.

Pour la mesure **VECTOR** [50], chaque terme utilisé dans une glose "WordNet" est associé à un vecteur de contexte. Chaque glose est représentée par un vecteur de glose qui est la moyenne de tous les vecteurs de contexte des termes trouvés dans la glose.

La similarité entre les deux concepts est égale au cosinus entre les deux vecteurs de glose. Nous nous fixons au préalable un seuil pour la distance sémantique le seuil est une valeur comprise entre 0 et 1 ; la valeur 1 indiquant que les deux entités sont totalement similaires.

Nous assumons que la mesure **WUP** [60] est utile à notre proposition et donne de bons résultats, car elle est largement utilisée dans ce domaine. Mais nous pourrions implémenter n'importe quel algorithme de calcul de mesure de similarité sémantique [61] dans l'outil SAWS.

### 3.4.3 Le standard WSDL

Le besoin d'une description claire de la communication entre services Web a abouti à des normalisations au niveau des messages échangés et des protocoles. Une grammaire XML bien structurée était proposée pour décrire les services Web et paramétrer les échanges de messages. Un langage de description s'est basé sur cette grammaire et est devenu le standard des services Web, c'est le "Web Service Description Language" (WSDL). Dans le chapitre des services Web, on a bien présenté les principes du standard WSDL et ces deux spécifications largement utilisées par les chercheurs et les industriels voir chapitre 1.

### 3.4.4 Les phases du processus d'annotation sémantique des services web par une ontologie construite à partir les données ouvertes liés(LOD)

Le processus d'annotation sémantique dans l'ensemble passe par quatre (4) phases, depuis le chargement du fichier de description du Service Web jusqu'à la production et validation.

**Phase de chargement** : Cette phase permet de charger un fichier de description WSDL et leur XML schéma et une ontologie biomédicale construite à partir des données ouvertes liées.

**Phase d'extraction** : La phase d'extraction constitue une partie importante de notre processus d'annotation, l'information utile extraite est stockée dans une liste qui sera utilisée pour les étapes suivantes. Les informations prises en charge sont :

- Les informations extraites à partir du schéma XML du fichier WSDL (element, simpleType, complexeType).
- Les informations qui se trouvent dans la balise < operation > et < message >.
- Les concepts d'ontologie construite tels que les classes et les sous classes.

Nous rappelons que l'extraction des informations se fait à partir du fichier WSDL et fichier owl de l'ontologie construite.

**Phase de nettoyage (cleaning)** : Des travaux de nettoyage sont nécessaires pour rendre les paramètres de service web dans une meilleure forme pour l'annotation sémantique, le but de cette phase est de trouver les mots clés dans une chaîne irrégulière des mots qui sont en mesure d'expliquer les paramètres le plus clairement.

Prenant en considération les noms des paramètres, beaucoup des paramètres (message, operation, element. . . etc.) du service web ont le même nom du paramètre sous la forme de "getA", ou sous la forme "setA" ou sous la forme "ARequest" ou "AResponse"."A" est le mot clé. Par exemple, le nom du message est nommé "GetDrugRequest" d'un service web, nous pouvons extraire le mot clé "Drug". Puis, nous séparons ces mots selon la majuscule de la première lettre ou espace blanc. Après la séparation, certains mots vides devraient être négligés. Les mots vides sont maintenus

dans une table dans notre base de données.

**Phase d'annotation sémantique** : L'annotation sémantique des documents WSDL est possible grâce à l'extensibilité de WSDL 2.0. En effet, conceptuellement WSDL 2.0 est doté des constructions suivantes : "interface", "opération", "message", "binding", "service" et "endpoint". Les trois premiers à savoir "interface", "operation" et "message" concernent la définition abstraite du service tandis que les trois autres sont relatifs à l'implémentation du service.

SAWSDL (Semantic annotation for WSDL ) est un langage sémantique de description de service Web. Il est évolutif et compatible avec les standards des services Web existants, et plus spécifiquement avec WSDL. SAWSDL augmente l'expressivité du langage WSDL avec la sémantique en utilisant des concepts analogues à ceux utilisés dans OWL-S. D'une part SAWSDL, fournit un mécanisme permettant d'annoter sémantiquement les types de données, les opérations, les entrées et les sorties de WSDL et d'autre part, il ajoute des éléments pour spécifier les préconditions, les effets et les catégories des services Web, cela se fait grâce à l'attribut "sawSDL". Il existe trois extensions de cet attribut. La première est "modelReference" et permet d'associer un composant WSDL ou XML Schema à un concept d'une ontologie. Les deux autres sont "liftingSchemaMapping" et "loweringSchemaMapping" et permettent de spécifier la correspondance entre les données sémantiques et les éléments XML. Les "SchemaMapping" sont utilisés pour établir la correspondance entre les structures des entrées et des sorties, et sont utiles lorsque les structures XML demandées par le client et celles fournies par le service sont différentes.

L'annotation des interfaces, opérations, entrées/sorties et les types XML simples s'effectue en leur associant un concept dans une ontologie par le biais de l'attribut "modelReference". Cependant, l'annotation des types de données XML complexes peut nécessiter en plus un "Schema Mapping". En effet, deux services Web peuvent manipuler le même type complexe, mais avec deux structures différentes.

La phase d'annotation consiste à extraire des concepts du fichier "Schema XML" (élément, types simples et les types complexes) et les comparer avec les concepts d'ontologie choisis.

Les quatre essentiels ajouts de "modelReference" sont :

- ✓ **Annotation des interfaces avec le modèle de référence.**
  - **Annotation des opérations avec le modèle de référence.**
  - **Annotation de document "schéma XML" avec le modèle de référence :** "ModelReference" définit la sémantique supplémentaire pour les composants "schéma XML".
  - **Annotation des types simples avec le modèle de référence.**
  - **Annotation des types complexes avec le modèle de référence :** Il existe deux principales techniques pour annoter les types complexes qui peuvent être résumées comme suit :

1. L'annotation à bas niveau consiste en une annotation d'un élément membre ou un attribut du type complexe.
2. L'annotation de haut niveau consiste en une annotation du conteneur du type complexe (le type complexe lui-même) :

En ce qui concerne le bas niveau, tous les éléments membres et attributs d'un type complexe peuvent être annotés. Dans certains cas, les membres d'un type complexe correspondent aux concepts dans un modèle sémantique. Dans ce cas, les éléments membres et attributs peuvent être annotés par l'ajout d'un attribut "modelReference" à l'élément du schéma ou attribut. Un type complexe peut être annoté à la fois le haut et le niveau bas. Ces annotations sont indépendantes les unes des autres.

✓ **Annotation des éléments avec le modèle de référence.**

➔ **Annotation des attributs avec le modèle de référence.**

## 3.5 Conclusion

Ce chapitre a été consacré à l'annotation sémantique des services web à base des données ouvertes liées, après avoir proposé une méthode de création et d'enrichissement d'ontologie biomédicale "semi-automatique" afin d'enrichir les concepts d'ontologie en s'appuyant sur les instances disponibles dans le nuage "Linked Open Data". Après la construction d'ontologie du domaine, viendra l'étape d'annotation sémantique de service web à base de "WordNet", ou notre system SAWS reçoit en entrée un document WSDL syntaxiquement décrit un service Web et une ontologie biomédicale construite à partir des données ouvertes liées (LINKED OPEN DATA) qui sont utilisées pour le processus d'annotation, le résultat est un fichier WSDL enrichi avec les nouvelles balises et les attributs. L'approche montre l'efficacité du processus d'annotation des applications Services Web.

Dans le chapitre suivant, nous allons passer à l'étape d'implémentation de notre système illustré par une étude de cas réel.

# *Implémentation*

## 4.1 Introduction

L'amélioration des technologies des Services Web visent à des techniques plus sophistiquées pour décrire les Services Web, mettant l'accent sur le concept de Services Web sémantiques. ces derniers peuvent être publiés, découverte, composée avec d'autres ressources et exécutée à travers le Web dans une tâche conduite de manière automatique. La technologie des Services Web Sémantiques peut optimiser plusieurs processus dans le domaine biomédicale.

