



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THEME :

Accélérer la découverte de services web
Sémantiques

Etudiant(e) : « **Khedim Asmaa** »

Encadrant(e) : « **Bedad Fatima** »

Année Universitaire 2015/2016

Résumé

Les technologies courantes permettent l'utilisation des services web mais seulement au niveau syntaxique, donc, ils fournissent seulement un ensemble de services rigides qui ne peuvent pas s'adapter à un environnement en cours d'évolution sans l'intervention des utilisateurs humains. Une solution possible à tous ces problèmes est susceptible d'être fournie en convertissant les services web en services web sémantiques, qui sont sémantiquement des applications annotées et qui peuvent être invoquées, découverts, peuvent se composer pour fournir d'autres fonctionnalités et s'exécuter à travers le web d'une manière automatique ou semi-automatique.

Plusieurs solutions ont été proposées pour les spécifications des services web sémantiques tels que les OWL-S et WSMO. Cependant, ces technologies 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 interpréter la signification des entrées, des sorties et des contraintes applicables aussi bien que le contexte dans lequel les services peuvent être employés.

Le présent mémoire présente un axe d'activités qui rentrent dans le cadre de services web sémantiques et propose une approche à base des ontologies et une mesure de similarité pour accélérer la découverte des applications SWS WSMO [1]. L'approche consiste en deux grandes phases : Tout d'abord, l'extraction d'informations pertinentes à partir du code source du web service du SWS WSMO, Deuxième phase est l'analyse de l'information extraite à l'aide de l'ontologie du domaine et la mesure de similarité.

Un ensemble d'outils, supportant l'approche proposée. Cela nous a permis d'évaluer et de valider l'approche sur un cas réel.

Mots clefs : Service Web sémantique, web service, WSMO, ontologie du domaine, mesure de similarité.

Table des matières

Introduction général	9
----------------------------	---

Chapitre I : l'évolution du web

I.1 Introduction :	12
I.2. L'histoire du web ::	12
I.3. Le web:	13
I.4. Services web.....	13
I.4.1. Architecture SOA.....	13
I.4.2. L'infrastructure des services web :	14
I.4.3 Les caractéristiques d'un service Web ::	15
I.5. Le web sémantique :	16
I.5.1. Les Ontologies :	16
I.5.1.1. Rôle des ontologies au sein du web sémantique :	16
I.5.1.3. Types d'ontologies :	17
I.6. Les services Web sémantique	18
I.6.1. L'infrastructure des services Web sémantiques:.....	18
I.6.2. Approches proposées pour la réalisation des services Web sémantiques.....	19
I.6.3. Comparaison et discussion.....	20
I.7. Les éléments de WSM.....	22
I.7.1. Les buts::.....	23
I.7.2. Ontologies :	23
I.7.3. Services Web:.....	23
I.7.4. Médiateurs :	23
I.7.5. Le Langage:.....	23
I.7.5.1. Syntaxe conceptuelle	24
I.7.5.2. Syntaxe des expressions logiques.....	24
I.8. Conclusion:.....	25

Chapitre 2 : L'approche proposée

II.1. Introduction:.....	27
II.2. Notre Approche:.....	27

II.3. Algorithme de l'approche proposée :.....	28
II.3.1 Phase extraction des informations pertinentes:.....	29
II.3.2. Phase d'analyse	30
II.3.2.1 Identification des concepts d'ontologie:.....	30
II.3.2.2 la mesure de similarité:.....	32
II.4. Les techniques de la mesure de similarité:.....	32
II.4.1. Les mesures de similarité basées sur la longueur du chemin entre les concepts :...	32
II.4.2. Les mesures basées sur le contenu d'information:.....	34
II.4.3. Les mesures basées sur le type des relations entre les concepts	34
II.4.4. La mesure de Wu-Palmer :.....	35
II.5. WordNet:.....	36
II.5.1 Structuration:.....	37
II.5.1.1. Notion de synset :.....	37
II.5.2. les relations dans WordNet:.....	38
II.6. Conclusion:.....	39
Chapitre III : Implémentation	
III.1. Introduction :.....	41
III.2. L'outil de développement:.....	41
III.3. Les résultats de notre approche	41
III.3.1. Exemple d'illustration:.....	41
III.3.2. Résultat de la première phase:.....	43
III.3.3. Résultat de la deuxième phase :.....	43
III.4. Capture d'écran de l'application :	47
III.5. Conclusion	48
Conclusion général.....	49
Bibliographie.....	50

Liste des figures :

Figure I.1 : les éléments de base dans une architecture SOA	13
Figure I.2 : l'infrastructure des services web.....	14
Figure I.3 : Exemple graphique d'une ontologie.....	17
Figure I.4 : Infrastructure des services Web sémantiques.....	19
Figure I.5 : Les éléments de base d'OWL-S.	20
Figure I.6 : Les éléments qui composent le WSMO	22
Figure I.7 : Un exemple d'un Web Service WSML	24
Figure II.1 : Processus d'accélérer la découverte SWS WSMO.....	28
Figure II.2: L'algorithme proposé	29
Figure I.3 : Exemple graphique d'une ontologie.....	30
Figure II.4: une partie du programme OWLModel.....	31
Figure II.5:Un fragment duWordNet, à partir duquel on déduit que doctor est plus similaire à nurse.....	33
Figure II.6 : Les relations conceptuelles	36
FigureII.7 : Ensemble des différents sens du mot child dans WordNet.....	37
Figure II.8 :Une illustration des nœuds (synsets) et des liens (relations) d'une partie de graphe représentant WordNet.....	38
Figure III.1: Environnement d'éclipse.....	41
Figure III.2: Extraits du code source du service web de SWS WSMO.....	42
Figure III.3 : résultat du code de filtrage.....	43
Figure III.4 : les concepts de l'ontologieseducation.owl.....	44
Figure III.5 : résultat de mesure de similarité.....	45
Figure III.6: les concepts de l'ontologies generations.owl.....	46
Figure III.7 : l'interface de notre application.....	47
Figure III.8 : le choix de service web et l'ontologie.....	48

Liste des Tableaux :

Tableau I.1 : Caractéristiques des principales approches des services Web sémantiques.....	21
Tableau II.1 : Un ensemble de relations représentées dans WordNet.....	39
Tableau III.1 : résultat de la mesure de similarité entre web service et l'ontologie education.owl.....	45
Tableau III.2 : résultat de la mesure de similarité entre le web service et l'ontologie generations.owl.....	47

INTRODUCTION GÉNÉRALE

Le World Wide Web (WWW), ou tout simplement le Web, permet d'accéder à des sources de données hétérogènes résidant n'importe où sur Internet. Au cours des dernières années, le Web a rencontré deux changements révolutionnaires qui visent à le transformer d'une collection de documents statiques en un environnement intelligent et dynamique pour l'intégration de données.

La première technologie est celle des services Web. Les services Web sont des applications auto-descriptives, modulaires et faiblement couplées qui fournissent un modèle simple de programmation et de déploiement d'applications, basé sur des normes, et s'exécutant au travers de l'infrastructure Web [2]. La collaboration entre les services Web s'appuie sur une architecture à trois composants à savoir : le fournisseur, le client et l'annuaire. Les services web offrent des fonctionnalités accessibles via le Web en utilisant un ensemble de technologies courantes, telles que SOAP, WSDL, UDDI et XML.

La deuxième technologie est dite le Web sémantique. C'est un Web intelligent qui permet d'enrichir les données existantes sur le Web par des descriptions formelles indiquant leur signification. La technologie du Web sémantique est basée sur les ontologies. Le Web sémantique, quant à lui, est une extension du Web actuel, dans lequel les informations ont un sens bien défini afin de permettre aux ordinateurs et aux utilisateurs de travailler en coopération [3].

L'application du Web sémantique aux services Web a donné naissance aux services Web sémantiques (SWS). D'une manière automatisée, les utilisateurs et les agents logiciels devraient pouvoir découvrir, appeler, composer, et surveiller l'exécution des services Web.

Plusieurs approches ont été développées pour les services Web sémantiques (OWL-S, WSMO, WSDL-S, etc.), ainsi chaque initiative a acquis un succès considérable et a traité certains aspects pragmatiques. Chaque approche peut être caractérisée par (1) un modèle conceptuel décrivant les principes fondamentaux et (2) un langage ou un ensemble de langages qui fournissent les moyens pour réaliser le modèle. Dans cette mémoire nous utilisons WSMO (*Web Service Modeling Ontology*) comme plateforme de modélisation des services Web sémantiques. WSMO possède quatre éléments principaux : Les *ontologies* définissent la terminologie utilisée par les autres éléments. Les *but*s indiquent ce que l'utilisateur attend du service. Les *services Web* définissent les fonctionnalités offertes. Les *médiateurs* lient les différents éléments afin de permettre l'interopérabilité entre les composants hétérogènes.

La découverte des SWS est le processus qui permet de localiser tous les services web qui matchent les besoins fonctionnels du demandeur (requête) [1], à cause du nombre important de services qui offrent des fonctionnalités similaires, cette phase peut retourner un nombre important de services comme résultat.

Notre travail présente une approche pour l'accélération de la découverte des services sémantiques WSMO en utilisant les ontologies de domaines et la technique de mesure de similarité. L'approche se déroule en deux phases successives :

1. L'extraction d'informations pertinentes à partir du code source des web services des SWS WSMO.
2. L'analyse de l'information extraite à l'aide de l'ontologie du domaine en faisant appel à une mesure de similarité.

Organisation du mémoire :

Le reste du mémoire est organisé en trois parties :

Chapitre I : Etat de l'art

Cette partie présente les notions de base qui doivent être acquises avant d'entamer notre problème. Elle présente ensuite un état de l'art des techniques et des approches existantes. Cette partie regroupe : *l'évolution du Web, Services Web sémantiques.*

Dans ce chapitre nous définissons la notion Des services Web sémantiques, ont été présentés tout en détaillant l'infrastructure et les ontologies du service. Nous avons terminé par une étude comparative entre les ontologies des services, tout en motivant le choix du WSMO dans notre approche pour SWS.

Chapitre II :

Dans ce chapitre on va expliquer et détailler notre approche qui est divisé en deux phases .la première consiste à extraite les informations pertinentes du service web de WSMO et les représentés et la deuxième phase permet d'analyser les informations pertinentes du SWS WSMO en utilisant l'ontologie du domaine et la mesure de similarité mais avant il faut déterminer les concepts de l'ontologie du domaine qui appartiennent à l'application Web, ainsi de suite on a défini et expliquer en détail la mesure de similarité utilisé pour l'analyse, et on a terminé le chapitre par la définition du WordNet pare ce que on a utilisé des approches qui se basent sur WordNet.

Chapitre III :

La troisième partie c'est « l'implémentation » comme son nom l'indique, nous permettra de vous présenter le fruit de notre travail. Dans cette partie on a représenté les résultats de notre approche on a utilisé deux ontologies pour voir la différence.

L'évolution du web

I.1 Introduction :

Aujourd'hui, le Web est un univers en perpétuelle expansion, constitué de pages et d'applications Web qui regorgent de vidéos, de photos et de contenu interactif. Et aussi le Web est un réservoir de ressources de tout genre. Plus ces ressources s'accumulent, plus la recherche d'information devient difficile. C'est pourquoi Tim Berners-Lee propose d'ajouter à toutes ces ressources une sémantique qui permettrait aux systèmes informatiques de comprendre le sens : c'est la naissance du Web sémantique. D'autre part, une autre technologie, dite les services Web, a permis de rendre accessible des programmes (services) via Internet selon des protocoles standards. L'ajout d'une couche sémantique, similaire à celle du Web sémantique, aux services Web, a permis de créer des services Web sémantiques .

I.2. L'histoire du web :

Tim Berners-Lee, un physicien britannique, a inventé le web au CERN¹ en 1989. À l'origine, le projet, baptisé « World Wide Web », a été conçu et développé pour que des scientifiques travaillant dans les universités et les instituts du monde entier puissent s'échanger des informations instantanément.

Le premier site web créé au CERN – et dans le monde – était destiné au projet World Wide Web lui-même. Il était hébergé sur l'ordinateur NeXT de Tim Berners-Lee. Le site décrivait personnes et comment configurer son propre serveur. L'ordinateur NeXT – le serveur web d'origine – est encore au CERN. En 2013, le CERN a entrepris de remettre en service le premier site web, et a même rétabli le site web à son adresse d'origine.

