

UNIVERSITE KASDI MERBAH OUARGLA

*FACULTE DES SCIENCES ET SCIENCES DE
L'INGENIEUR*

DEPARTEMENT DES MATHEMATIQUES ET D'INFORMATIQUE



N° d'ordre :

N° de série :

MEMOIRE

En vue de l'obtention du diplôme de

MASTER ACADEMIQUE

Domaine : Mathématique et Informatique

Filière : Informatique

Spécialité : Informatique Fondamentale

Présenté par:

Moussaoui Kamal

Feredj Dhia Elhak

Thème

Conception et développement d'un Outil de recherche sur le web

à base d'agent

Soutenu publiquement

le : 01 / 07 / 2013

Devant le jury composé de :

M	HARROUZ ABDELHAKIM	MA	Président	UKM Ouargla
Mme	KHELILI KHALIDA FARIDA	MA	Rapporteur	UKM Ouargla
M	BELMIR ABDELKADER	MA	Examineur	UKM Ouargla

Année Universitaire : 2012 /2013

REMERCIEMENTS

Comment ne pas remercier الخالق, qui nous a donné la santé, la volonté et le courage de faire ce travail.

Nous tenons tout d'abord à remercier notre promoteur, Mme Khelili K-F, à qui nous doit le sujet, pour sa disponibilité, et pour ses précieux conseils tout au long de ce travail.

Nous tenons également à remercier les membres du jury qui ont bien voulu accepter d'examiner ce travail.

Nous adressons un remerciement particulier à M. Khaldi Bilal, pour son aide, gentillesse et son soutien .

Bien entendu, Nous tenons surtout à remercier nos parents pour leurs sacrifices et leur patience, tout au long de nos vies.

Nous remercions toutes les personnes qui ont contribué de près ou de loin dans l'accomplissement de ce travail.

Et à tous ceux que nous n'avons pas cités, nous remercions toutes nos familles et nos amis.

Merci pour tout le monde

Kamal et Dhia



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 МАМА.*

*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 préservez et
vous accordez santé, longue vie et bonheur.*

À mes chers frères et sœurs,

*À mes oncles, tantes, cousins et cousines
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 les personnes qui m'ont connu, soutenu et aimé.

À ma Moussiba

À АSМА

À ma mère, À ma mère, À ma mère, ...

Je dédie ce modeste travail.

КАМАЛ

Dédicace

*À mes très chers parents adorés qui m'ont fait bénéficiaire, grâce
à leur invocation et à leur bénédiction, à avoir la confiance et
l'espoir au cours de la rédaction de ce travail À mes chers
frères et sœurs,
À mes oncles, tantes, cousins et cousines
À toute ma famille,
À toute personne qui a participé de près ou de loin pour
l'accomplissement de ce modeste travail.*

*À tous mes amis et anciens collègues avec qui j'ai partagé les
plus beaux et inoubliables moments de ma vie,
À toutes les personnes qui m'ont connu, soutenu et aimé.
Je dédie ce modeste travail.*

*Enfin, pour finir et pour être sûre de n'oublier personne, nous
dédions tout le monde.*

À ma mère.. À mon père

DHIA ELHAK

Résumé

Aujourd'hui, le web est la plus large ressource le plus utilisé pour la recherche, et aussi la publication des informations, mais d'autre part, l'augmentation des données accessibles via le réseau pose des problèmes tels que la saturation du réseau et donc trouvé de l'information pertinente dans le web est devenue une tâche qui n'est pas facile, c'est pour cette raison que des millions d'internautes ne parviennent pas à satisfaire la totalité de leurs besoins en information facilement et rapidement.

Donc, il est indépassable de trouver des nouvelles solutions afin d'améliorer la recherche d'information dans le Web, les moteurs de recherche intelligents apparaissent dans ce contexte comme une solution prometteuse permettant la limitation de l'impact de ces problèmes.

Dans ce travail, on essayera de proposer une architecture à base d'agents d'un moteur de recherche d'information sur le Web, avec l'utilisation d'une ontologie qui nous servira à avoir de la sémantique dans notre outil de recherche.