Après avoir accompli la phase de l'analyse et la conception , nous entamons la dernière phase , la phase de l'implémentation de prototype. Au cours de ce chapitre nous allons mettre en œuvre les divers outils de programmation et l'environnement de développement de notre choix propre. Ensuite nous allons exposer un aperçu des différentes interfaces et le fonctionnement de notre application .

## 4.2 Outils d'implémentation

Le développement d'un tel prototype nécessite le passage par intermédiaires et l'utilisation de quelques outils. Dans ce qui suit, nous citons les outils qui ont été utilisés.

### 4.2.1 Java

Java [62] est à la fois un langage de programmation informatique orienté objet et un environnement d'exécution informatique portable créé par James Gosling et Patrick Naughton employés de Sun Microsystems avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

Java est à la fois un langage de programmation et un environnement d'exécution. Le langage Java a la particularité principale que les logiciels écrits avec ce dernier sont très facilement portables sur plusieurs systèmes d'exploitation tels que Unix, Microsoft Windows, Mac OS ou Linux avec peu ou pas de modifications... C'est la plate-forme qui garantit la portabilité des applications développées en Java.

## 4.2.2 NetBeans

NetBeans [63] est un environnement de développement intégré (IDE) pour Java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, XML et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages web).

NetBeans est disponible sous Windows, Linux, Solaris, Mac OS X et Open VMS. NetBeans est lui-même développé en Java, ce qui peut le rendre assez lent et gourmand en ressources mémoires.

## 4.2.3 Jena

Jena [64] est un framework en Java qui permet de construire des applications Web Sémantique. Il fournit un environnement de programmation pour RDF, RDFS et OWL. Il inclut également un moteur de règles d'inférence. Le framework contient :

- ✓ Un api RDF.
- ✓ Un api pour la lecture et l'écriture de RDF en RDF/XML, N3, et N-Triples.
- ✓ Un api OWL.
- ✓ Un système de persistance.
- ✓ RDQL, un langage de requête pour RDF.

## 4.2.4 L'API JDOM

JDOM [65] est une API open source Java dont le but est de représenter et de faciliter la manipulation au sens large de document XML : lecture d'un document, représentation sous forme d'arborescence, manipulation de cet arbre, définition d'un nouveau document, exportation vers plusieurs formats cibles, de manière intuitive pour un développeur Java sans requérir une connaissance pointue de XML. Par exemple, JDOM utilise des classes plutôt que des interfaces. Ainsi pour créer un nouvel élément, il faut simplement instancier une classe.

Dans le rôle de manipulation sous forme d'arbre, JDOM offre une plus grande facilité pour répondre aux cas les plus classiques d'utilisation. Cette facilité d'utilisation de JDOM lui permet d'être une API dont l'utilisation est assez répandue. JDOM vérifie que les données contenues dans les éléments respectent la norme XML.

## 4.3 Plan de l'implémentation d'annotation sémantique de service web par l'ontologie construite

La Figure 4.1 présente l'architecture du prototype, ensuite nous détaillerons l'implémentation des principaux modules développés.

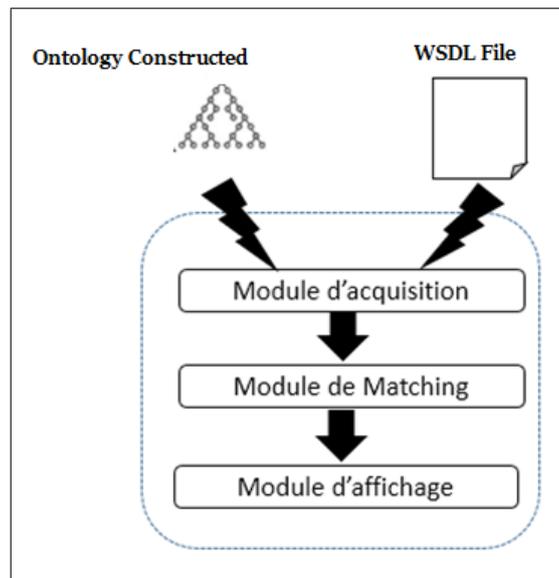


FIGURE 4.1 – Architecture globale de l' annotation sémantique.

comme illustré dans la figure précédente, l'outil est constitué de trois modules :

**Module d'acquisition :** Ce module permet le chargement du fichier WSDL choisi et l'ontologie de domaine. Des dossiers sont configurés par défaut mais l'utilisateur peut toujours sélectionner lui même son dossier.

**Module de matching :** Ce module permet de visualiser dans un tableau l'ontologie avec leurs mesures de similarité correspondantes (Les scores). Nous rappelons que nous avons utilisé l'algorithme de calcul de similarité de Wu et Palmer [53]. L'utilisateur peut décider donc sur le seuil à fixer pour le "matching" et par conséquent une annotation du WSDL effectuée selon le mécanisme préconisé par l'approche SAWSDL. Comme résultat, nous aurons un fichier WSDL annoté, un fichier enrichi sémantiquement qui peut servir pour les étapes suivantes.

**Module d'affichage :** permet d'afficher le document WSDL annoté.

## 4.4 Présentation de quelques fenêtres

Dans ce qui suit, nous allons présenter quelques interfaces du prototype réalisé avec explications pour bien illustrer son fonctionnement :

### 4.4.1 Fenêtre principale

L'interface principale est divisée en deux parties principales. La première partie à gauche est consacrée pour le chargement des fichiers d'ontologie (OWL) et la seconde, à droite, pour les fichiers de description des services Web (Fichiers WSDL).

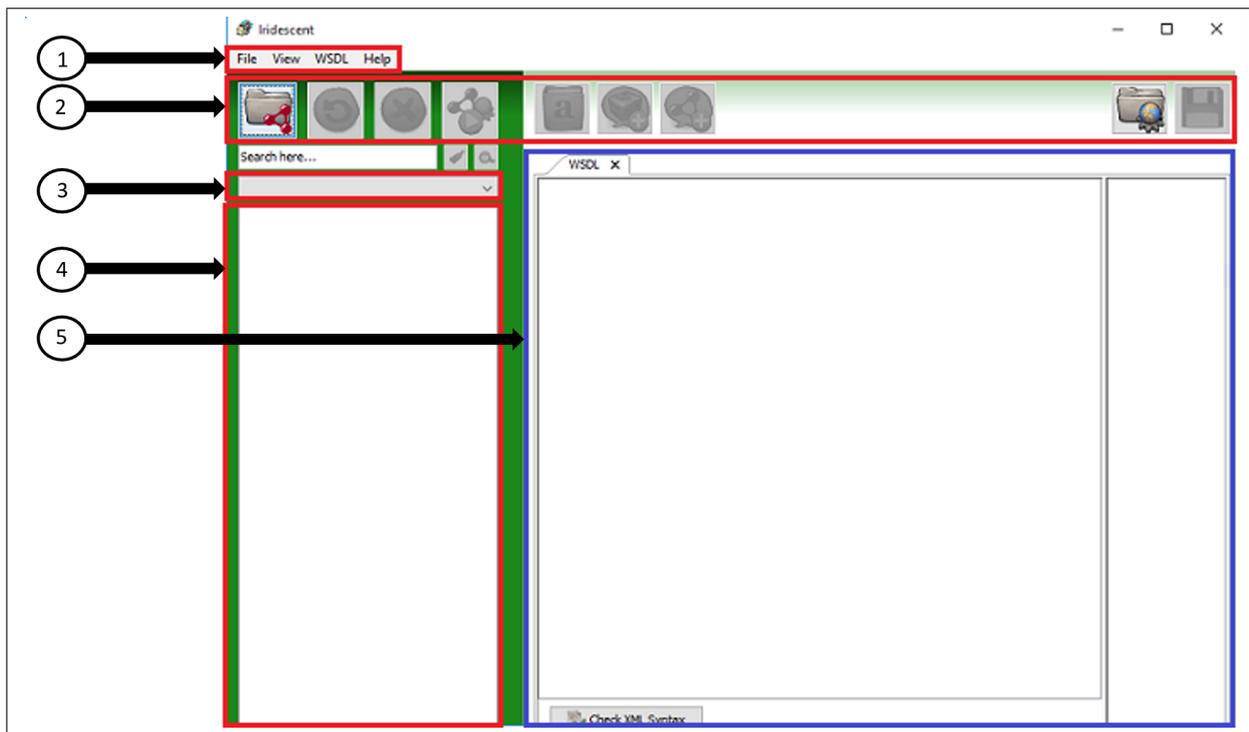


FIGURE 4.2 – Fenêtre principale de l'outil "SAWS".