Le 30 avril 1993, le CERN a mis le logiciel du World Wide Web dans le domaine public. Puis il a émis la version suivante de l'application sous licence libre afin d'accélérer sa diffusion. En donnant libre accès au logiciel nécessaire pour faire fonctionner un serveur web, ainsi qu'au navigateur et à la bibliothèque de codes associés, il a permis à la Toile de se tisser [4].

I.3. Le web :

Le web était un système d'hypertextes distribués permettant l'accès à des sources de données hétérogènes utilisant un ensemble de standards pour : décrire les données (HTML), localiser les documents sur le réseau (URI) et accéder à ces documents (HTTP). Du fait de la simplicité du HTML qui permet la création de pages web avec un minimum d'efforts, le web a bénéficié d'un grand succès dans un temps relativement peu.

¹ CERN : Conseil européen pour la recherche nucléaire

Le nombre de sites Web et de serveurs Web a grimpé rapidement jusqu'à plus de 644 millions de sites Web le mois de Novembre 2015 [5].

Premièrement, la communauté du web vise une extension sémantique du web actuel syntaxique en augmentant les informations existantes sur le web avec des descriptions formelles de leurs significations en utilisant des descriptions formelles à base de logiques. Cette annotation sémantique sera appréhendable par la machine et permettra donc un accès et une intégration plus faciles de la quantité énorme d'information disponible par rapport à ce qui a été réalisé avec la recherche basée sur les mots clés. En second lieu, le web change d'une collection de pages web statiques en un environnement dynamique avec l'arrivée de la technologie des services web qui rend les composants logiciels accessibles via des protocoles web [6].

La technologie des services web sémantiques a vu le jour à travers ces deux dernières technologies en appliquant les techniques du web sémantique sur les services web.

I.4. Services web :

Le paradigme des services web repose sur une architecture par composants qui utilise des protocoles d'Internet comme infrastructure pour gérer la communication entre les composants. Les services web offrent un modèle à base de composants en ligne encapsulant une application logique derrière une interface interactive uniforme et standardisée. Les services Web sont basés sur le modèle SOA (service oriented architecture) [6].

I.4.1. Architecture SOA:

L'architecture SOA fournit un modèle de référence pour la mise en œuvre des services Web. La figure suivante illustre les composants de cette architecture [1].

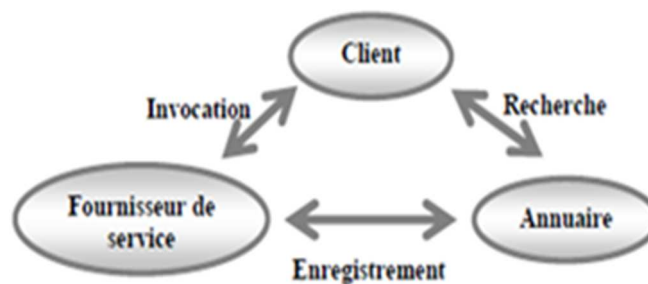


Figure I.1 : les éléments de base dans une architecture SOA

Les services Web peuvent être publiés, découverts et invoqués via un réseau en utilisant des protocoles standards. L'architecture SOA illustre trois types d'entités :

- **Client** : demande l'exécution d'un service Web. Un client peut être un utilisateur, une application ou même un service Web.
- **Fournisseur de service** : crée un service Web, le met à la disposition des clients et l'exécute en cas d'appel.
- **Annuaire** : est un registre pour la publication des services Web.

I.4.2. L'infrastructure des services web :

Bien que le web soit constitué de plateformes totalement hétérogènes où les intérêts des différents acteurs du marché s'entremêlent, ceci ne l'a pas empêché de se développer et d'être universel. Ce succès est dû essentiellement à l'édiction d'un ensemble de standards ouverts, dont les plus connus est le protocole HTTP. Le HTTP offre un mécanisme d'échange de données de toute sorte quel que soit la nature des plateformes impliquées.

Le cycle de vie d'un service web se déroule de la façon suivante : une fois créé, le service est d'employé sur le réseau (local ou Internet). Puis un utilisateur ayant des besoins spécifiques va rechercher un service correspondant à ses besoins à l'aide d'un annuaire spécialisé. Enfin, une fois le service trouvé, l'utilisateur va invoquer le service : une communication va être mise en place entre l'utilisateur et le service web. Ce cycle de vie, représenté par la figure 2 fait appel à trois grandes technologies : SOAP, WSDL, UDDI [6].

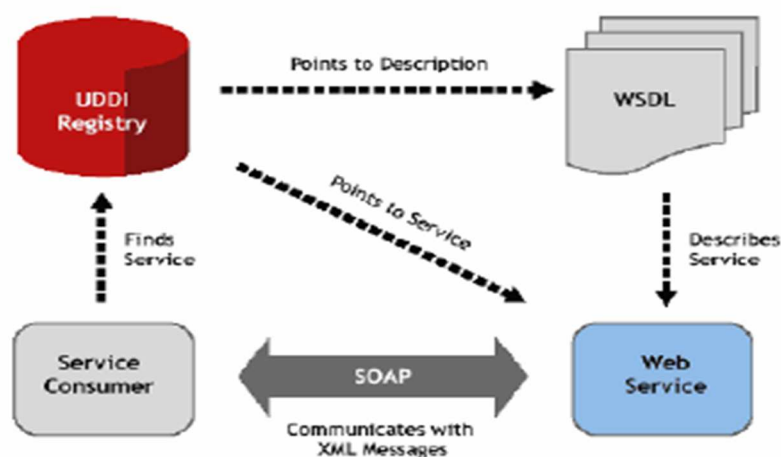


Figure I.2 : l'infrastructure des services web

1. **SOAP** (Simple Object Access Protocol) : assure la communication avec et inter services web.

2. **WSDL** (Web Services Description Language) : offre un schéma formel de description des services Web.

3. **UDDI** (Universal Description, Discovery and Integration) : offre une manière uniforme de définir des registres des services web et aussi un schéma uniformément extensible de descriptions des services web.

I.4.3 Les caractéristiques d'un service Web :

La technologie des services Web repose essentiellement sur une représentation standard des données (interfaces, messageries) au moyen du langage XML. Cette technologie est devenue la base de l'informatique distribuée sur Internet et offre beaucoup d'opportunités au développeur Web.

Un service Web possède les caractéristiques suivantes :

- Il est accessible via le réseau ;
- Il dispose d'une interface publique (ensemble d'opérations) décrite en XML ;
- Ses descriptions (fonctionnalités, comment l'invoquer et où le trouver ?) sont stockées dans un annuaire ;
- Il communique en utilisant des messages XML, ces messages sont transportés par des protocoles Internet (généralement HTTP, mais rien n'empêche d'utiliser d'autres protocoles de transfert tels : SMTP, FTP, BEEP..) ;
- L'intégration d'application en implémentant des services Web produit des systèmes faiblement couplés, le demandeur du service ne connaît pas forcément le fournisseur. Ce dernier peut disparaître sans perturber l'application cliente qui trouvera un autre fournisseur en cherchant dans l'annuaire.

I.5. Le web sémantique :

Le web sémantique a attiré depuis ces dernières années l'attention de nombreux chercheurs. 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 [1]. L'objectif du Web sémantique est de lever les difficultés rencontrées sur le Web d'aujourd'hui (recherche d'information, services, ...), en rendant la grande masse d'information disponible,

accessible et interprétable par les machines, en plus de l'automatisation de certaines fonctionnalités grâce à la représentation sémantique du contenu des documents, des services, des ressources sur le Web au sens large [7].

Les ontologies représentent la technologie clé pour la réalisation du Web sémantique.

I.5.1. Les Ontologies :

La technologie du Web sémantique s'appuie sur la notion d'ontologie qui représente la couche sémantique ajoutée au Web traditionnel.

Introduit en intelligence artificielle (IA) il y a 20 ans, le terme *ontologie* est cependant utilisé en philosophie depuis le XIX ième siècle [8]. Dans ce domaine, le terme ontologie signifie la science ou la théorie de l'être. En IA, et avec l'émergence de l'ingénierie des connaissances, les ontologies sont apparues comme des réponses aux problématiques de représentation et de manipulation des connaissances au sein des systèmes informatiques ils visent à établir des représentations à travers lesquelles les machines puissent manipuler la sémantique des informations. De même que les champs de l'ingénierie des connaissances, de la modélisation conceptuelle, et de la modélisation du domaine commençaient à converger, la signification du terme a évolué.

I.5.1.1. Rôle des ontologies au sein du web sémantique :

Dans le cadre du web sémantique, les ontologies sont utilisées comme structure sémantique centralisée pour :

- Donner un sens global aux balises des documents XML;
- Mettre à disposition un vocabulaire de termes sémantiques communs pour l'annotation partagée de documents web ;
- Permettre la réécriture de requêtes et la recherche d'informations adaptées basée sur la sémantique ;
- Permettre la réingénierie des systèmes d'information ;
- Assurer l'interopérabilité des systèmes d'information du web ;
- Fournir un ensemble de langages de métadonnées sémantiques de haut niveau d'écrivant les ressources sur le web.

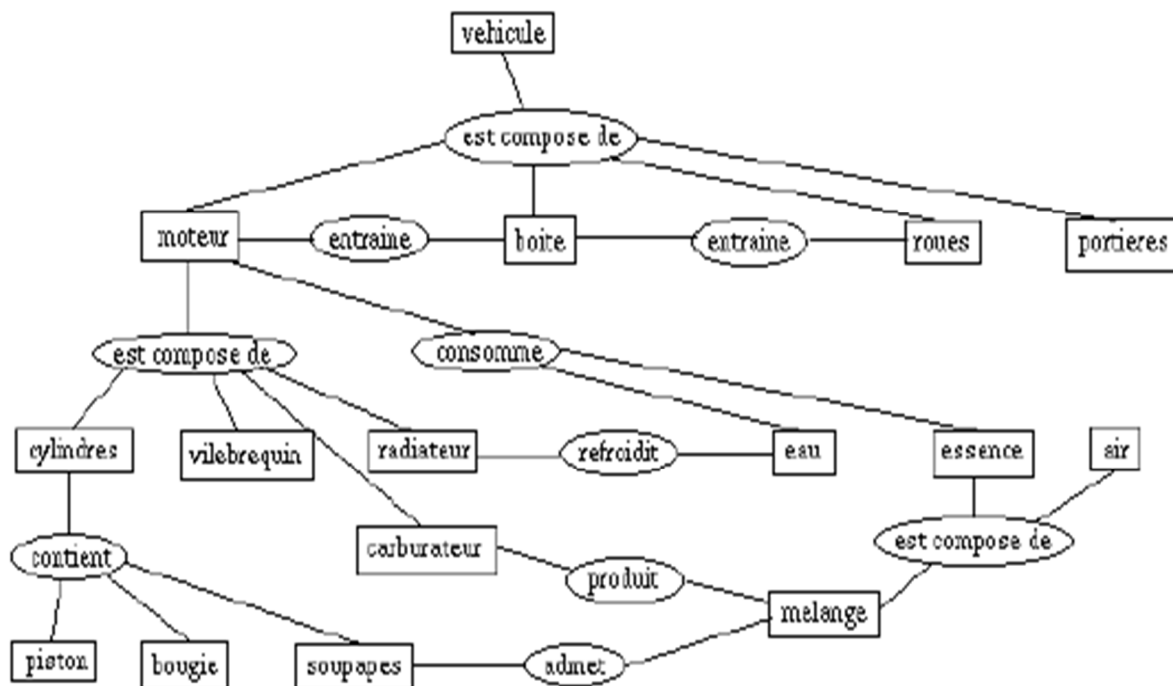


Figure I.3 : Exemple graphique d'une ontologie

I.5.1.2. Types d'ontologies :

Plusieurs diverses typologies d'ontologies ont été proposées dans la littérature.

- **Ontologies du domaine** : ces ontologies expriment des conceptualisations spécifiques à un domaine. Elles sont réutilisables pour plusieurs applications de ce domaine. L'ontologie du domaine caractérise la connaissance du domaine où la tâche est réalisée. Par exemple, dans le contexte du e-learning, le domaine peut être celui de la formation.
- **Ontologies applicatives** : ces ontologies contiennent des connaissances du domaine nécessaire à une application donnée, elles sont spécifiques et non réutilisables. Par exemple, dans le contexte du e-learning, une application peut être : la formation de statistiques et probabilités.
- **Ontologies génériques ou ontologies de haut niveau (top-ontologies)** : ces ontologies expriment des conceptualisations valables dans différents domaines.

Ce type d'ontologies est fondé sur la théorie de l'identité. Son sujet est l'étude des catégories des choses qui existent dans le monde. Comme les concepts de haute abstraction tels que les entités, les évènements, les états, les processus, les actions, le temps, l'espace, les relations, les propriétés, etc.