Mots clés :

Outil de recherche d'information, recherche d'information, Web sémantique, Agent, ontologie.

ملخص

اليوم، شبكة الإنترنت هي المورد الأكبر و الأكثر استخداما للعثور على المعلومات، وكذلك نشر المعلومات، ولكن من ناحية أخرى، فإن ازدياد البيانات ينطوي على مشاكل مثل ازدحام شبكة، وكذلك البحث في الشبكة يصبح بطيئا جدا. وأن الملايين من متصفحى الإنترنت تفشل في تلبية جميع احتياجاتهم من المعلومات.

ولذلك، فمن الضروري إيجاد حلول جديدة لمعالجة المعلومات. محركات البحث الذكية تعتبر في هذا السياق حلا واعدة للحد من آثار هذه المشاكل.

في هذا العمل، نقترح هندسة تعتمد على وكلاء مع استخدام الأنطولوجيا كأساس لتطوير محرك البحث الخاص بنا.

كلمات مفتاحية :

أدوات البحث عن المعلومات، البحث عن المعلومات، ويب دلالي، وكيل، أنطولوجيا.

Abstract

Today, the web is the largest and most used resource for finding information, and also the publication of information, but on the other hand, the increase in accessible data networks involves problems such as congestion network and also research in the web become very slowly, and millions of the net users surfers fail to satisfy their full informational needs.

Therefore, new solutions become available to manipulate the information from the resource. the intelligent search engines appear in this context as a promising solution to limiting the impact of these problems.

In this work, we propose an architecture based on agents with the use of ontology as a basis for developing our search engine.

Keywords :

Search information tools, Information Retrieval, Semantic Web, Agent, ontology.

Table des matières

Liste des Figures

Liste des Tableaux

Introduction générale

I.	CENTRES D'INTERET	1
II.	CONTEXTE ET PROBLEMATIQUE	2
III.	ORGANISATION DE MEMOIRE	3
	<u>Chapitre I : LES OUTILS DE RECHERCHE SUR LE WEB :</u>	
1.	INTRODUCTION	6
2.	OUTILS DE RECHERCHE SUR LE WEB	7
2.1.	Les annuaires ou répertoires thématiques	7
2.1.1.	Qu'est-ce qu'un annuaire ?	7
2.1.2.	Avantages attribués aux annuaires	9
2.1.3.	Inconvénients attribués aux annuaires	9
2.2.	Les moteurs de recherche	10
2.2.1.	Qu'est-ce qu'un moteur de recherche ?	10
2.2.2.	Principes de base	11
2.2.3.	Structure d'un moteur de recherche	12
2.2.4.	Fonctionnement d'un moteur de recherche	14
2.2.5.	Le Robot (Spider)	14
2.2.5.1.	Composition du robot	15
2.2.5.2.	Fonctionnement générale du robot	15
2.2.6.	Architectures des moteurs de recherche	16
2.2.6.1.	Architecture générale des premiers moteurs de recherche	16
2.2.6.2.	Architecture distribuée et adaptative	16
2.2.6.3.	Architecture moderne d'un moteur de recherche	17
2.2.7.	Domaines d'utilisations des moteurs de recherche	18
2.2.8.	Les restrictions d'accès à l'information	19
2.2.9.	Avantages attribués aux moteurs de recherche	19
2.2.10.	Inconvénients attribués aux moteurs de recherche	19
2.3.	Les méta-moteurs	20
2.3.1.	Qu'est-ce qu'un méta-moteur	20
2.3.2.	Principes de base	20
2.3.3.	Quelques méta-moteurs	21
2.3.4.	Avantages attribués aux méta-moteurs	22
2.3.5.	Inconvénients attribués aux méta-moteurs	22