1. Barre des menus.
2. barre d'outils.
3. l'ontologie chargée OWL.
4. La zone du contenu OWL.
5. La zone du contenu WSDL.

#### 4.4.2 Panneau d'ontologie

L'ontologie est située à gauche de la fenêtre principale. Elle fournit une barre d'outils pour les connexions liées à l'ontologie, et une zone qui héberge les ontologies vues dans une structure hiérarchique arborescente.

##### 4.4.2.1 Barre d'outils de gestion des fichiers d'ontologie (Ouvrir, Recharger, Fermer)

Les utilisateurs peuvent ouvrir et charger un fichier d'ontologie, recharger le fichier d'ontologie actuel ou le fermer, en utilisant les boutons respectifs de la barre d'outils de l'ontologie.



FIGURE 4.3 – Barre d'outils de gestion des fichiers d'ontologie (Ouvrir, Recharger, Fermer).

Le fichier d'ouverture d'ontologie peut également être effectué via le menu File-> Open -> Ontology. À l'ouverture, la boîte de dialogue "Open Ontology" apparaît. Dans le cas où un fichier local doit être chargé, l'option "From File System" doit être choisie et l'utilisateur doit parcourir le fichier. Au lieu de cela, l'option "From URL " permet d'entrer une URL où le fichier d'ontologie souhaité se trouve en ligne.

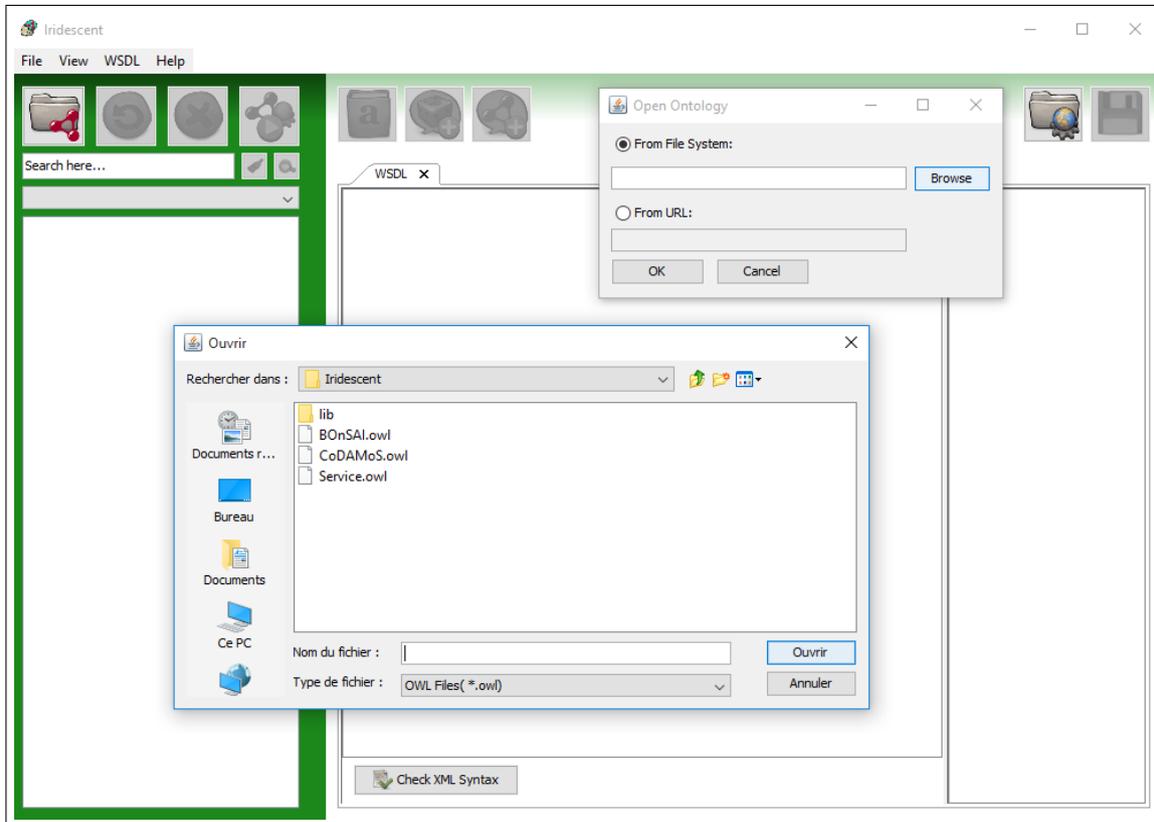


FIGURE 4.4 – Boîte de dialogue Open Ontology.

En appuyant sur OK, l'arbre de l'ontologie est chargée et les boutons de la barre d'outils de gestion de l'ontologie sont activés.

#### 4.4.2.2 Commutation entre les ontologies

Plusieurs ontologies peuvent être ouvertes en même temps en utilisant l'une des méthodes décrites ci-dessus. L'utilisateur peut alors basculer entre les fichiers ouverts en utilisant la liste déroulante.

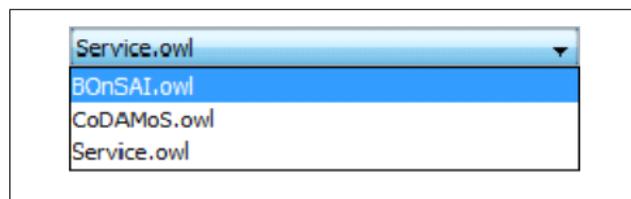


FIGURE 4.5 – Passage entre plusieurs ontologies à l'aide de la liste déroulante.

#### 4.4.2.3 Recherche de la classe d'ontologie

Une autre caractéristique fournie est la recherche de classe d'ontologie qui est enrichie d'une auto complétude. L'utilisateur peut commencer à taper le mot-clé pour la recherche et les complétions automatiques apparaissent ci-dessous. Traditionnellement, ils peuvent être sélectionnés à l'aide de la souris ou du clavier.

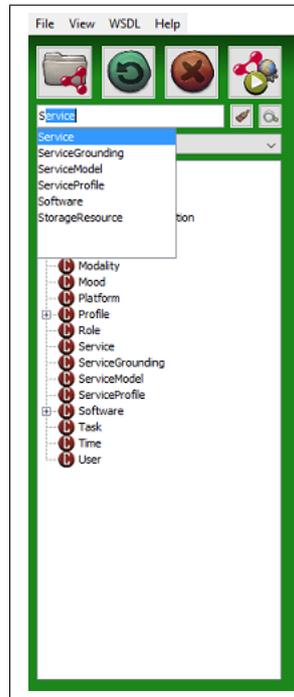


FIGURE 4.6 – Recherche d'une classe d'ontologie.

En appuyant sur Entrée ou sur le bouton de la loupe, l'utilisateur navigue sur la classe correspondante sur l'Arbre de l'ontologie.



FIGURE 4.7 – Sélection de la classe résultante.

Appuyez sur le bouton avec l'icône de la brosse pour effacer le champ de texte de recherche. Le bouton avec l'icône de la loupe, exécute la requête de recherche. Alternativement, cela peut se faire en appuyant sur le bouton Entrer.



FIGURE 4.8 – Barre de recherche.

### 4.4.3 Panneau de service

Le panneau de service héberge les fichiers de description du service Web. Il permet également de les éditer et de les enregistrer.