I.6. Les services Web sémantique :

Les standards utilisés par les services Web offrent des définitions syntaxiques qui ne permettent pas de décrire complètement les fonctionnalités des services et ne peuvent pas être comprises par les programmes. Il faut l'intervention des humains pour interpréter la signification des entrées, des sorties, des contraintes et du contexte dans lequel les services peuvent être employés.

La technologie du Web sémantique a permis de doter les services Web par une couche sémantique. La combinaison entre le Web sémantique et les services Web a permis de créer une nouvelle technologie dite les services Web sémantiques [1].

Les services Web sémantiques sont nécessaires dans le but d'automatiser les fonctionnalités suivantes:

- ***La découverte automatique des services Web*** : ça consiste à localiser automatiquement des services Web qui satisfont des contraintes exigées. Grâce aux SWS, les informations nécessaires pour la découverte des services Web peuvent être spécifiées formellement.
- ***L'invocation automatique des services Web*** : ça consiste à exécuter automatiquement un service Web identifié par un utilisateur ou un agent logiciel. Après une requête, un service Web est exécuté comme un ensemble d'appels de fonction grâce à une API fournie par les SWS.
- ***La composition et l'interopérabilité automatique des services Web*** : cette tâche implique la sélection, la composition et l'interopérabilité automatique des services Web pour effectuer d'autres tâches. Grâce aux SWS, les informations nécessaires pour sélectionner et composer des services vont être encodées avec ces services Web.
- ***La surveillance de l'exécution automatique d'un service Web*** : cette tâche permet de savoir l'état de la requête d'un client pendant l'exécution d'un service Web. Ainsi, les SWS fournissent des descriptions de l'état d'exécution des services [1].

I.6.1. L'infrastructure des services Web sémantiques

Le développement des services Web sémantiques (SWS) est effectué en trois dimensions: les activités d'utilisation, l'architecture et l'ontologie de service (Figure I-4).

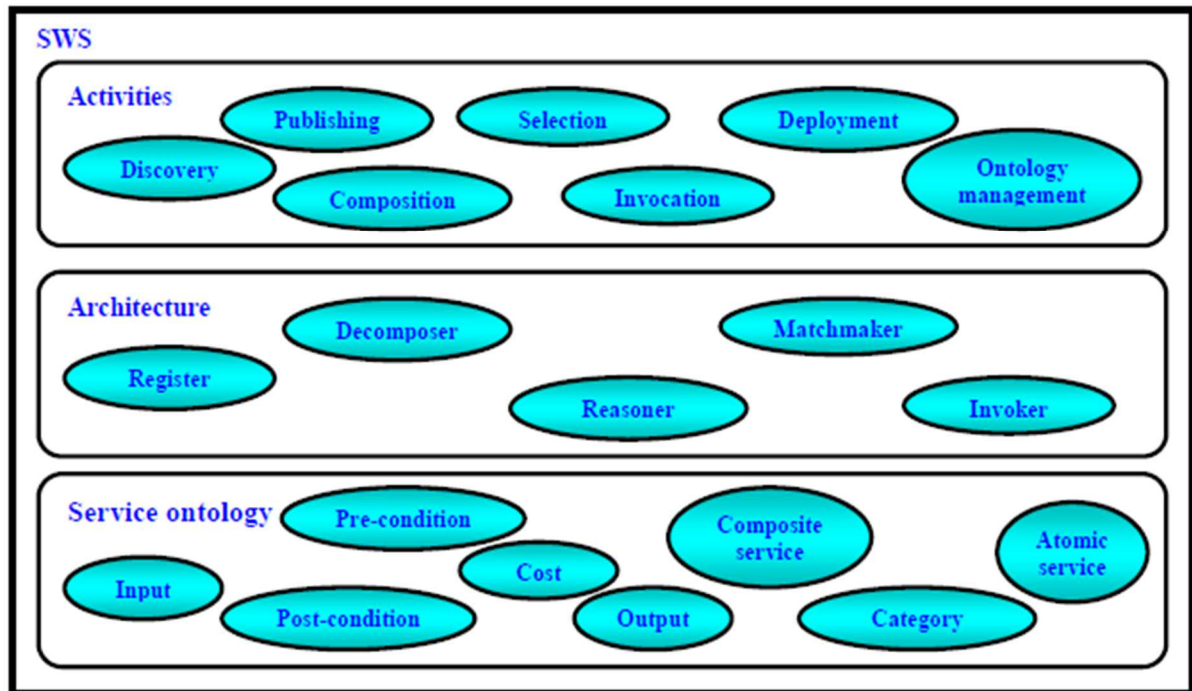


Figure I.4 : Infrastructure des services Web sémantiques.

Ces trois dimensions sont relatives aux besoins des SWS au niveau application, physique et conceptuel. **Les activités d'utilisation** définissent les besoins fonctionnels qu'un framework implémentant les SWS devrait supporter. **L'architecture des SWS** décrit les composants nécessaires pour accomplir ces activités. **L'ontologie de service** intègre tous les concepts utilisés pour la description des services Web sémantiques.

I.6.2. Approches proposées pour la réalisation des services Web sémantiques :

Dans la littérature, diverses approches ont été proposées pour permettre la réalisation des services Web sémantiques. Parmi ces travaux on a : WSDL-S, OWL-S et le WSMO.

1. WSDL-S : (Web Service Description Language-Semantic)

Est un langage de description sémantique des services Web. Propose un mécanisme pour lier les descriptions fonctionnelles existantes dans le WSDL d'un service Web avec une sémantique.

2. OWL-S (Ontology Web Language for Services):

Est un langage d'ontologie pour services Web. L'OWL-S définit un ensemble d'ontologies de haut niveau utilisées pour décrire les différents aspects d'un service Web

sémantique. Cet ensemble est organisé en trois zones conceptuelles : "ServiceProfile", "ServiceModel" et "ServiceGrounding" [1].

- La classe *ServiceProfile* donne les informations nécessaires à un agent pour publier ou découvrir un service.
- La classe *ServiceModel* donne les conditions nécessaires à l'exécution d'un service ainsi que ses conséquences. Il inclut des informations concernant les entrées, sorties, pré-conditions et effets d'un service
- La classe *ServiceGrounding* spécifie la manière, pour un agent, d'accéder au service concerné.

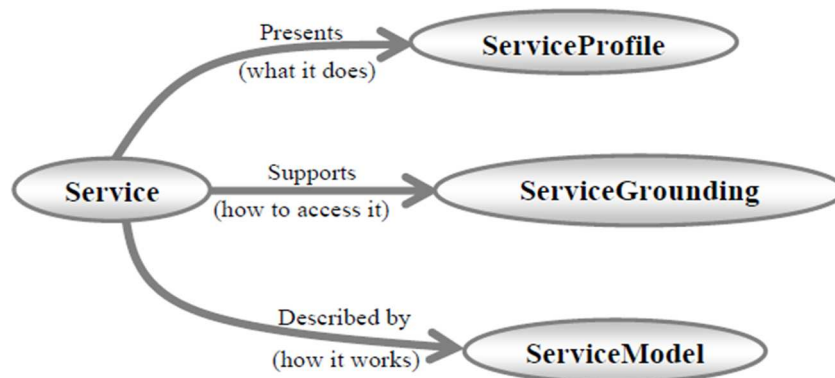


Figure I.5 : Les éléments de base d'OWL-S.

3. Le WSMO (Web Service Modeling Ontology):

Le WSMO (Web Service Modeling Ontology) [9] a un but de décrire tous les aspects liés aux services Web sémantiques. Le WSMO propose un modèle conceptuel basé sur WSMF (Web Services Modeling Framework) [8]. Pour la spécification de ses éléments, comme langage le WSMO utilise WSML (Web Service Modeling Language).

- Le WSMO est un modèle conceptuel qui vise à décrire tous les aspects liés aux services et qui sont accessibles par une interface de service web [7].
- WSMO fournit des spécifications ontologiques pour les éléments du noyau des services web sémantiques, son but final est de permettre l'automatisation (totale ou partielle) des activités telles que la découverte, la sélection, la composition, la médiation, l'exécution et la surveillance impliquées dans l'intégration intra et inter entreprise des services web. WSMO fournit trois principales catégories pour structurer les descriptions sémantiques. Premièrement, il fournit des moyens pour décrire les services web ; en second lieu, il

fournit des moyens pour décrire les objectifs des utilisateurs se rapportant à la résolution des problèmes. Troisièmement, il fournit des moyens pour assurer l'interopérabilité entre les diverses descriptions sémantiques des environnements hétérogènes : ontologies et médiateurs [9].

I.6.3. Comparaison et discussion :

Le tableau suivant représente les principales caractéristiques des approches de spécification des services Web sémantiques.

	Owl-S	WSMO	WSDL-S
Porté	Modèle pour la description sémantique des services web	Modèle + langage pour la description sémantique des services web	Annotation sémantique des descriptions WSDL
Architecture	DAML-S Virtual Machine Matchmaker	Service Registry Profile Crawler	Mots clés Classification Ontologie de domaine
Activités d SWS	Découverte Exécution Interopération Composition Surveillance	Découverte Médiation Composition Grounding Exécution	Découverte Composition Interopération
Eléments principaux	Service Profile Service Model Service Grounding	Ontologies Buts (Goals) Services web Médiateurs	Opérations/ WSDL descriptions
L'ontologie de service	OWL-S	WSMO	Non définie
Langage	OWL	WSML	Non définie

Tableau I.1 : Caractéristiques des principales approches des services Web sémantiques

Les approches sont présentées selon : leurs portées (i.e. les moyens offerts pour la mise en œuvre des SWS), l'architecture sous-jacente, les activités permises par les SWS, les éléments principaux, l'ontologie de service et le langage utilisé pour la spécification des éléments.

Les approches les plus répandues pour la mise en œuvre des SWS sont OWL-S et WSMO [10]. Dans ce rapport on va opter pour WSMO comme plateforme de description des SWS. Donc on a favorisé WSMO par rapport à OWL-S pour les raisons suivantes :

- L'approche OWL-S est basée sur le langage OWL. Cependant, OWL n'a pas été conçu pour la définition de la sémantique des processus. Ainsi, OWL-S possède une expressivité assez limitée. De ce fait, WSMO/WSML était conçu pour remédier à ces limitations en fournissant plusieurs couches d'expressivité, permettant ainsi de fournir des définitions riches aux services Web.
- OWL-S hérite les insuffisances d'OWL [11]. OWL-S ne fait aucune distinction explicite entre la communication et la coopération entre les services Web. WSMO fait cette distinction en termes de chorégraphie et orchestration en appliquant ainsi le principe de séparation des activités entre la communication et la coopération et rendant le modèle conceptuel d'autant plus claire.
- OWL-S n'a pas considéré explicitement le problème d'hétérogénéité dans l'ontologie de service elle-même. Les médiateurs ne sont pas des éléments de l'ontologie mais plutôt une partie de l'infrastructure sous-jacente aux services Web. WSMO considère les médiateurs comme une partie de l'ontologie de service.
- Le concept de médiateur de WSMO est très important pour résoudre le problème d'interopérabilité. La définition des buts est nécessaire pour l'automatisation de la chorégraphie et Services Web Sémantiques de l'orchestration. WSMO fait une distinction entre le service désiré (but) et le service offert ; ce qui le différencie du OWL-S qui ne modélise pas les buts. WSMO offre la possibilité d'ajouter des propriétés non fonctionnelles qui peuvent être utilisées pour décrire plusieurs aspects, tels que le temps de réponse.

I.7. Les éléments de WSMO :

WSMO consiste en quatre éléments : ontologies, Web Services, Goalset Médiateurs.

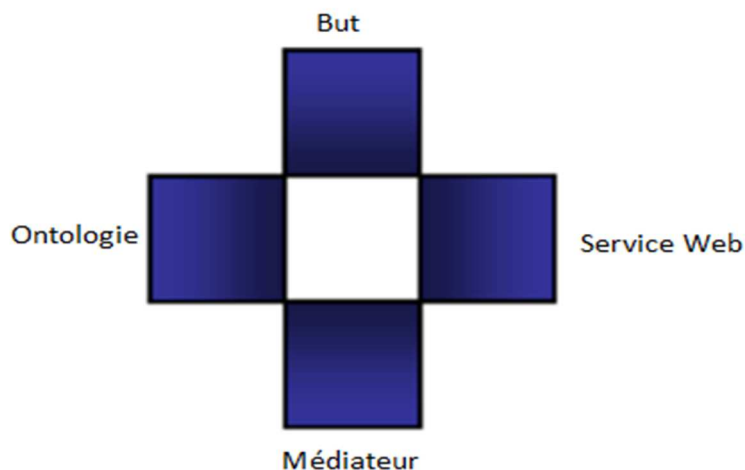


Figure I.6 : Les éléments qui composent le WSMO

I.7.1. Les buts:

Les buts fournissent les moyens pour déterminer les demandes des utilisateurs en termes de conditions fonctionnelles et non fonctionnelles. Les buts décrivent les objectifs des utilisateurs en recherchant un service web. Ils décrivent les aspects liés aux désirs des utilisateurs en ce qui concerne la fonctionnalité demandée [9].