2.4	L'indexation	22
2.4.1.	Analyse lexicale	23
2.4.2.	Elimination des mots vides	23
2.4.3..	Lemmatisation	23
2.4.4.	Pondération	24
2.4.5.	Création de l'index	24
2.4.6.	Les langages d'indexation	24
3.	MOTEURS DE RECHERCHE INTELLIGENTS	25
3.1.	Exemple d'un moteur de recherche intelligent	26
3.1.	WolframAlpha	26
4.	COMPARAISON ENTRE LES OUTILS DE RECHERCHE	27
5.	CONCLUSION	29
Chapitre II: LES AGENTS :		
1.	INTRODUCTION	31
2.	GENERALITE SUR LES AGENTS	31
2.1.	Définitions	31
2.2.	Caractéristiques des agents	33
2.3.	Typologie des agents	33
2.3.1.	Agents réactifs	34
2.3.2.	Agents cognitifs (délibératifs)	34
2.4.	L'interaction entre les agents	34
2.4.1.	La Coopération	35
2.4.2.	La Collaboration	35
2.4.3.	La Coordination	35
2.4.4.	La Négociation	36
2.4.5.	La Communication	36
3.	LES SYSTEMES MULTI-AGENTS (SMA)	38
4.	LES AGENTS INTELLIGENTS	39
4.1.	Définition	39
4.2.	Les agents et le web	40
4.3.	Les caractéristiques des agents intelligents	41
4.4.	Les Caractéristiques additionnels	43
4.5.	Classification des agents intelligents	44
4.5.1.	Classification selon la plate-forme d'informations	44
4.5.2.	Classification PUSH/PULL	44
4.5.3.	Classification Client / Serveur	45

4.5.4.	Classification par Tâches.....	45
4.6.	Le fonctionnement d' un Agent intelligent.....	47
4.7.	Quelques agents intelligents sur le web	48
4.7.1.	WebSeeker.....	48
4.7.2.	Quelques Fonctions.....	48
5.	CONCLUSION.....	49
Chapitre III : LES ONTOLOGIE DANS LES MOTEURS DE RECHERCHE :		
1.	INTRODUCTION.....	52
2.	L'ONTOLOGIE.....	53
2.1.	Notion d'ontologie.....	53
2.2.	Définitions.....	53
2.3.	L'objectif d'ontologie	54
2.4.	Que représente-t-on dans une ontologie ?.....	54
2.4.1.	Concepts	55
2.4.4.	Axiomes.....	56
2.4.5.	Instances.....	56
2.5.	Typologie des ontologies.....	56
2.5.1.	Les ontologies de représentation.....	56
2.5.2.	Les ontologies génériques	56
2.5.3.	Les ontologies de domaine	57
2.5.4.	Les ontologies de tâches	57
2.5.5.	Les ontologies d'application.....	57
2.6.	Utilisation des ontologies.....	57
2.6.1.	La connaissance du domaine.....	57
2.6.2.	La communication.....	57
2.6.3.	L'interopérabilité.....	57
2.6.4.	L'aide à la spécification des systèmes.....	58
2.6.5.	L'indexation et la recherche d'information.....	58
2.7.	Construction d'une ontologie	58
2.8.	Quelques méthodologies de construction d'ontologie.....	59
2.9.	Cycle de vie d'une ontologie.....	59
3.	WEB SEMANTIQUE.....	60
3.1.	Principe général du web sémantique.....	61
3.2.	Langages du Web sémantique.....	62
4.	LES ONTOLOGIES ET LE WEB SEMANTIQUE.....	64
4.1.	La recherche d'information guidée par les ontologies dans le web sémantique.....	64