#### 4.4.3.1 Fichiers de description du service d'ouverture

Le bouton "Ouvrir WSDL / SAWSDL" sur le côté droit de la barre d'outils du service peut être poursuivi pour ouvrir un fichier de description de service. En outre, l'élément de menu Fichier -> Ouvrir WSDL / SAWSDL a le même effet. Des dossiers peuvent être chargés localement ou par URL. À cette fin, un cadre de dialogue apparaît, semblable à la boîte de dialogue "Open Ontology".



FIGURE 4.9 – Barre d'outils du panneau de service.

Les icônes ne sont activés que si l'ontologie "owl" et le fichier "wsdl" sont disponible(sont ouvertes).

Les éléments du service sont affichés à droite dans une hiérarchie arborescente. Le code est affiché à gauche pour l'élément sélectionné et tous ses enfants. Par conséquent, la sélection de la racine affiche le code entier.

Plusieurs fichiers WSDL peuvent être chargés, chacun sur un onglet différent.

Certains fichiers WSDL incluent l'élément Schéma dans un fichier .xsd séparé. Lors de l'ouverture d'un fichier WSDL, l'application vérifie si le fichier .xsd peut être trouvé. Dans le cas où il ne peut pas, une boîte de dialogue apparaît, pour informer l'utilisateur et lui permettre de localiser manuellement le fichier localement.

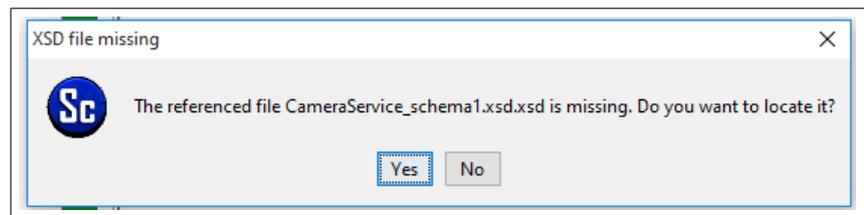


FIGURE 4.10 – Boîte de dialogue pour le fichier .xsd non trouvé.

Manuellement ou automatiquement, le fichier .xsd est ouvert sur un onglet séparé, tout comme tout fichier de description de service.

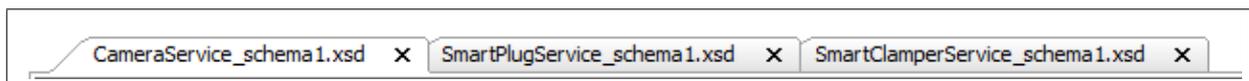


FIGURE 4.11 – Fichier xsd chargé sur un autre onglet.

#### 4.4.3.2 Vérification de syntaxe wsdl

En appuyant sur le bouton "Vérifier la syntaxe XML", vérifie la syntaxe XML du code et affiche un message

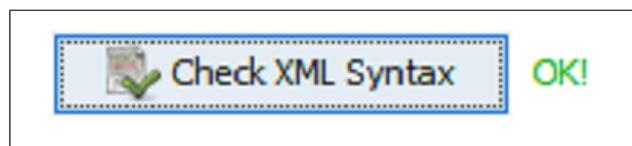


FIGURE 4.12 – Syntaxe XML correcte.

Une syntaxe XML incorrecte affiche un message en conséquence.



FIGURE 4.13 – Syntaxe XML erronée.

#### 4.4.3.3 Ouverture WSDL, OWL



FIGURE 4.14 – Boutons d'ouverture WSDL et OWL.

#### 4.4.4 SAWSDL Namespace

Les utilisateurs peuvent ajouter l'attribut `sawSDL : namespace` au noeud racine WSDL.



FIGURE 4.15 – L'ajout du bouton de l'espace de nom SAWSDL.

Cela peut se faire soit en appuyant sur le bouton "Add SAWSDL Namespace", soit en cliquant sur le bouton droit de noeud de la racine de l'arbre WSDL.

Une fois l'espace de noms ajouté, un nouveau noeud est créé en tant qu'enfant du noeud racine, représentant l'espace de noms SAWSDL.

Lorsque l'espace de noms est ajouté, l'icône du bouton " Add SAWSDL Namespace " change et en appuyant dessus, l'espace de noms existant se supprime automatiquement.

#### 4.4.5 modelReference et SchemaMapping

SAWSDL fournit des mécanismes pour référencer des concepts de modèles définis à l'extérieur du document WSDL. Cela se fait grâce à l'attribut `"sawSDL`.

Le mécanisme d'annotation utilisé est automatique et il représente celui proposé par l'approche SAWSDL. SAWSDL définit comment on peut ajouter des annotations sémantiques dans les différentes parties d'un document WSDL comme les entrées, les sorties, les structures des messages, les interfaces et les opérations. L'ajout est réalisé en mettant des attributs `"modelReference"`, `"liftingSchemaMapping"` ou bien `"loweringSchemaMapping"`.

##### 4.4.5.1 Ajout de modèles de références

`"modelReference"` permet d'associer un composant WSDL ou XML Schema à un concept d'une ontologie.

Pour ajouter un attribut `sawSDL : modelReference` à un élément de fichier WSDL il existe trois façons.

1. Faire glisser une classe d'ontologie, et en la déposant sur le noeud d'arbre WSDL souhaiter.
2. Ajouter un bouton de référence de modèle, ceci ouvre une boîte de dialogue où la classe d'ontologie, et le noeud WSDL de destination doit être sélectionnés.
3. Cliquez avec le bouton droit de la souris sur le noeud WSDL, où l'annotation doit être ajoutée et sélectionnez "Ajouter une référence de modèle". Une boîte de dialogue apparaît,

sur laquelle l'arborescence de l'ontologie active est présentée. L'utilisateur choisit la classe d'ontologie souhaitée pour l'annotation.

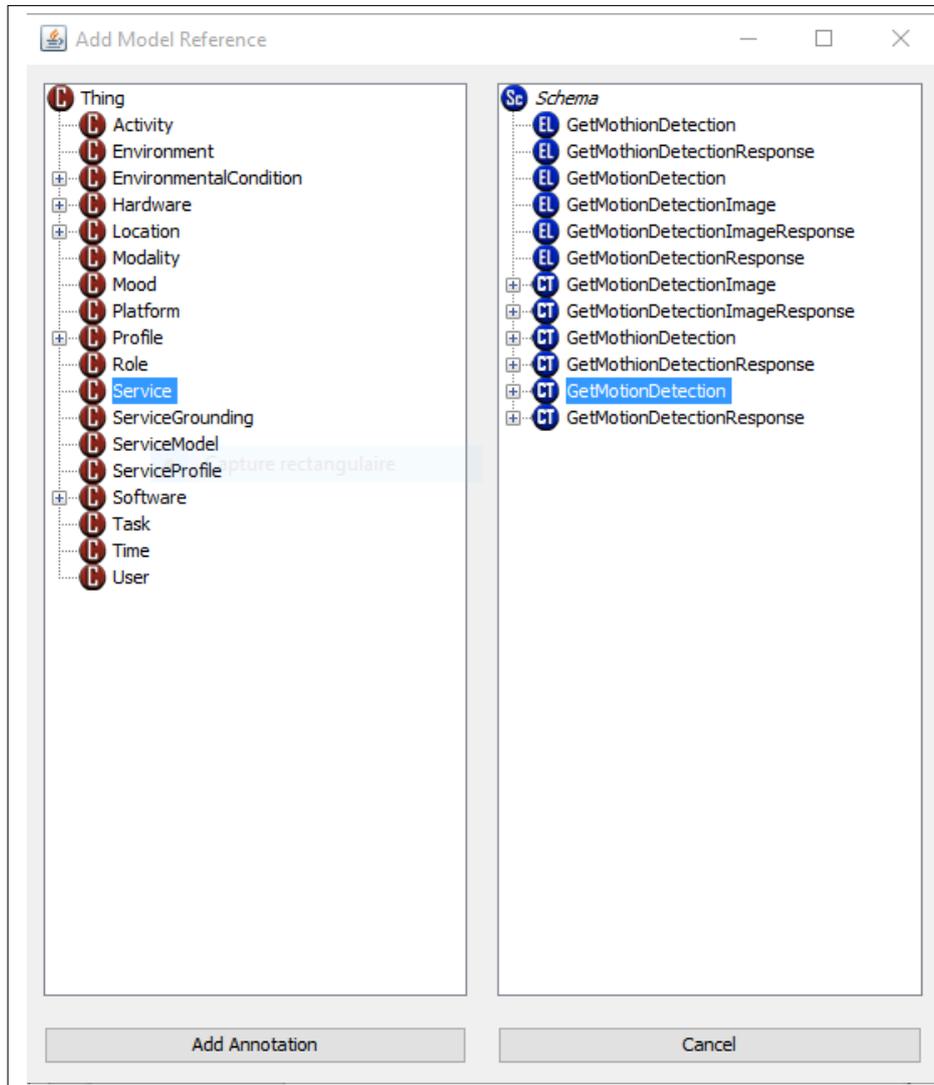


FIGURE 4.16 – Boite de dialogue d'ajout d'un modèle de référence.