I.7.2. Ontologies :

Les ontologies fournissent une spécification formelle et explicite des vocabulaires utilisés par les autres éléments de modélisation [9]. En outre les ontologies facilitent l'interaction entre les ressources de l'information par des liens à la même ontologie ou des liens entre les ontologies.

I.7.3. Services Web:

L'élément Service Web du WSMO fournit un modèle conceptuel pour décrire d'une manière explicite et unifiée tous les aspects du service, en incluant ces propriétés non fonctionnelles, sa fonctionnalité et les interfaces indispensables pour l'obtenir. Un service web est une entité logicielle capable par invocation d'atteindre les objectifs des utilisateurs (goals) et qui peut fournir différents services [8].

I.7.4. Médiateurs :

Les médiateurs jouent un rôle central dans WSMO car ils permettent de résoudre les hétérogénéités qui peuvent surgir entre les composants du WSMO [9]. Les médiateurs du WSMO connectent les éléments du WSMO d'une manière faiblement couplée.

I.7.5. Le Langage :

Le WSML (Web Service Modeling Language) est un langage formel pour la description des ontologies et des services web sémantiques qui prend en considération tous les aspects des descriptions de service web identifiés par WSMO.

Dans sa version courante, WSML², pour la description fonctionnelle des goals et des services web, WSML offre un cadre syntaxique. Cependant, WSML ne spécifie pas formellement encore la sémantique exacte des descriptions fonctionnelles des services. La description du comportement dynamique des services web (chorégraphie et orchestration) dans le contexte du WSML, est actuellement en cours d'étude, et n'a pas été encore intégrée dans WSML.

² <http://www.wsmo.org/wsml/wsml-syntax>

I.7.5.1. Syntaxe conceptuelle : La syntaxe conceptuelle du WSML permet la modélisation des ontologies, Web Services, goals et mediators [1].

- Ontologies. Une ontologie dans WSML se consiste des éléments suivants : concept, relation, instance, relationInstance et axiom. En complément, une ontologie peut avoir des propriétés non fonctionnelles et peut importer d'autres ontologies.
- Web Services. Conformément au modèle conceptuel du WSMO, un Web Service possède une capacité et un nombre d'interfaces. (Figure I.7)

```

webService _"http://example.org/bookService"
  nonFunctionalProperties {_"http://example.org/bookOntology#",
    dc#title hasValue "Example Book Buying Service"
    dc#description hasValue "A simple example Web service for adding items
      to a shopping cart"
  }
  endnonFunctionalProperties {_"http://example.org/bookOntology#",
  importsOntology _"http://example.org/bookOntology"
  capability
    sharedVariables { ?cartId , ?item}
    precondition
      definedBy
        ?cartId memberOf string and item memberOf book.
    postcondition
      definedBy
        forall cart ( cart [ id hasValue cartId ] memberOf cart implies
          cart [ items hasValue item] ) .
  }

```

Figure I.7 : Un exemple d'un Web Service WSML

Le constructeur **sharedVariables** est utilisé pour identifier les variables qui sont partagées entre les préconditions et les postconditions et entre les assumptions et les effets.

I.7.5.2. Syntaxe des expressions logiques :

La syntaxe générale des expressions logiques pour WSML a un style de logique du premier ordre, dans le sens où elle a des constantes, des symboles, des fonctions, des variables, des prédicats et des liaisons logiques habituelles. En outre, WSML a des extensions basées sur F-Logic pour modéliser les concepts, les attributs, les définitions des attributs, les subconcepts et les relationships entre les concepts membership [1]

I.8. Conclusion :

Dans ce chapitre on a intéressé aux deux technologies mères des SWS, à savoir le Web sémantique et les services Web. Ensuite les services Web sémantiques ont été présentés tout en détaillant l'infrastructure et les ontologies de service ainsi on a fait une étude comparative entre les ontologies de service. On a terminé par la présentation des éléments de base constituant WSMO : les *ontologies* qui fournissent la terminologie utilisée par les autres éléments WSMO ; les *services Web* qui fournissent un accès aux services ; les *buts* qui représentent les désirs de l'utilisateur ; et les *médiateurs* qui traitent le problème d'interopérabilité entre les différents éléments de WSMO.

L'approche proposée

II.1. Introduction :

Dans tous les domaines de l'informatique dans lesquels on désire analyser de manière automatique un ensemble de données (au sens le plus large) il est nécessaire de disposer d'un opérateur capable d'évaluer précisément les ressemblances ou les dissemblances qui existent au sein de ces données. Sur cette base, il devient alors possible d'ordonner les éléments de l'ensemble, de les hiérarchiser ou encore d'en extraire des invariants. Pour qualifier cet opérateur nous utiliserons dans ce chapitre le terme de fonction de similarité ou plus simplement celui de similarité, on va détailler notre approche et expliquer les méthodes utilisées.

II.2. Notre Approche :

Dans ce projet nous allons diviser notre approche en deux phases. La première phase est l'extraction d'informations pertinentes à partir du service web. L'élément de base étant un SWS WSMO. La deuxième phase est l'analyse qui commence par une comparaison réalisée entre les informations extraites précédemment et les concepts d'ontologie du domaine en faisant appel à des techniques de distance sémantique. Cette opération de comparaison sémantique permet le calcul des liens sémantiques. Il en résultera le domaine de SWS WSMO et l'inférence de nouvelles ontologies importées et du même domaine des SWS. Initialement, nous supposons que :

- A partir des services web, nous pouvons extraire des concepts ou bien des informations et les représenter dans un tableau.
- Un concept de l'ontologie du domaine est indiqué par un champ du tableau.
- Il existe au préalable une ontologie qui est construite par des experts et que cette ontologie est spécifique à un domaine.

Nous présentons ensuite, dans leurs détails, les étapes du processus d'accélération de la découverte des SWS WSMO.

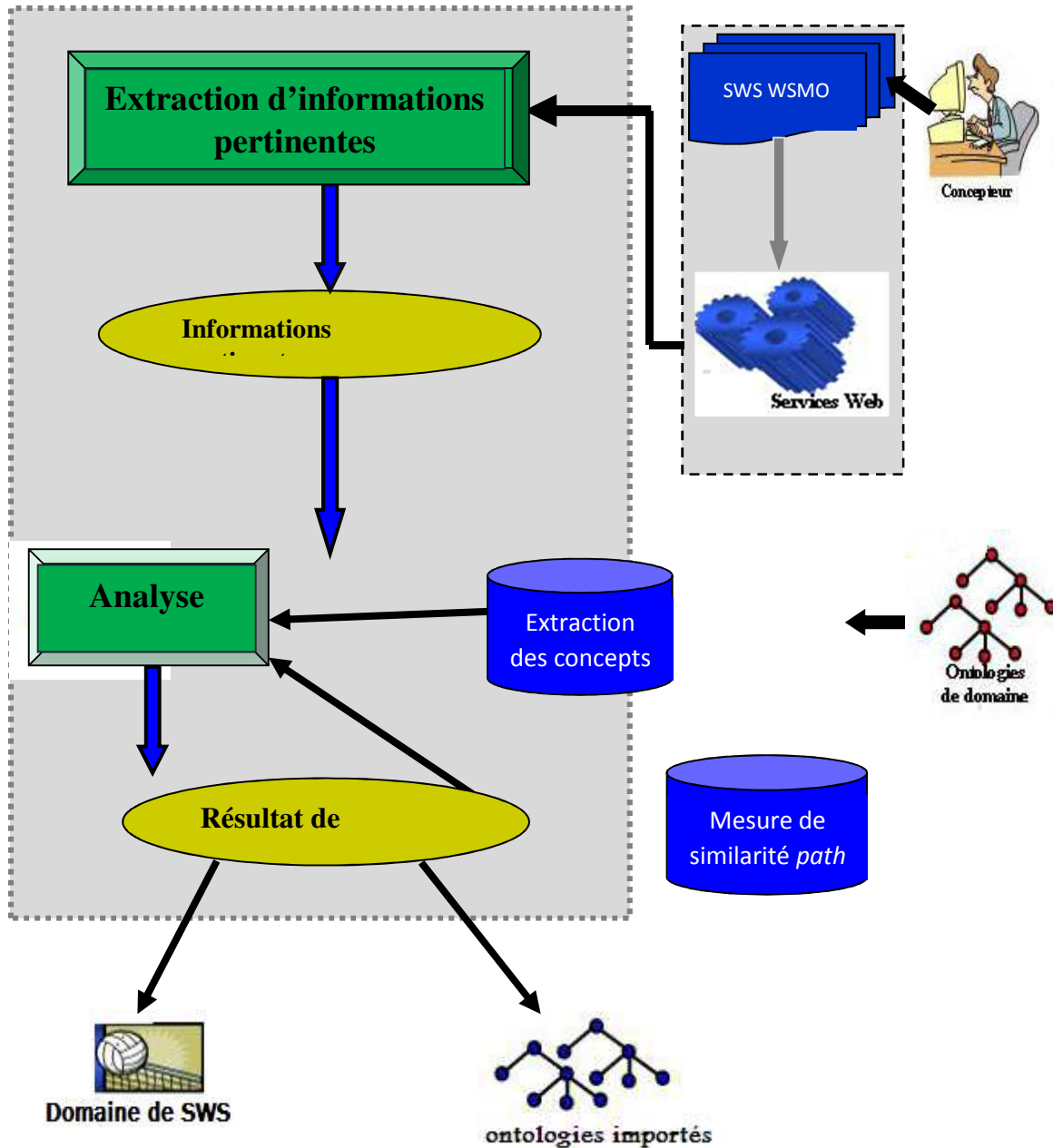


Figure II.1: Processus d'accélérer la découverte SWS WSMO.

II.3. Algorithme de l'approche proposée :

Comme il est décrit, notre approche consiste à accélérer la découverte des SWS WSMO et de plus établir un ensemble enrichi de concepts et de relations. Cet ensemble réfère en premier lieu aux concepts identifiés dans l'étape précédente. Il est en second lieu enrichi des relations de l'ontologie possédant comme extrémité l'un des concepts identifiés.

Les concepts de l'autre bout de chaque relation sont aussi ajoutés à cet ensemble. A ce niveau-là, des groupes de concepts seront formés. Chaque groupe représente un domaine pour

la classification de notre objectif (SWS WSMO) et même un ensemble des ontologies importées selon chaque concept O_j de la deuxième étape (cf. figure II.2).

Algorithme :accélérer la découverte des SWS WSMO
<p>Entrées Les concepts d'Ontologie du domaine. Les informations pertinentes du service web (champs de tableaux). Une mesure de similarité. Une stratégie de calcul de similarité entre ensembles « la méthode path ». Un seuil pour la mesure de similarité.</p> <p>Sorties Un ensemble de concepts.</p> <p>Début-Algo 1- Créer un tableau d'éléments V1. Chaque élément possède comme attributs les champs des informations pertinentes du service web SWS WSMO . 2- Créer un tableau d'éléments V2. Chaque élément correspond à un concept de l'ontologie de domaine. 3- Pour chaque élément O_i du tableau V1 Faire Pour chaque élément O_j du tableau de concepts V2 Faire Calculer la similarité entre le concept de l'élément O_i et les concepts de l'élément O_j un par un ; Si pour un élément de O_i la similarité est égal au seuil, Alors marquer l'élément O_i correspondant au "domaine" de l'élément O_j . Fin-Si Fin-Pour Fin-Pour Fin-Algo</p>

Figure II.2: L'algorithme proposé

II.3.1 Phase extraction des informations pertinentes :

C'est la phase de l'extraction d'informations retenues du service web. Dans cette phase on va commencer par le filtrage de l'élément structurant le service web du SWS WSMO suivi de l'extraction des mots clés {' , ' memberOf' } qui se trouvent en variables partagées des services web et enfin l'extraction des informations pertinentes qui se trouvent après le mot clé ' memberOf' et avant le mot clé '['.

Le filtrage consiste à parcourir le code du service web, et garder uniquement les plus pertinentes, le résultat en est un ensemble de concepts. Ces concepts vont être représentés sous forme de tableaux logiques pour faciliter leurs manipulations.

II.3.2. Phase d'analyse :

II.3.2.1 Identification des concepts d'ontologie

Ce traitement permet de déterminer les concepts de l'ontologie du domaine qui appartiennent à l'application Web.

Il est appliqué aux ontologies du domaine, et consiste à enlever les espaces, les traits d'union, etc. et garder uniquement les racines des termes telles qu'elles apparaissent dans WordNet¹. On en recueille un ensemble des termes qui peuvent être identifiés plus tard comme des concepts d'ontologie.

Maintenant, nous allons configurer notre projet Java et donner tous les étapes de l'application pour instancier automatiquement une ontologie OWL développée sous protégé² par un programme Java (sous eclipse). Avant de vous lancer un projet OWL, il faut sélectionner le dossier d'installation Protégé² de votre projet.

Puis ajouter tous les fichiers jar de l'installation à votre classpath du projet. Réglez votre chemin de sortie du compilateur pour les plugins et classes.

Notre programme java permet de lire une ontologie et stocker ses éléments dans un tableau de chaîne de caractères nommé "concepts", ensuite affiche les éléments de ce tableau.

Auparavant, il faut créer un tableau de caractère nommé le tableauontologie pour le remplir avec les concepts d'ontologie, Puis on a choisi notre ontologie owl, le programme qui suit permet la fonction qui clarifier le chargement de notre fichier (ontologie.owl).

```
    //***** Creer un modele OWL*****//
    JenaOWLModel owlModel = ProtegeOWL.createJenaOWLModel();

    //***** Chargement du fichier*****//

        try {

            URI fileURI = new URI(URL); // URL= ontologie .owl //
            owlModel.load(fileURI,FileUtils.langXMLAbbrev);

        }

        catch (Exception e) {

            e.printStackTrace();

        }

    }
```

Figure II.3: la fonction qui clarifier le chargement de notre fichier (ontologie.owl).

¹ WordNet est une base de données lexicale qui organise les noms et les verbes dans des concepts (synset) en hiérarchies de relations is-a. Chaque concept est décrit par une brève close. (Voir aussi annexe wordnet).

² <http://protege.stanford.edu/download/download.html>

L'interface la plus important est `OWLModel` , qui permet d'accéder au contenu de haut niveau des ressources dans l'ontologie. Vous pouvez utiliser `OWLModel` pour créer, interroger et supprimer des ressources de différents types, puis utilisez les objets retournés par la `OWLModel` d'effectuer des opérations spécifiques. Dans notre programme, on va créer un nouveau `OWLNamedClass` (ce qui correspond à `owl: Class` dans le langage OWL), puis obtient son URI.

Et on utilise un `Iterator` pour le parcours de cette collection d'ontologie owl , qui sert à afficher la hiérarchie des sous classes d'une classe (`ontologie.owl`) comme suit :

```
//----- Fonction récursive pour afficher l'hierarchie des sous classes d'une classe -----//
private void printClassTree(RDFSCls cls, String indentation) {
    tableauontologie [Nbr] = cls.getName(); Nbr+=1;
    for (Iterator it = cls.getSubclasses(false).iterator(); it.hasNext(); ) {
        RDFSCls subclass = (RDFSCls) it.next();
        if (! cls.isSubclassOf(subclass)) printClassTree(subclass, indentation + "  ");
    }
}

//***** Afficher l'hierarchie des classes *****//
//System.out.println("\n---- filtrage d'une ontologie :");
RDFProperty subC = owlModel.getRDFProperty(RDFSNames.Slot.SUB_CLASS_OF);
OWLNamedClass owlT = owlModel.getOWLThingClass();
Collection results = owlModel.getRDFResourcesWithPropertyValue(subC, owlT);
```

Figure II.4: une partie du programme `OWLModel`.

L'API `Protege-OWL` fait une distinction claire entre les classes nommées et les classes anonymes. Les classes nommées sont utilisées pour créer des individus (les concepts), alors que les classes anonymes sont utilisées pour spécifier les caractéristiques logiques (restrictions) des classes nommées. Nous nous intéressons aux classes nommées.

Dans notre programme, il ya deux types de classes nommées : `RDFSCls` (`rdfs: Class`) et `OWLNamedClass` (`owl: Class`). Une fois que vous avez créé les classes, vous pouvez les organiser dans une relation sous-classe.

Les classes et les classes de sous-classe sont définies plus tard les concepts d'ontologie owl.

Le stockage des éléments du tableau se fait au niveau de la procédure `printClassTree`.

Plus exactement, il faut stocker la variable indentation dans une case du tableau et avancer à la case suivante.

II.3.2.2 la mesure de similarité :

Ainsi de suite, on va calculer la similarité entre les informations retenues du service web qui se trouvent actuellement dans un tableau et les concepts des ontologies du domaine.

Une mesure de similarité vise à quantifier la proximité sémantique des deux concepts. Pour calculer cette similarité sémantique, nous avons utilisé des approches qui se basent sur les liens taxonomiques « is-a » (hyperonymes) du WordNet.

Pour appliquer la similarité, d'abord on a fixé un seuil. Le seuil est une valeur prise entre 0 et 1. La valeur 1 indique qu'il y a une équivalence sémantique totale entre deux concepts.

Il existe plusieurs techniques pour le calcul de la mesure de similarité (on va les détailler ci-dessous), dans notre approche nous avons utilisé la mesure de similarité basées sur la longueur du chemin entre les concepts.

II.4. Les techniques de la mesure de similarité :

La similarité joue un rôle central dans notre processus d'accélération des services web sémantiques. Elle renvoie une valeur numérique indiquant si les deux éléments ont un degré élevé ou bas de similitude.

Pour calculer cette similarité sémantique, nous avons utilisé des approches qui se basent sur WordNet (on va le détailler dans la section II.5). Ces techniques peuvent être classifiées comme suit :

II.4.1. Les mesures de similarité basées sur la longueur du chemin entre les concepts :

La mesure [12] détermine le plus court chemin entre deux concepts et multiplie cette valeur par le maximum des deux chemins là où ils apparaissent dans la hiérarchie « is-a ». La mesure [13] 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, cette mesure est calculée comme suit :

$$\text{sim}(c_1, c_2) = \frac{2\text{prof}(c)}{\text{prof}(c_1) + \text{prof}(c_2)}$$

La mesure *path* est égale à l'inverse de la longueur du plus court chemin entre les deux concepts. Pour notre approche on va utiliser la mesure de similarité *path* (pattern matching)

$$A * A \rightarrow R$$

$$sim(c_1, c_2) = 1/d(c_1, c_2)$$

$d(c_1, c_1) = 0$ et par la suite $sim(c_1, c_1)$ tend vers l'infini.

- Pour cette raison on pose :

$$sim(c_1, c_2) = 1/1+d(c_1, c_2)$$

Ce qui donne $sim(c_1, c_1)=1$, et pour tout autre objet c_2 , $sim(c_1, c_2) < 1$.

Pour calculer cette distance on utilise WordNet. Dans notre évaluation, nous s'intéressons à la relation hyponymie « is-a ».

Dans l'exemple suivant on va montrer comment on va calculer la similarité entre deux concepts, en utilisant bien la hiérarchie du Word Net et la mesure de similarité path.

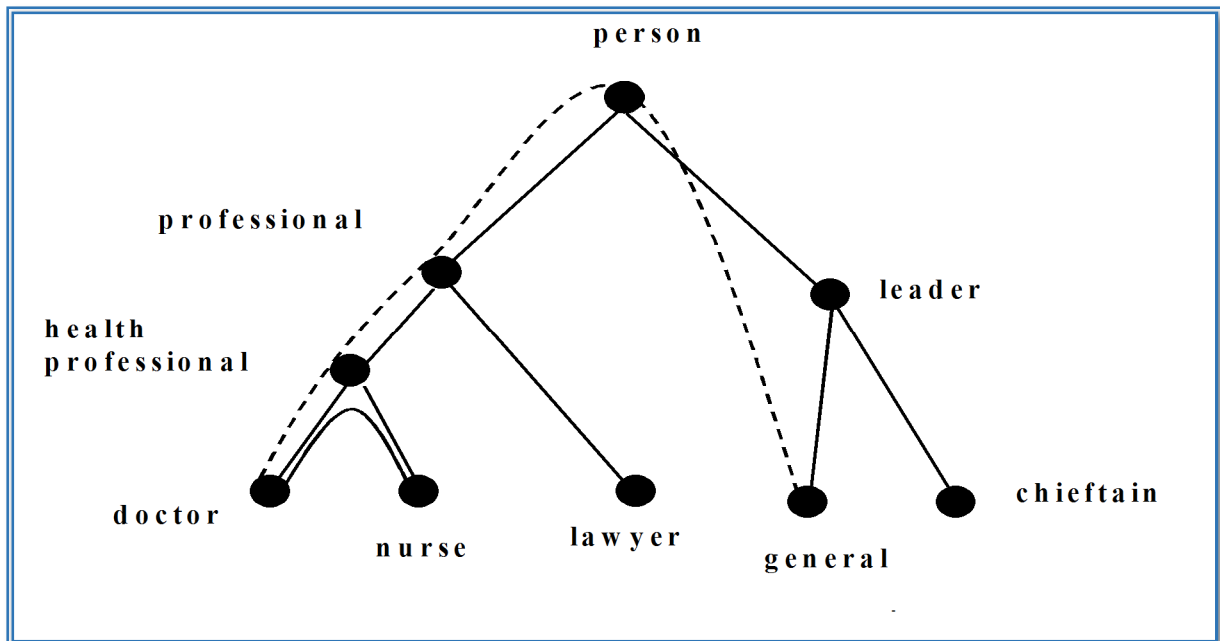


Figure II.5: Un fragment du WordNet, à partir duquel on déduit que doctor est plus similaire à nurse

Nous considérons les paires $(doctor, nurse)$ et $(doctor, general)$ de la figure II.5. Dans WordNet, le join de $doctor$ et $nurse$ ($doctor \vee nurse$) est le nœud *healthprofessional*. Alors la distance entre $doctor \vee nurse$ est 1 d'où la similarité est égale à :

$$Sim(doctor, nurse) = 1/1+1 = 1/2 = 0.5$$

D'autre part, le join $(doctor \vee general)$ est le nœud *person*. Alors la distance entre $doctor \vee general$ est 3, d'où la similarité est égale à

$$\text{Sim}(\text{doctor}, \text{general}) = 1/1+3 = 1/4 = 0.25$$

Le fait de dire, que $0.5 > 0.25$ on déduit que *doctor* est plus similaire a *nurse*.

Donc pour la similarité entre les concepts *doctoret nurse*, sont similaires car la distance qui les sépare dans le graphe est faible, la distance dans un graphe étant donnée par le plus court chemin à parcourir le long des arcs pour joindre un concept a partir d'un autre.

Le calcul du degré de match des concepts c_1 et c_2 est réalisé par la fonction *path*tel que :

$$\text{Sim}(c_1, c_2) = \begin{cases} 1 & \text{si } c_1 \text{ equivalentClass } c_2 \\ 0.5 & \text{si } c_1 \text{ subClass } c_2 \\ 0 & \text{sinon} \end{cases}$$

En résumé le pattern matching (path) constitue à notre sens le meilleur moyen pour mettre en œuvre l'intégration des concepts des services web et des ontologies.

II.4.2. Les mesures basées sur le contenu d'information :

La mesure *RES* [14] est basée sur le contenu d'information. Les mesures *LIN* [15] et *JCN* [16] 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.

II.4.3. Les mesures basées sur le type des relations entre les concepts :

La mesure *HCO* [17] est basée sur le chemin, et classifie les relations dans WordNet selon leurs directions. Elle évalue la similarité entre les concepts en déterminant le plus court chemin avec le moindre changement de direction.

La mesure *LESK* [18] 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* [19], 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.

Le calcul de la similarité peut être aussi effectué entre deux groupes d'entités en utilisant l'une des stratégies suivantes.