Le résultat est l'ajout de l'attribut de référence du modèle à l'élément désiré et la création du noeud correspondant comme enfant du noeud de l'arbre WSDL.

#### 4.4.5.2 Ajout de mappage de schéma

"liftingSchemaMapping" et "loweringSchemaMapping" permettent de spécifier la correspondance (dit Mapping) entre les données sémantiques et les éléments XML. Les "SchemaMapping" sont utilisés pour établir la correspondance entre les structures des entrées et des sorties.

L'ajout d'attributs de mappage de schéma peut se faire de manière similaire. Les utilisateurs peuvent soit cliquer avec le bouton droit de la souris sur le noeud sélectionné, soit appuyer sur le bouton "Add Lifting/lowering Schema Mapping".

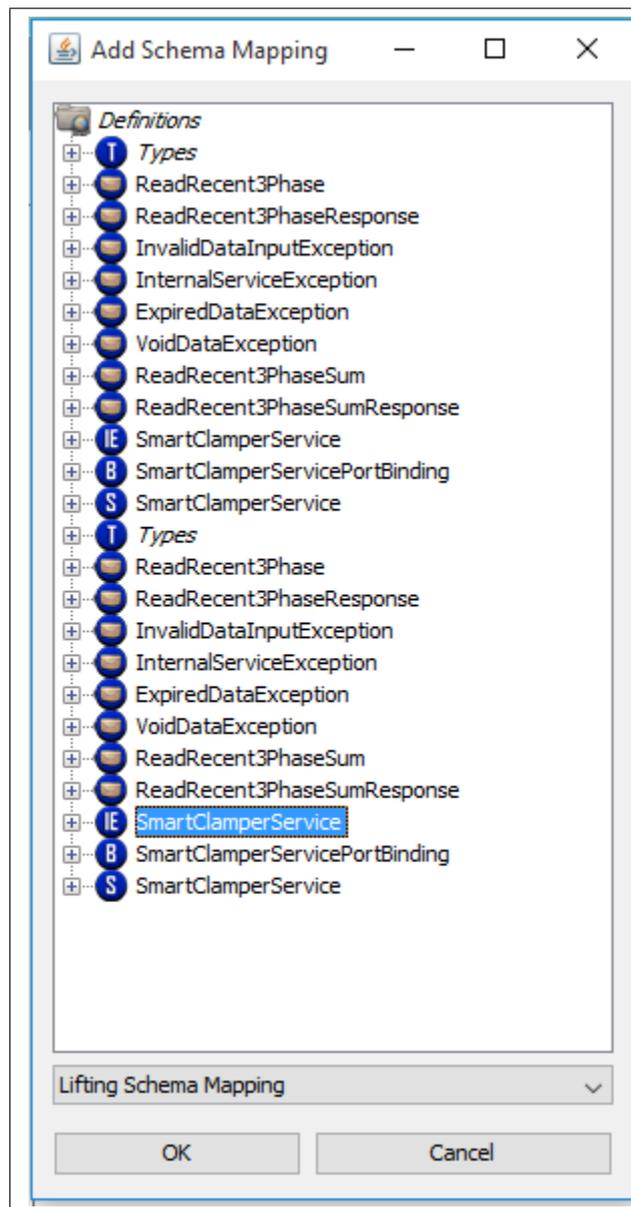


FIGURE 4.17 – la boîte de dialogue d’ajout du mappage de schéma.

Sur la boîte de dialogue affichée, les utilisateurs doivent sélectionner le nœud où l’attribut de mappage de schéma qui va être ajouté, puis sélectionnez son type ( `liftingSchemaMapping` ou `loweringSchemaMapping` ). En cliquant sur OK, une boîte de dialogue invite l’utilisateur à URL souhaitée de la transformation de mappage de schéma (généralement un fichier XSLT ou SPARQL).

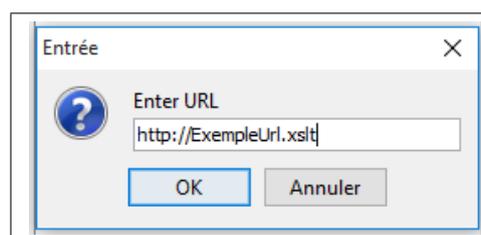


FIGURE 4.18 – Ajout de l’URL du mappage de schéma.

Par conséquent, l'attribut de mappage de schéma est ajouté au composant WSDL et le nœud correspondant est créé.

#### 4.4.5.3 Suppression des éléments SAWSDL

La suppression de l'espace de noms SAWSDL, des références de modèle ou des mappages de levage et de réduction de schéma peut se faire en cliquant avec le bouton droit de la souris sur les nœuds d'arbre correspondants et en sélectionnant "Supprimer"

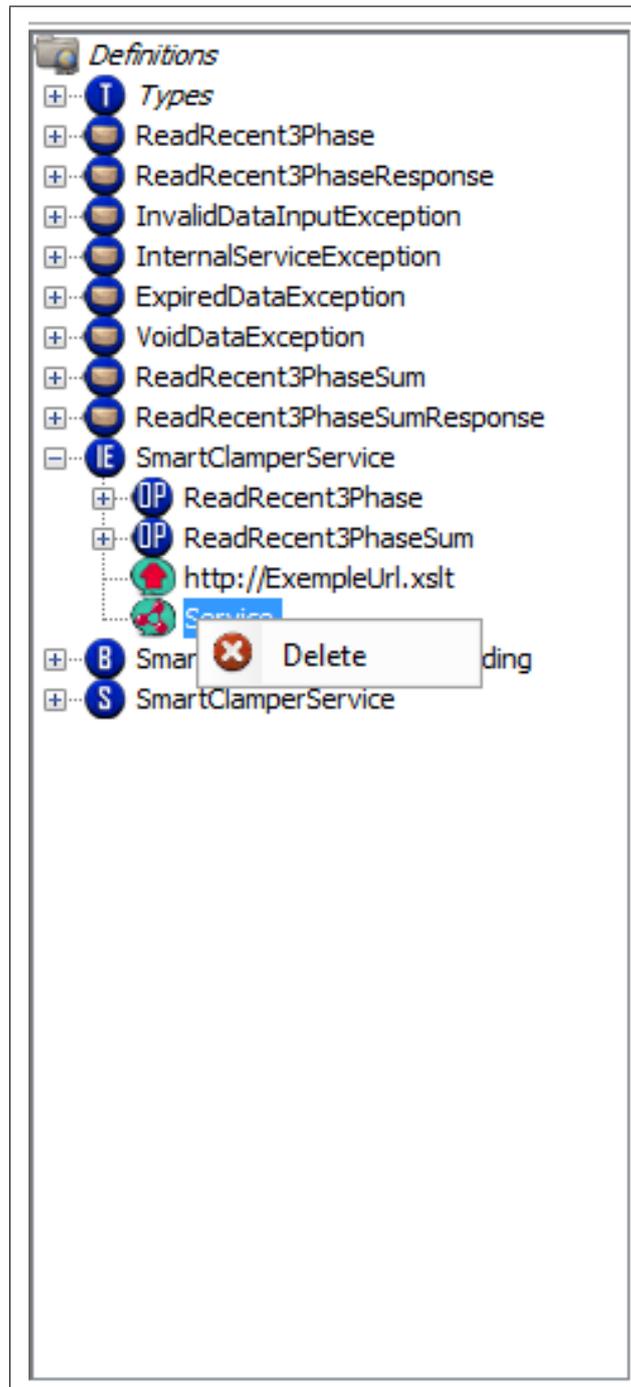


FIGURE 4.19 – Suppression d'un élément SAWSDL .

#### 4.4.6 Le processus de "matching"

En appuyant sur le bouton "Matching", l'outil recherche d'annotation recommandée qui pourrait être effectuée, en fonction de la similitude entre les noms de classes d'ontologie et les noms de composants de WSDL. Le bouton est activé si une ontologie et une description de service sont ouvertes.



FIGURE 4.20 – Button de matching.

À partir d'ontologie "owl" (l'ontologie biomédicale construite à partir des données ouvertes liées) et un fichier "wsdl", le processus de "matching" permet d'associer à chaque élément WSDL le concept correspondant de l'ontologie de domaine.

On définit la fonction "matching" qui associe un élément WSDL avec le concept le plus proche dans l'ontologie de domaine. La fonction match est définie comme suit :

$$Match : E_{wsdl} \rightarrow E_{ontologie}$$

$$e \rightarrow c, \text{telque}(\text{sim}(e, c) = \text{MAX}_{i=1}^n \text{sim}(e, c_i) \text{et} (\text{sim}(e, c_i) \geq \text{seuil}))$$

Avec  $E_{wsdl}$  l'ensemble des éléments du document WSDL,  $E_{ontologie}$  l'ensemble de concepts de l'ontologie de domaine,  $e$  est un élément de  $E_{wsdl}$ ,  $c$  est un concept de  $E_{ontologie}$ ,  $n$  le nombre de concept de l'ontologie de domaine et  $\text{sim}$  une mesure de similarité.

Dans la boîte de dialogue suivante, toutes les recommandations sont affichées. Pour chacun d'entre eux, l'élément de service et classe d'ontologie correspondante est affichée. Les utilisateurs peuvent sélectionner les recommandations souhaitées pour s'engager dans la description du service, ou même utiliser les boutons "Sélectionner tout" et "Supprimer tout".

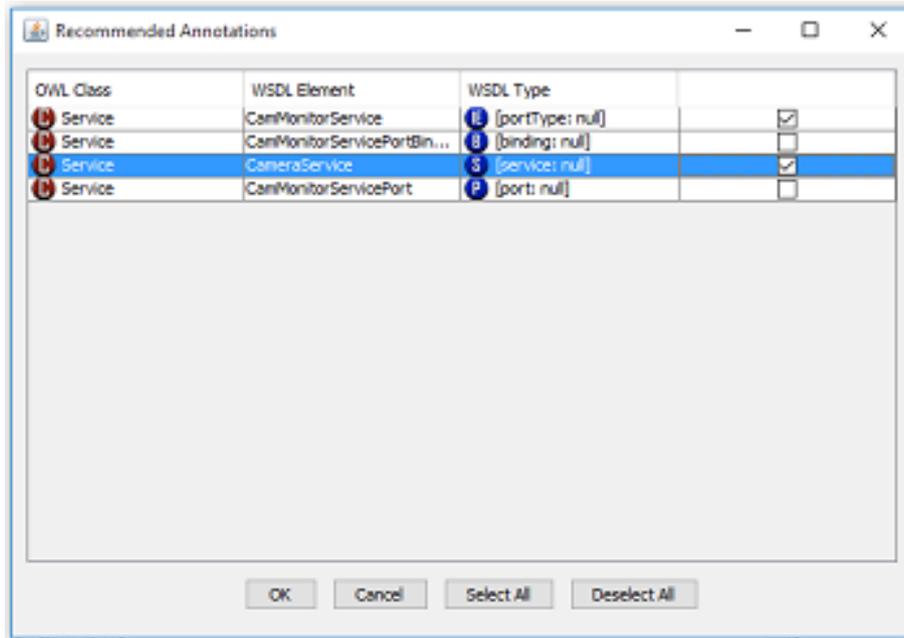


FIGURE 4.21 – Boite de Dialogue de recommandation.

Toutes les sélections sont engagées dans la description du fichier de service.

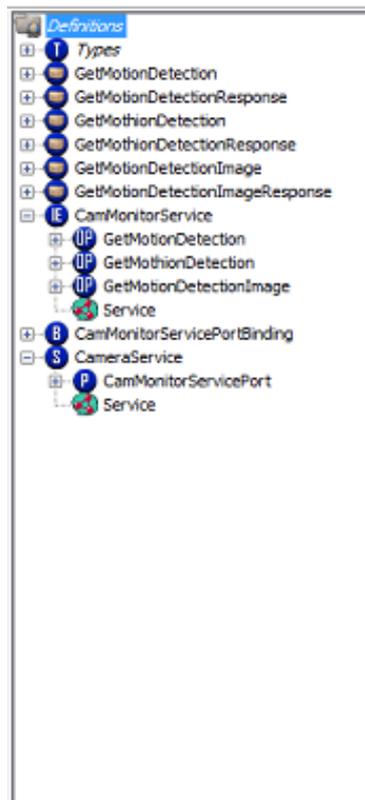


FIGURE 4.22 – Réussite des annotations recommandées ajoutées

Une fois l'étape d'annotation du fichier de description WSDL avec l'ontologie choisie est effectuée avec succès. Le résultat de cette étape est un fichier WSDL annoté que l'utilisateur peut stocker

dans un dossier par défaut.

L'enregistrement du fichier peut se faire en appuyant sur le bouton "Enregistrer", ou en sélectionnant Fichier -> Enregistrer dans le menu.

## 4.5 Conclusion

Dans ce dernier chapitre, nous avons présenté une implémentation et une expérimentation du prototype SAWS qui a pour but l'annotation sémantique des services web à partir des données ouvertes liées. Nous avons également expérimenté notre prototype sur un cas réel, celui du Service Web biomédicale.

L'expérimentation a révélé que l'approche est plus flexible au niveau de l'annotation sémantique mais en revanche elle a révélé aussi le fait qu'il peut y avoir des problèmes de complétudes dus aux insuffisances des informations obtenues à partir du fichier WSDL, de plus, nous devons noter que certaines fonctionnalités du prototype sont actuellement implémentées de façon relativement simplifiée et en conséquence nous demeurons convaincus, compte tenu de l'intérêt de cette approche, que ce prototype doit être soutenu et poursuivi afin de lui apporter toutes les améliorations nécessaires.

## *Conclusion générale et perspectives*

Le présent travail a été mené dans le but de proposer une nouvelle approche d'annotation sémantique des entrées et sorties des services web basée sur les données liées ouvertes (Linked Open Data) telles que DBpedia. Ces dernières sont utilisées pour la construction d'une ontologie biomédicale.

Pour exprimer le cadre de notre étude et ces frontières ; en premier lieu, nous avons montré l'étape de construction d'une ontologie biomédicale à partir des données ouvertes liées, et une autre étape d'annotation sémantique des services Web par l'ontologie construite.

Pour l'étape de construction d'ontologie, nous avons utilisé l'algorithme de calcul de similarité de Wup qui met à la disposition des utilisateurs, une ontologie qui répond au besoin du domaine biomédical, le processus de création de l'ontologie est subséquent "mapping" avec un certain nombre des sources de données disponibles sur "LOD".

Concernant l'étape d'annotation sémantique des services Web nous avons utilisé un "module de matching" qui est inclus dans le processus d'annotation, il permet d'associer à chaque élément WSDL le concept correspondant d'ontologie. Le résultat de cette étape est un fichier WSDL annoté.

Ce projet a fait l'objet d'une expérience intéressante qui nous a permis d'améliorer nos connaissances et nos compétences dans le domaine de la programmation, nous avons appris à mieux manipuler le langage java, il nous a été utile dans la mise en œuvre de notre application. Ainsi, la réalisation de ce projet est une opportunité de découvrir la technologie des services web et son importance sur le marché et aussi une occasion de nous familiariser avec l'environnement du travail et de la vie professionnelle.