- a) **Lien simple** : La similarité entre deux groupes est celle des deux membres les plus proches (similaires) dans ces deux groupes. Elle est calculée comme suit :

$$\text{Sim}(r,s) = \text{Max} \{ \text{sim}(i,j) : \text{où } i \text{ est dans le groupe } r \text{ et } j \text{ est dans le groupe } s \}$$

- b) **Lien complet** : La similarité entre deux groupes est celle des deux membres les plus dissimilaires dans ces deux groupes. Elle est calculée comme suit :

$$\text{Sim}(r,s) = \text{Min} \{ \text{sim}(i,j) : \text{où } i \text{ est dans le groupe } r \text{ et } j \text{ est dans le groupe } s \}$$

- c) **Lien moyen** : La similarité entre deux groupes est la moyenne des similarités entre les membres des deux groupes.

$$\text{Sim}(r,s) = \text{Trs} / (\text{Nr} * \text{Ns}) ; \text{où } \text{Trs} \text{ est la somme de toutes les similarités, } \text{Nr} \text{ et } \text{Ns} \text{ est le nombre d'éléments de } r \text{ et de } s.$$

II.4.4. La mesure de Wu-Palmer :

Dans un domaine de concepts, la similarité est définie par rapport à la distance qui sépare deux concepts dans la hiérarchie et également par leur position par rapport à la racine. La similarité entre C1 et C2 est :

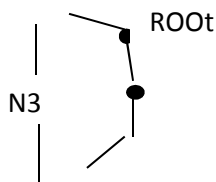
$$\text{ConSim}(C1, C2) = \frac{2 * N3}{N1 + N2 + 2 * N3}$$

Plus formellement cette mesure devient :

$$\text{ConSim}(C1, C2) = \frac{2 * \text{depth}(C)}{\text{depth}_C(C1) + \text{depth}_C(C2)}$$

Où C est le plus petit généralisant (PPG) de C1 et C2 (en nombre d'arcs), depth(C) est le nombre d'arcs qui sépare C de la racine et depth_C(Ci) avec i le nombre d'arcs qui séparent Ci de la racine en passant par C.

Cette mesure a l'avantage d'être simple à implémenter et d'avoir d'aussi bonnes performances que les autres mesures de similarité [15].



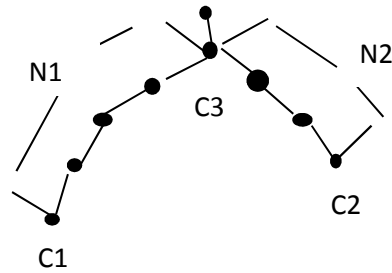


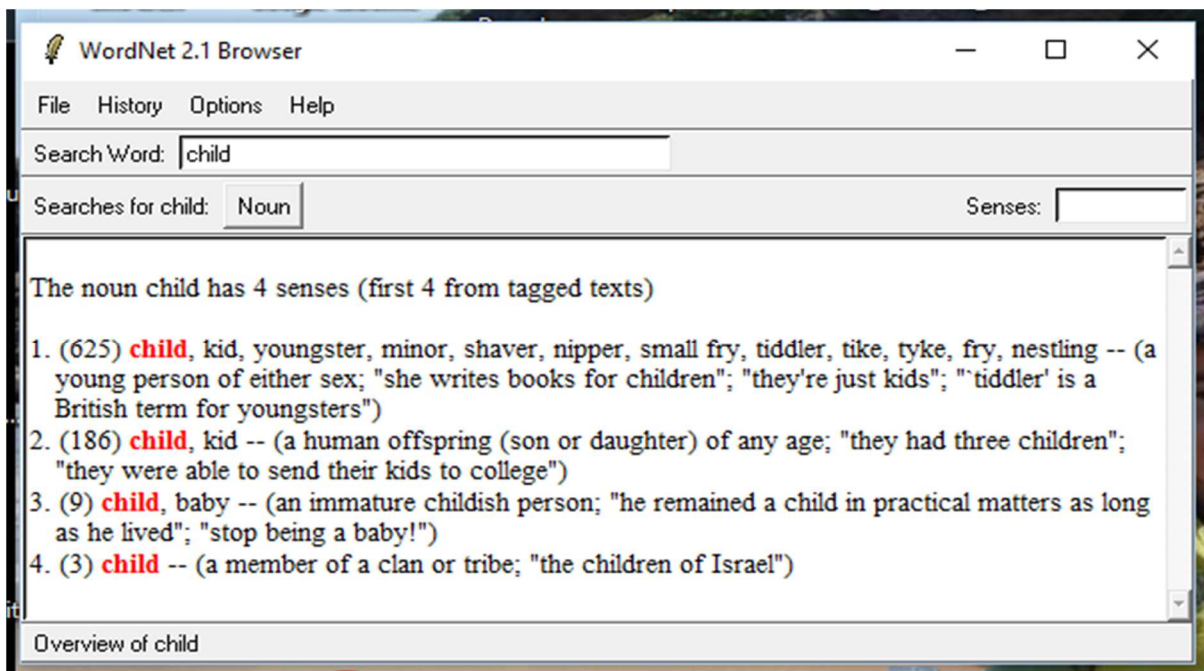
Figure II.6 : Les relations conceptuelles [13]

Maintenant dans la section suivante on va présenter le WordNet et ses relations :

II.5. WordNet :

WordNet [20] est une base de données lexicale développée depuis 1985 par des linguistes du laboratoire des sciences cognitives de l'université de Princeton. C'est un réseau sémantique de la langue anglaise, qui se fonde sur une théorie psychologique du langage. La première version diffusée remonte à juin 1991.

Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise. Le système se présente sous la forme d'une base de données électronique qu'on peut télécharger sur un système local. Des interfaces de programmation sont disponibles pour de nombreux langages [21].



FigureII.7 : Ensemble des différents sens du mot child dans WordNet**II.5.1 Structuration :**

Wordnet peut être vue comme un large graphe ou un réseau sémantique, où chaque nœud représente un concept du monde réel. Par exemple, un concept peut être un objet tel que house, une entité telle que teacher ou un concept abstrait tel que art.

Chaque nœud est constitué d'un ensemble de mots, où chacun représente le concept associé à ce nœud. Par la suite, chaque nœud peut être vu comme un ensemble de mots dont chacun représente le même concept. Par exemple le concept car est représenté par l'ensemble des mots {car, auto, automobile, motorcar}.

Dans la terminologie de WordNet, cet ensemble est connu sous le nom de synset. Les synsets sont des ensembles de synonymes qui regroupent des items lexicaux ayant des significations similaires [21].

II.5.1.1. Notion de synset :

Le *synset* (ensemble de synonymes) est la composante atomique sur laquelle repose WordNet. Un synset correspond à un groupe de mots interchangeable, dénotant un sens ou un usage particulier. Un synset est défini d'une façon différentielle par les relations qu'il entretient avec les sens voisins.

Les synsets sont reliés par des arcs qui décrivent les relations entre les différents concepts. Ils sont divisés en 4 catégories (noms, verbes, adjectifs et adverbes). La relation « is-a » est restreinte aux noms et verbes [21] .

La figure II.8 montre une petite portion de Wordnet sous forme de nœuds et arcs.

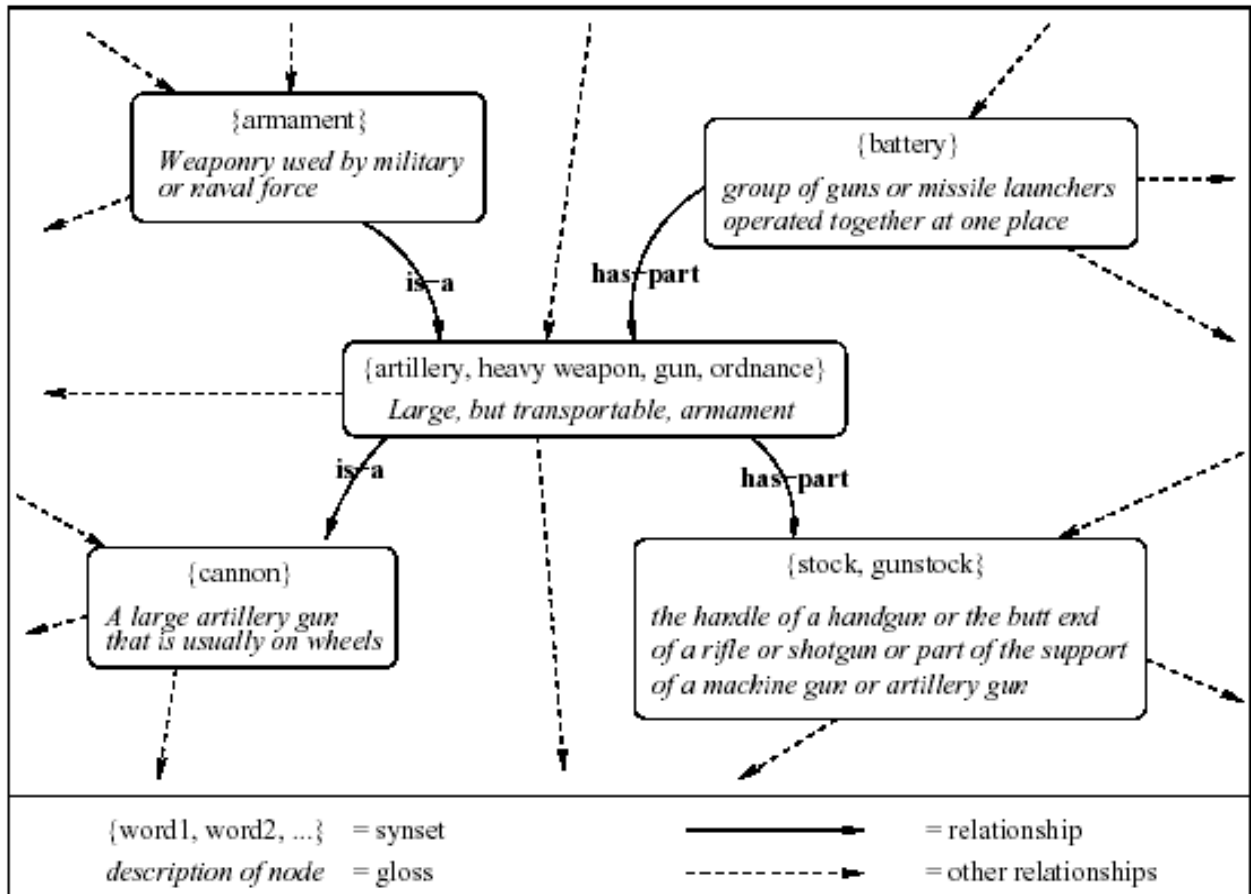


Figure II.8 : Une illustration des nœuds (synsets) et des liens (relations) d'une partie de graphe représentant WordNet

II.5.2. Les relations dans WordNet :

Les relations qui interviennent entre les wordforms sont appelées relations lexicales, en occurrence la synonymie, par contre celles qui relient les wordmeanings sont dites relation sémantiques telle que l'hyponymie. La plupart des relations dans WordNet concernent des synsets de la même catégorie, excepté les relations pertains to et attribute définies entre les adjectifs et noms.

Relation	Description	Example
Hypernym	is a generalization of	<i>furniture</i> is a hypernym of <i>chair</i>
Hyponym	is a kind of	<i>chair</i> is a hyponym of <i>furniture</i>
Troponym	is a way to	<i>amble</i> is a troponym of <i>walk</i>
Meronym	is part/substance/member of	<i>wheel</i> is a (part) meronym of a <i>bicycle</i>
Holonym	contains part	<i>bicycle</i> is a holonym of a <i>wheel</i>
Antonym	opposite of	<i>ascend</i> is an antonym of <i>descend</i>
Attribute	attribute of	<i>heavy</i> is an attribute of <i>weight</i>
Entailment	entails	<i>ploughing</i> entails <i>digging</i>
Cause	cause to	<i>to offend</i> causes <i>to resent</i>
Also see	related verb	<i>to lodge</i> is related to <i>reside</i>
Similar to	similar to	<i>dead</i> is similar to <i>assassinated</i>
Participle of	is participle of	<i>stored</i> (adj) is the participle of "to <i>store</i> "
Pertainym of	pertains to	<i>radial</i> pertains to <i>radius</i>

Tableau II.1 :Un ensemble de relations représentées dans WordNet

II.6. Conclusion :

Dans ce chapitre nous avons présenté et expliqué notre approche, on a fait une comparaison entre les informations pertinentes à partir du service et les informations extraites à l'aide de l'ontologie du domaine par le calcul de la mesure de similarité nous avons utilisé des approches qui se basent sur WordNet.

Cette mesure de similarité pouvant se baser sur divers aspects de la représentation des concepts, soit sur la longueur du chemin entre les concepts, le contenu d'information ou le type des relations entre les concepts. Dans notre approche nous avons utilisé mesure de similarité basée sur la longueur du chemin entre les concepts.