Au terme de ce travail qui peut être vu comme un début et non comme une fin, nous pouvons proposer un nombre de perspectives. Ces dernières peuvent venir compléter, améliorer, voir étendre ce modeste travail. Parmi ces perspectives, on peut citer :

- ✓ Une première perspective de nos travaux est le développement d'outils permettant l'annotation sémantique des Services Web car nous avons remarqué une absence des Services Web

sur le Net au domaine biomédical et il n'existe pas à nos jours des outils automatiques ou semi automatiques pour permettre l'annotation sémantique des éléments de service web.

- ✓ Une deuxième perspective est de proposer des outils automatiques ou semi-automatiques pour la création d'ontologie de domaine à partir des données liées ouvertes , un projet de recherche qui reste toujours un cas d'étude.

## *Bibliographie*

- [1] RABAHALLAH K. Structuration des données et des services pour le e-learning, Mémoire de Magister en Informatique Option : Ingénierie des Systèmes Informatiques, Université Mouloud Mammeri de Tizi-Ouzou, Tizi-Ouzou, 06/06/2016.
- [2] <https://openclassrooms.com/courses/les-services-web>, dernier accès : 06/05/2017.
- [3] LABLANCHE C. SEINE F et Gastaud S. Les Web Services, Mémoire licence en Informatique, Université de Nice-Sophia Antipolis, Nice, 2005.
- [4] MEKHZOUMI D. Reformulation de requêtes pour la découverte des services web sémantiques, mémoire magistère en Informatique option Réseaux et Systèmes Distribués, université Abderrahmane Mira de Bejaia, 2009.
- [5] BRIAN S. SOAP Web Services, Master of Science Computer Science School of Informatics University of Edinburgh 2003.
- [6] BAGET J-F. CANAUD E. EUROME J et HACID MS. Les langages du Web sémantique, Université Claude Bernard Lyon 1, 20 novembre 2013.
- [7] DUMAS M et FAUVET M.C. Intergiciel et construction d'applications réparties, mémoire de licence en Informatique, Université Paris, 12/06/2008.
- [8] MORA O. Conception et développement d'un service pour la recherche d'objets immobiliers, Bachelor en Informatique de gestion, Université de Fribourg, Suisse, Septembre 2007.
- [9] BEKKOUCHE A. Composition des Services Web Sémantiques À base d'Algorithmes Génétiques, Magistère en Informatique, Université Aboubekr Belkaid Tlemcen , Tlemcen , 2012.
- [10] HACHEMI Fakhr-eddine. Description sémantique des objets d'apprentissage à base de modèles de contenu, Mémoire en Magister, Option : S.I.C (Systèmes d'Information et de Connaissance), Université Abou Bekr Belkaid – Tlemcen, 2009.
- [11] Gueffaz M . Model Checking et Web Sémantique, these en informatique, Université de Bourgogne, 11 Décembre 2012.
- [12] BENLOUFFOK F. HAMANA B. Conception et réalisation d'une ontologie médicale, mémoire de master en Informatique option Administration et Sécurité des Réseaux, Université A/Mira de Bejaia, 2014.
- [13] EDDINE JRIDI J. L'ingénierie de documents d'affaires dans le cadre du web sémantique, Thèse doctorat en Informatique, Université de Montréal, juin 2014.

- [14] TUAN ANH Ta. Web sémantique et réseaux sociaux-construction d'une mémoire collective par recommandation mutuelle et (re-)présentations, thèse en informatique et réseaux, Paris, juillet 2005.
- [15] BOUARROUDJ S. Raisonnement sur une ontologie enrichie par des règles SWRL pour la recherche sémantique d'images annotées, Mémoire de magister en Informatique option : Intelligence Artificielle, Ecole doctorale en Informatique de l'Est –Pole ANNABA, Université 20 Aout 1955, Skikda, 2010.
- [16] KHELIF Mohamed Khaled. Web sémantique et mémoire d'expériences pour l'analyse du transcriptome, Thèse doctorat en Informatique, UNIVERSITE DE NICE-SOPHIA ANTI-POLIS - UFR Sciences, Ecole Doctorale Sciences et Technologies de l'Information et de la Communication (S.T.I.C), 4 avril 2006.
- [17] MOUSSAOUI K et FEREDJ D.E. Conception et développement d'un Outil de recherche sur le web à base d'agent, Mémoire master en Informatique Fondamentale, UNIVERSITE KASDI MERBAH OUARGLA, Ouargla, 01 / 07 / 2013.
- [18] PHUC HIEP LUONG. Gestion de l'évolution sémantique d'entreprise. Thèse doctorat en Informatique Temps Réel, Robotique, Automatique. Ecole des Mines de Paris, 14 décembre 2007.
- [19] MARTIN D. BURSTEIN Lu M et HOBBS J. Owl-s : Semantic markup for Web services, rapport technique, W3C, [http ://www.w3.org/Submission/OWL-S/](http://www.w3.org/Submission/OWL-S/), 2004.
- [20] MICHA M. Médiation Sémantique Orientée Contexte pour la Composition de Services Web. Thèse en Informatique et Information pour la Société, Université Claude Bernard Lyon I, 15 Novembre 2007 .
- [21] DEHANE Aïcha Djihad et KEBIR Zohra. Evaluation des techniques de codage d'ontologies sur les performances de la composition de services Web, Mémoire Master Option : Modèle Intelligent et Décision (M.I.D), Université Abou Bakr Belkaid– Tlemcen, 2012.
- [22] HEMAM née HIOUAL O. Composition sémantique des services web dans un contexte d'ebXML, Thèse doctorat en Informatique, Université Mentouri de Constantine, 09/10/2011.
- [23] GHARZOULI Mohamed. Composition des Web Services Sémantiques dans les systèmes Peer-to-Peer, Thèse Pour obtenir le diplôme de Doctorat en sciences en Informatique, Université Mentouri Constantine, 25/09/2011.
- [24] BERNERS-LEE, Tim, HENDLER, James et LASSILA. The semantic web. Scientific American. mai 2001. pp. 29-37.
- [25] PRONGUE N. Modélisation et transformation des métadonnées de RERO et Linked Open Data, Mémoire de master en Informatique documentaire, Haute école de gestion de Genève (HEG-GE), Genève, 29 aout 2014.
- [26] ATANASSOVA I. Exploitation informatique des annotations sémantiques automatiques d'Excom pour la recherche d'informations et la navigation, Thèse doctorat en Informatique-Linguistique, UNIVERSITÉ PARIS-SORBONNE, 14 janvier 2012.