Implémentation

III.1. Introduction :

Ce chapitre nous permet de présenter les résultats de notre application dans sa phase de test et d'essai, ce qui nous permettra d'envisager les améliorations possibles. Les difficultés rencontrées nous ont permis de dégager un ensemble de pistes susceptibles de nous aider.

III.2. L'outil de développement :

L'outil de développement que nous avons utilisé c'est Eclipse. Eclipse est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la Fondation Eclipse visant à développer un environnement de production de logiciels libre qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java.

Son objectif est de produire et fournir des outils pour la réalisation de logiciels, englobant les activités de programmation (notamment environnement de développement intégré et frameworks) mais aussi d'AGL (Atelier de génie logiciel) recouvrant modélisation, conception, testing, gestion de configuration, reporting... Son EDI (Environnement de développement), partie intégrante du projet, vise notamment à supporter tout langage de programmation à l'instar de Microsoft Visual Studio.

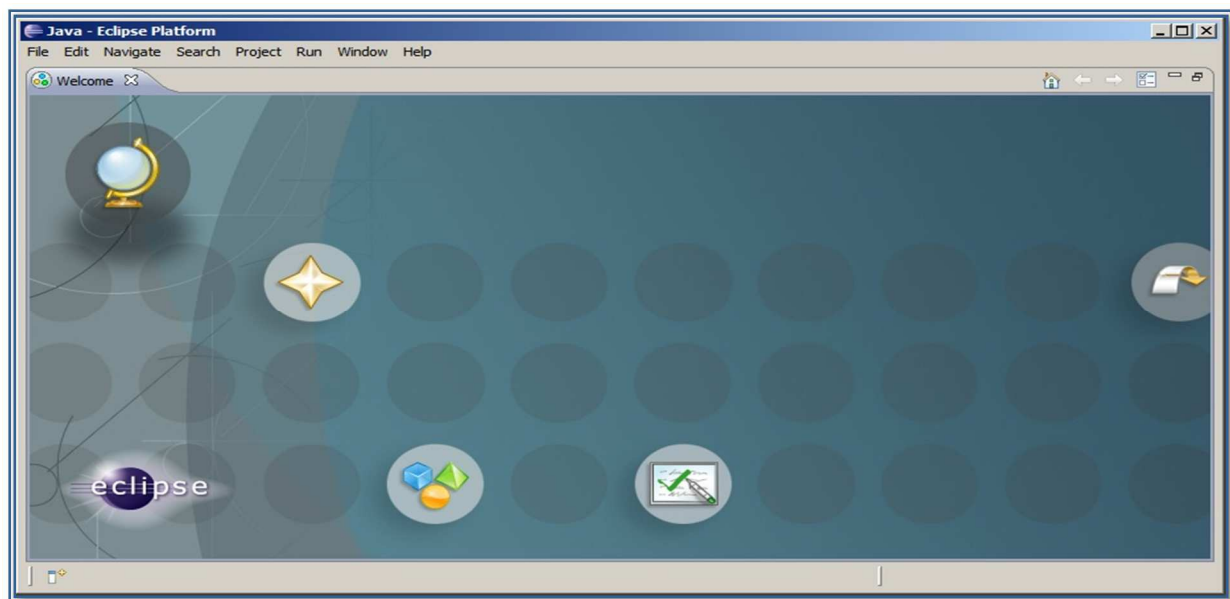


Figure III.1: Environnement d'éclipse.

III.3. Les résultats de notre approche :

III.3.1. Exemple d'illustration :

Afin de mettre en œuvre notre approche et d'en évaluer les performances, nous considérons un exemple illustratif tiré du web service de SWS proposé par l'outil de WSMO Studio.

Le code source de notre exemple du web service est présenté dans la Figure III.2.

```

wsm1Variant _"http://www.wsmo.org/wsm1/wsm1-syntax/wsm1-rule"
namespace { _"http://www.example.org/ontologies/example#",
            dc _"http://purl.org/dc/elements/1.1#",
            foaf _"http://xmlns.com/foaf/0.1/",
            wsm1 _"http://www.wsmo.org/wsm1/wsm1-syntax#",
            loc _"http://www.wsmo.org/ontologies/location#",
            oo _"http://example.org/ooMediator#" }

/*****
* WEBSERVICE
*****/
webService _"http://example.org/Germany/BirthRegistration"
  nfp
    dc#title hasValue "Birth registration service for Germany"
    dc#type hasValue _"http://www.wsmo.org/TR/d2/v1.2/#services"
    wsm1#version hasValue "$Revision: 1.1 $"
  endnfp
  importsOntology { _"http://www.example.org/ontologies/example",
                  _"http://www.wsmo.org/ontologies/location" }

  capability _"http://example.org/Germany/BirthRegistration#cap1"
    sharedVariables ?child

    precondition
      nonFunctionalProperties
        dc#description hasValue "The input has to be boy or a girl
        with birthdate in the past and be born in Germany."
      endNonFunctionalProperties
      definedBy
        ?child memberOf Child
        and ?child[hasBirthdate hasValue ?birthdate]
        and wsm1#dateLessThan(?birthdate, wsm1#currentTime())
        and ?child[hasBirthplace hasValue ?location]
        and ?location[locatedIn hasValue oo#de]
        or (?child[hasParent hasValue ?parent]
        and ?parent [hasCitizenship hasValue oo#de] ) .

    assumption
      nonFunctionalProperties
        dc#description hasValue "The child is not dead"
      endNonFunctionalProperties
      definedBy
        ?child memberOf Child
        and naf ?child[hasObit hasValue ?x].

    effect
      nonFunctionalProperties
        dc#description hasValue "After the registration the child
        is a German citizen"
      endNonFunctionalProperties
      definedBy
        ?child memberOf Child
        and ?child[hasCitizenship hasValue oo#de].

  interface _"http://example.org/Germany/BirthRegistration#iface1"
    choreography _"http://example.org/tobedone"
    orchestration _"http://example.org/tobedone"

```

Figure III.2: Extraits du code source du service web de SWS WSMO.

III.3.2. Résultat de la première phase :

Le filtrage du code du web service permet d'extraire tous les mots qui figurent après le mot-clé {" memberOf " } et avant le mot-clé {" ["}.

le filtrage de notre code donne lieu à trois thèmes remarquables qui forment l'ensemble {Child, location, parent}.



```
<terminated> TestBufferedReader [Java Application] C:\Program Files\Java\jre1.8.0_91\bin\javaw.exe (2 ma
Le mot qui précède '[' est : child
child
Le mot qui précède '[' est : child
child
Le mot qui précède '[' est : location
location
Le mot qui précède '[' est : Child
Child
Le mot qui précède '[' est : parent
parent
```

Figure III.3 : résultat du code de filtrage.

III.3.3. Résultat de la deuxième phase :

On fera l'analyse avec deux ontologies du domaine qui sont : education.owl, generations.owl. On fixe préalablement le seuil à 1.

- **Pour l'ontologie education.owl :** Les résultats de la phase d'analyse sont mentionnés dans le tableau (Tableau III.1).

Avant de calculer la similarité entre les informations retenues du services web et les concepts d'ontologies du domaine, on va déterminer les concepts d'ontologies qui appartiennent à l'application web, la figure suivante représente les concepts d'ontologies.

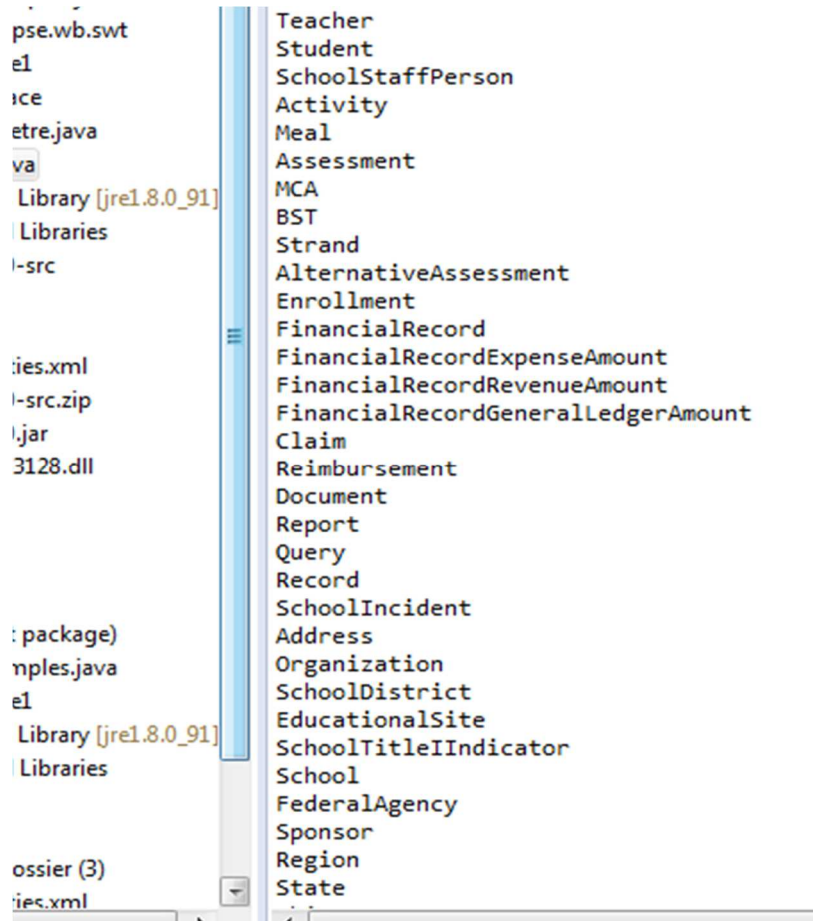


Figure III.4 : les concepts de l'ontologies education.owl.

Puis on va appliquer la mesure de similarité entre les concepts du service web et les concepts d'ontologie du domaine « **educationl.owl** »

Les concepts d'ontologie \ Les concepts du service web	Child	Location	Parent
Person	0.333333333	0.166666666	0.166666666
StaffPerson	-1	-1	-1
Teacher	0.142857142	0.1	0.1
Student	0.2	0.125	0.125
SchoolStaffPerson	-1	-1	-1
Activity	-1	-1	-1
Meal	0.090909909	0.166666666	0.07142857
Assessment	-1	-1	-1
MCA	-1	-1	-1
BST	-1	-1	-1
Strand	-1	-1	-1
AlternativeAssessment	-1	-1	-1
Enrollment	-1	-1	-1
FinancialRecord	-1	-1	-1
FinancialRecordExpenAmount	-1	-1	-1
FinancialRecordRevenuAmount	-1	-1	-1

FinancialRecordGnrlLedgerAmnt	-1	-1	-1
Claim	-1	-1	-1
Reimbursement	-1	-1	-1
Document	-1	-1	-1
Report	-1	-1	-1
Query	-1	-1	-1
Record	-1	-1	-1
SchoolIncident	-1	-1	-1
Address	-1	-1	-1
Organization	-1	-1	-1
SchoolDistrict	-1	-1	-1
EducatiounalSite	-1	-1	-1
SchoolTitleIndicator	-1	-1	-1
School	-1	-1	-1
FederalAgency	-1	-1	-1
Sponsor	0.16666666666666	0.11111111111111	0.11111111111111
Region	0.11111111111111	0.5	0.083333333333
State	-1	-1	-1
Thing	-1	-1	-1
Contact	-1	-1	-1

Tableau III.1 : résultat de la mesure de similarité entre web service et l’ontologie education.owl

La figure suivante présente le résultat de mesure de similarité :

```

-1.0
  filtrage---->Child
  onthologie---->Sponsor
  0.1666666666666666
  filtrage---->Child
  onthologie---->Region
  0.1111111111111111
  filtrage---->Child
  onthologie---->State
-1.0
  filtrage---->Child
  onthologie---->Thing
-1.0
  filtrage---->Child
  onthologie---->Contact
-1.0
  ce web service n'appartient pas au domaine de cette ontologie
  
```

Figure III.5 : résultat de mesure de similarité

Pour cette analyse aucun résultat de calcul de la mesure de similarité n’est égal à 1(seuil=1), donc on peut conclure que ce web service n’appartient pas au domaine de cette ontologie.

- **Pour l’ontologie generations.owl** : Les résultats de la phase d’analyse figurent dans le tableau (**Tableau III.2**). La figure **III.6** représente les concepts de l’ontologie generations.