- [27] BERNERS-LEE et Tim. Linked Data. World Wide Web Consortium. 2010. [Consulté le 13 avril 2017]. Disponible à l'adresse <http://www.w3.org/DesignIssues/LinkedData.html>
- [28] Dumas M et Fauvet M.C. Intergiciel et construction d'applications réparties, mémoire de licence en Informatique, 12/06/2008.
- [29] PRIE Y et GARLATTI S. Méta-données et annotations dans le Web sémantique, in Le Web sémantique, CHARLET J. L AUBLET P et R EYNAUD C. (Ed.), Hors série de la Revue Information - Interaction - Intelligence (I3), 4(1), Cépaduès, Toulouse, pp. 45-68. 2004.
- [30] DESMONTILS E et JACQUINacquin C. Indexing a Web Site with a Terminology Oriented Ontology. In The Emerging Semantic Web Amsterdam : IOS Press,pp.181-197. 2002.
- [31] GUERAICH Epouse BENMAZA S. Une approche basée annotation sémantique de documents pour la gestion d'une mémoire d'entreprise, Doctorat en Sciences Spécialité Informatique, Université Mentouri Constantine, 28 / 02 / 2012.
- [32] AMARDEIH Florence. Web Sémantique et Informatique Linguistique :propositions méthodologiques et réalisation d'une plateforme logicielle, Thèse doctorat en Informatique, Université Paris X - NANTERRE, Mai 2007.
- [33] HEPP M. Semantic Web and semantic Web services : father and son or indivisible twins ", IEEE Internet Computing, vol. 10, no. 2, pp. 85-88. 2006.
- [34] CHIFU V.R. SALMONIE I et CHIFU E.S. Taxonomy Learning for Semantic Annotation of Web Services, Proceedings of the 11th WSEAS International Conference on Computers, ACM, pp. 300-305. 2007.
- [35] HEß A and KUCHMMERICK N. Learning to Attach Semantic Metadata to Web Services, The Semantic Web, Springer Berlin, Heidelberg, pp. 258-273. 2003.
- [36] LERMAN K. PLANGRASOCHOK A and KNOBLOCK C.A. Automatically labelling the inputs and outputs of web services, In Proceedings of the National Conference on Artificial Intelligence AAAI, Charlotte, USA, pp. 1363-1369. 2006.
- [37] EMIL Ş. Chifu and Ioan Alfred Letia. Unsupervised Semantic Annotation of Web Service Datatypes, 2010.
- [38] YUAN-JIE LI. JIAN CAO. Web Service Classification Based on Automatic Semantic Annotation and Ensemble Learning, 2012.
- [39] BELHAJJAME K. EMBURY S.M. PATON N.W. Stevens R and Goble C. "Automatic annotation of Web Services Based on Workflow Definitions", ACM Transactions on the Web, vol. 2, no.2, pp. 1-34. 2008.
- [40] LIPING Z. GUANGYAO L. YONGQUAN L and JING S. Design of ontology mapping framework and improvement of similarity computation, Journal of Systems Engineering and Electronics, vol. 18, no. 3, pp. 641-645. 2007.
- [41] ABHIJIT A. PATIL. SWAPNA A. OUNDHAKAR . AMIT P. SHETH and KUNAL V. Meteor-s web service annotation framework. In WWW, pages 553–562, 2004.

- [42] SALOMIE I. CHIFU V.R and GIURGIU I. CUIUBUSAWS M. A Tool for Semantic Annotation of Web Services, 2008.
- [43] SONG T.X. TIAN P.J. LIU Y.H and HUANG B. Web Services Semantic Annotation and Auto-matching Based on SAWSDL, 2008.
- [44] ZHANG Z. CHEN S and ZHIYONG F. Semantic Annotation for Web Services Based on DBpedia, 2013.
- [45] GOSAL G. KOCHUTt K.J. KANNAN N and PROKINO. A Framework for Protein Kinase Ontology, in International Conference on Bioinformatics and Biomedicine(BIBM) 2011, IEEE : Atlanta, GA. p. 550 - 555.
- [46] SMITH et al. The OBO Foundry : coordinated evolution of ontologies to support biomedical data integration. NATURE BIOTECHNOLOGY, p. 1251-1255. 2007.
- [47] NOY N.F et al. BioPortal : ontologies and integrated data resources at the click of a mouse. Nucleic acids research, p.W170-W173. 2009.
- [48] GOH K.I et al. The human disease network. Proceedings of the National Academy of Sciences, p. 8685-8690. 2007.
- [49] HASSANZADEH O et al. A linked data space for clinical trials. arXiv preprint arXiv :0908.0567, 2009.
- [50] WISHART D.S et al. DRUGBANKrug. A comprehensive resource for in silico drug discovery and exploration. Nucleic acids research, p.D668-D672. 2006.
- [51] MENG L. HUANG R. and GU J.A .Review of Semantic Similarity Measures in WordNet. International Journal of Hybrid Information Technology. Vol.6, No. 2013.
- [52] RADA R. MILI H. BICHNELL E and BLETTNER M. Development and application of a metric on semantic nets. IEEE Transaction on Systems, Man, and Cybernetics : pp 17-30. 1989.
- [53] WU Z and PALMER M. Verb semantics and lexical selection. In proceedings of 32nd annual Meeting of the Association for Computational Linguistics, Las Cruces, New Mexico, June 1994.
- [54] LEACOCK C et CHODROW M. Combining Local Context and WordNet Similarity for Word Sense Identification. In WordNet : An Electronic Lexical Database, C. Fellbaum, MIT Press, 1998.
- [55] RESNIK P. Using information content to evaluate semantic similarity in taxonomy. In Proceedings of 14th International Joint Conference on Artificial Intelligence, Montreal, 1995.
- [56] LIN D. An Information-Theoretic Definition of similarity. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML'98). Morgan Kaufmann : Madison, WI, 1998.
- [57] JIANG J.J and CONRATH D.W. Semantic similarity based on corpus statistics and lexical taxonomy, Proceedings of International Conference on Research in Computational Linguistics, (1997) August 22-24, Taipei, TaiWan.

- [58] HIRST G et STONGE D. Lexical chains as representations of context for the detection and correction of malapropisms. In Christiane Fellbaum (editor), WordNet, An electronic lexical database, Cambridge, MA : The MIT Press .1998.
- [59] PATWARDHAN S. BANERJEE S and PEDERS T. Using measures of semantic relatedness for word sense disambiguation, Proceedings of 4th International Conference on Computational Linguistics and Intelligent Text Processing and Computational Linguistics, Mexico City, Mexico. 2003.
- [60] WU Z and PALMER M. Verb semantics and lexical selection. In proceedings of 32nd annual Meeting of the Association for Computational Linguistics. Las Cruces, New Mexico, June 1994.
- [61] MENG L. HUANG R and GU J. A Review of Semantic Similarity Measures in WordNet. International Journal of Hybrid Information Technology. Vol.6, 2013.
- [62] [http ://ipeti.forumpro.fr/t21-definition-de-langage-java-java-script](http://ipeti.forumpro.fr/t21-definition-de-langage-java-java-script), , dernier accès : 06/05/2017.
- [63] [http ://www.techno-science.net/?onglet=glossairedefinition=5346](http://www.techno-science.net/?onglet=glossairedefinition=5346), dernier accès : 06/05/2017.
- [64] HACINE GHERBI Ahcine. Construction d'une Ontologie pour le WEB Sémantique, Mémoire Magister Option : Ingénierie des Systèmes Informatiques, UNIVERSITÉ FERHAT ABBES –SÉTIF1, 2014.
- [65] [http ://jmdoudoux.developpez.com/cours/developpons/java/chap-jdom.php](http://jmdoudoux.developpez.com/cours/developpons/java/chap-jdom.php), dernier accès : 06/05/2017.

# Résumé

Avec la croissance du nombre de services publiés sur Internet, il est difficile pour les demandeurs de service de découvrir les services Web correspondants à leurs besoins. La raison de ce phénomène est que les services classiques ne disposent pas d'information sémantique pour pouvoir les découvrir ou sélectionner les services Web les plus appropriés.

Cependant, afin de réaliser les tâches automatiques telles que la découverte et la composition des services Web, il est indispensable de doter ces services par une information sémantique.

Le présent mémoire présente une nouvelle approche d'annotation sémantique des services Web existants. L'approche est supportée par un outil logiciel qui permet, à partir d'une ontologie de domaine et un fichier de description des services WSDL, l'annotation d'un service web existant selon le principe d'annotation proposée par le standard SAWSDL.

Ainsi, l'ontologie de domaine utilisée pour l'annotation a été créée en s'appuyant sur les instances disponibles dans les données ouvertes liées (Linked Open Data).

**Mots clés :** Annotation sémantique, Services Web, Données Ouvertes Liées, WSDL, SAWSDL, Ontologie.

# Abstract

With the growth in the number of services published on the Internet, it is difficult for service's providers to discover the Web services corresponding to their needs. The main reason for this phenomenon is that the conventional services do not have semantic information to discover them or to select the most appropriate web services.

However, in order to perform automatic tasks such as the discovery and the composition of Web services, it is essential to provide these services with a semantic information.

This project presents a new approach to semantic annotation of existing Web services. The approach is supported by a software tool that allows the annotation of an existing web service, and this is according to the annotation principle proposed by the SAWSDL standard, from an ontology of domain and a description file for WSDL services.

Thus, the domain ontology used for the annotation was created basing on the instances available in Linked Open Data.

**Keywords :** Semantic annotation, Web services, Linked Open Data, WSDL, SAWSDL, Ontology.