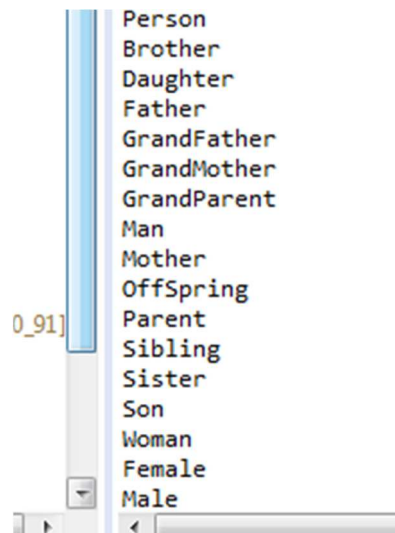



Figure III.6: les concepts de l'ontologies generations.owl

Les concepts d'ontologie \ Les concepts du service web	Child	Location	Parent
Person	0.33333333333333	0.16666666666666	0.16666666666666
Brother	-0.14285714285	0.1	0.125
Dauther	0.125	0.0909090909	0.11111111111111
Father	0.11111111111111	0.08333333333333	0.5
GrandFather	0.125	0.090909090909	0.14285714285
GrtandMother	0.125	0.090909090909	0.14285714285
GrandParnet	0.14285714285	0.1	0.166666666666
Man	0.2	0.125	0.125
Mother	0.11111111111111	0.08333333333333	0.5
OffSping	0.2	0.125	0.166666666666
Parent	0.125	0.090909090909	
Sibling	0.2	0.125	0.166666666666
Sister	0.14285714285	0.1	0.125
Son	0.125	0.090909090909	0.11111111111111
Woman	0.2	0.125	0.125
Female	0.16666666666666	0.14285714285	0.11111111111111

Mal	0.166666666666666	0.14285714285	0.1111111111111
Sex	-1	-1	-1

Tableau III.2 : résultat de la mesure de similarité entre le web service et l'ontologie generations.owl

Dans cette dernière analyse on trouve que le calcul de la mesure de similarité donne 1. La similarité (parent, parent) =1 : un cas évident. D'où la possibilité de conclure que ce service web sémantique appartient effectivement au domaine d'ontologie generations.owl.

III.4. Capture d'écran de l'application :



Figure III.7 : l'interface de notre application

La figure ci-dessus représente l'interface de notre application, en haut on trouve deux cases, la première pour choisir le web service et la deuxième pour choisir l'ontologie et en bas le bouton Analyser pour effectuer le calcul de la mesure de similarité.

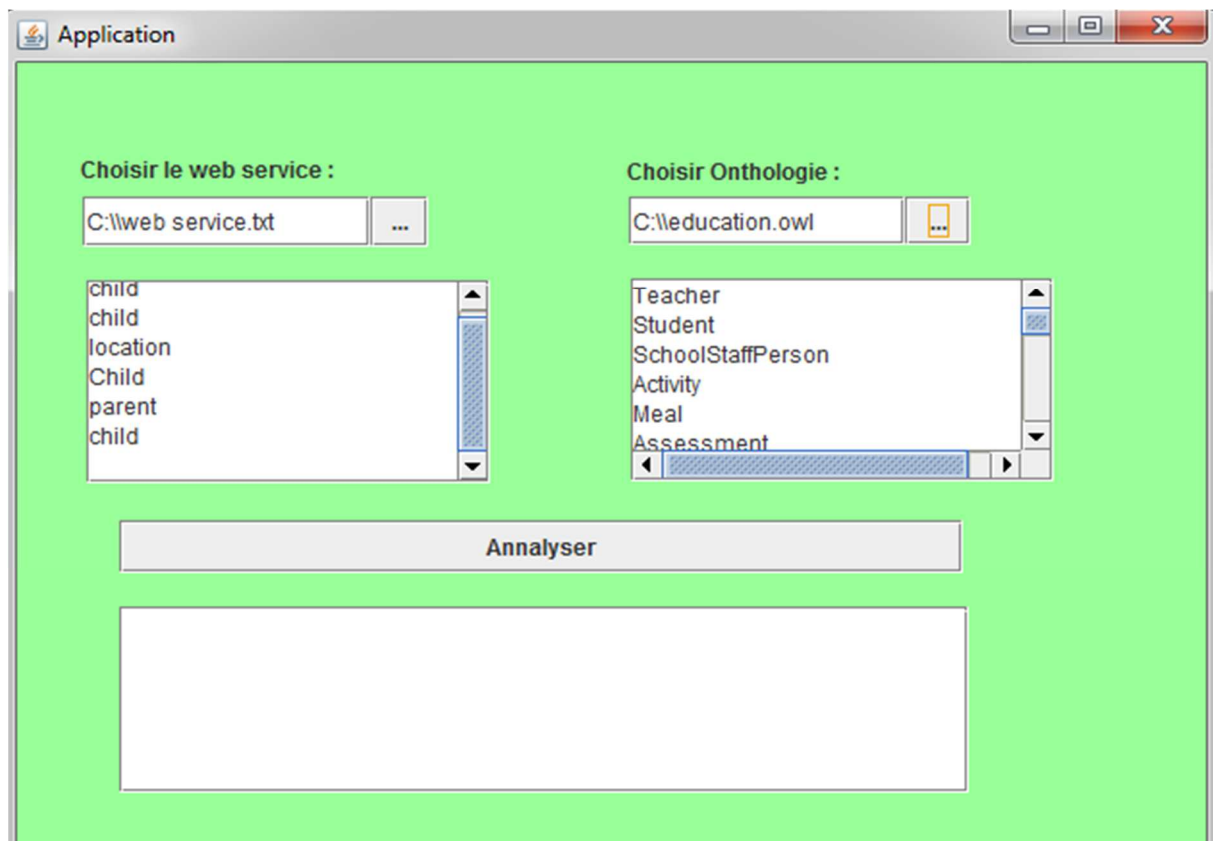


Figure III.8 : le choix de service web et l'ontologie

Une fois nous choisissons le web service et l'ontologie ,le programme fait le filtrage du web service et l'ontologie, le résultat du filtrage c'est des concepts qui sont affichés dans les deux cases. Après on clique sur le bouton analyser pour appliquer la mesure de similarité.

III.5. Conclusion :

Dans ce chapitre nous avons présenté les résultats de notre approche, on a effectué les calculs sur deux ontologies pour voir, Est ce que le web service appartient à l'un des ontologies ?

Après plusieurs expérimentations, L'objectif de ce travail est atteint, et les résultats initiaux obtenus sont encourageants et prometteurs. Surtout pour les services web qui présente moins de concepts.

CONCLUSION GÉNÉRALE

Les applications Web sont très répandues à travers le Web. Alors que ces applications s'intègrent de plus en plus dans des domaines diversifiés (commerce électronique, éducation, loisir), une nouvelle technologie dite les services Web sémantiques (SWS) a vu le jour. Cette technologie permet d'automatiser la découverte, la composition et l'invocation des services à travers le Web. Ainsi, la majorité des organisations tentent actuellement de faire migrer leurs applications Web courantes vers la nouvelle technologie des SWS. Plusieurs approches ont été développées pour les services Web sémantiques (OWL-S, WSMO, WSDL-S, etc.),

WSMO est l'un des frameworks les plus importants pour l'implémentation des services Web sémantiques. Il possède quatre éléments principaux : les *ontologies* qui fournissent la terminologie utilisée par les autres éléments WSMO ; les *services Web* qui fournissent un accès aux services ; les *buts* qui représentent les désirs de l'utilisateur ; et les *médiateurs* qui traitent le problème d'interopérabilité entre les différents éléments de WSMO.

Le but de notre travail est de proposer une approche pour accélérer la découverte des services sémantiques WSMO en utilisant les ontologies de domaines et la technique de mesure de similarité. Notre approche consiste en deux phases : la première est l'extraction d'informations pertinentes à partir de service web, l'élément de base étant un SWS WSMO, et la deuxième est l'analyse à l'aide de l'ontologie du domaine en faisant appel à une mesure de similarité. On utilise la mesure de similarité path (pattern matching) car est une meilleure moyenne pour mettre en œuvre l'intégration des concepts des services web et des ontologies.

Comme résultat on obtient le domaine de SWS WSMO et l'inférence de nouvelles ontologies importées.

Nous avons essayé de développer le processus d'accélération de la découverte de SWS et le rendre facile à utiliser. Néanmoins plusieurs travaux que nous n'avons pas pu accomplir et nous présentons comme perspectives s'ouvre à nous :

- Dans le processus d'accélération de la découverte de SWS, on utilise des mesures de similarité à base de WordNet. Cette base de données limite l'analyse à des applications Web écrites en anglais. Ce problème peut être résolu par l'utilisation de bases de données multilingues telles que EuroWordNet et ArabicWordNet.
- Dans les expérimentations discutées dans cette mémoire nous avons utilisé une seule mesure de similarité, à savoir la méthode *path*. D'autres expérimentations seront menées avec d'autres mesures de similarité pour prouver qu'on peut obtenir des résultats plus performants par notre processus d'accélération de la découverte de SWS.

Bibliographie :

- [1] bouchiha djelloul « Ré-ingénierie des Applications Web vers les Services Web sémantiques » thèse ;Sidi belabess ; Laboratory (EEDIS).12-01-2011.
- [2] Kadima H., Monfort V., « Les Web Services », mars 2003, page 1.
- [3] Burners-lee .t, HENDLER. j, LASSILA. o, «the semantic web», scientific american, vol. 284, no 5, 2001.
- [4] [http://home.cern/fr/topics/birth- web nvb 26](http://home.cern.fr/topics/birth- web nvb 26)
- [5] L'informaticien, [<http://www.linformaticien.com>], Consultation du novembre 2015.
- [6] :El bouhissi.H « Rétro-Ingénierie des Applications Web Services » ; Mémoire magister ; Sidi belabess ; 2010 N° 10
- [7] Chelbabi. M « Découverte de Services Web Sémantiques : une Approche basée sur le Contexte » mémoire magister ; 18-11-2006 ; N°9 .
- [8] Fensel D and Bussler C. «The Web Service Modeling Framework WSMF». *Electronic Commerce: Research and Applications*, 1(2): 113-137; Universiteit Amsterdam; (2002).
- [9] Roman D; Keller U; Lausen H; Lara R; Stollberg, M., Polleres A; Feier C; Bussler C and Fensel D. « Web Service Modeling Ontolog » *Applied Ontology journal*, Volume 1, Number 1, (2005), Pages: 77-106.
- [10] Wang, H-H., Gibbins, N., Payne, T., Saleh, A. and Sun, J. (2007). *A Formal Semantic Model of the Semantic Web Service Ontology (WSMO)*. In: Twelfth IEEE International Conference on Engineering of Complex Computer Systems, Auckland, New Zealand.
- [11] Bruijn, J., Lara, R., Polleres, A., and Fensel, D. (2005). *Owl dl vs. owl flight: conceptual modeling and reasoning for the semantic web*. In Proceedings of the 14th international conference on World Wide Web, Chiba, Japan, May 10-14, 2005. ACM Press, 623–632.
- [12]: Leacock, C., and Chodorow, M. (1998). *Combining local context and WordNet similarity for word sense identification*. In Fellbaum, C., ed., *WordNet: An electronic lexical database*. MIT Press. Pages: 265-283.
- [13] : Wu, Z., and Palmer, M. (1994). *Verb semantics and lexical selection*. In 32nd Annual Meeting of the Association for Computational Linguistics, Pages: 18.

[14] : Resnik, P. (1995). *Using information content to evaluate semantic similarity in a taxonomy*. In Proceedings of the 14th International Joint Conference on Artificial Intelligence. Pages: 448–453.

[15] : Lin, D. (1998). *An information-theoretic definition of similarity*. In Proceedings of the International Conference on Machine Learning.

[16] : Jiang, J., and Conrath, D. (1997). *Semantic similarity based on corpus statistics and lexical taxonomy*. In Proceedings on International Conference on Research in Computational Linguistics, Pages: 19-33.

[17] : Hirst, G., and St-Onge, D. (1998). *Lexical chains as representations of context for the detection and correction of malapropisms*. In Fellbaum, C., ed., *WordNet: An electronic lexical database*. MIT Press. Pages: 305–332.

[18] : Banerjee, S., and Pedersen, T. (2003). *Extended gloss overlaps as a measure of semantic relatedness*. In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Pages 805–810.

[19] Patwardhan, S. 2003. Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. Master's, Univ. of Minnesota, Duluth

[20] Miller G. 1995. Wordnet: A lexical database. *Actes de ACM* 38, pp. 39-41.

[21] François-Régis Chaumartin. (2007). *WordNet et son écosystème : un ensemble de ressources linguistiques de large couverture* ; Université de Paris, France, page 2.