4.2.	Exemple de l'utilisation d'ontologie dans la recherche d'information	66
4.2.1.	Google, vers un moteur de recherche sémantique.....	66
5.	CONCLUSION.....	67
Chapitre IV : MODELISATION :		
1.	INTRODUCTION.....	70
2.	LANGUAGE UML.....	71
2.1.	Définition UML	71
2.2.	Pourquoi UML ?	71
2.3.	Les diagrammes UML	71
2.3.1.	Diagrammes structurels ou diagrammes statiques.....	71
2.3.2.	Diagrammes comportementaux ou diagrammes dynamiques.....	72
2.3.3.	Les diagrammes en détails	72
2.3.3.1.	Diagramme de cas d'utilisation.....	72
2.3.3.2.	Diagramme de classes	72
2.3.3.3.	Diagramme d'objets.....	72
2.3.3.4.	Diagramme d'états-transitions.....	73
2.3.3.5.	Diagramme d'activités	73
2.3.3.6.	Diagramme de séquence et de communication.....	73
3.	L'IMPLEMENTATION DE NOTRE SYSTEME	73
3.1.	Diagramme de cas utilisation	73
3.2.	Diagramme de séquence	74
3.2.1.	Diagramme de séquence (Initialisation de Crawler).....	74
3.2.2.	Diagramme de séquence(Crawler)	75
3.2.3.	Diagramme de séquence(Réinitialisation de Crawler).....	75
3.2.4.	Diagramme de séquence(Système).....	76
3.3.	Diagramme de collaboration :	77
3.3.1.	Diagramme de collaboration (Crawler):	77
3.3.2.	Diagramme de collaboration (Model de recherche):.....	77
3.4.	Diagramme d'activité.....	78
3.5.	Diagramme de Classes (Système).....	79
4.	Conclusion.....	79
Chapitre V : IMPLEMENTATION :		
1.	INTRODUCTION.....	81
2.	ENVIRONNEMENT DU TRAVAIL	81
2.1.	Choix du langage de programmation	81
2.1.1.	Java.....	81

2.1.2.	Java Script.....	81
2.2.	Choix de la plate-forme	82
2.2.1.	Jade 4.3.0.....	82
2.3.	Choix des éditeurs	83
2.3.1.	NetBeans IDE 7.2.1	83
2.3.2.	Protégé 3.4.4.....	84
2.4.	Choix de Serveur	85
2.4.1.	Apache Tomcat Server 7.0.40.....	85
2.5.	Choix des outils et des technologies supplémentaires	85
2.5.1.	SPARQL.....	85
2.5.2.	SKOS.....	86
2.5.3.	Apache Lucene™ 4.3.....	86
2.5.4.	TIKA™ 1.3.....	86
2.5.5.	Crawler4J.....	86
3.	ARCHITECTURE GENERALE DU SYSTEME	87
3.1.	Le <i>Crawling</i>	87
3.2.	La recherche.....	87
3.3.	Les résultats.....	87
3.4.	Explication de l'architecture	89
4.	IMPLEMENTATION DES COMPOSANTS DU SYSTEME.....	89
4.1.	Coté administrateur	90
4.2.	Coté utilisateur	92
5.	L'INTELLIGENCE DE NOTRE OUTIL.....	93
6.	CONCLUSION.....	96
	Conclusion générale.....	98
	Bibliographie.....	100

Liste des Figures

Figure I-1 : *Les annuaires ou répertoires thématiques.*

Figure I-2 : *Moteur de recherche.*

Figure I-3 : *Architecture générale d'un moteur de recherche.*

Figure I-4 : *Fonctionnement d'un moteur de recherche.*

Figure I-5 : *Fonctionnement générale du robot.*

Figure I-6 : *Architecture originale du moteur de recherche **Altavista**.*

Figure I-7 : *Architecture du système **Harvest**.*

Figure I-8 : *Architecture du moteur de recherche **Google**.*

Figure I-9 : *Méta-moteur de recherche.*

Figure II-1. *L'agent dans son environnement*

Figure II-2 : *différents types d'interactions entre les agents.*

Figure II-3 : *La communication par tableau noir.*

Figure II-4 : *La communication par envoi de message.*

Figure II-5 : *Schéma de la représentation de l'intelligence pour un Agent.*

Figure II-6 : *Schématisation du mécanisme de l'intelligence chez un agent.*

Figure II-7 : *Capture d'écran de l'interface du webseeker.*

Figure III-1: *Historique de L'ontologie (informatique).*

Figure III-2 *Quelque utilisation des ontologies.*

Figure III-3 : *Les types de langage OWL*

Figure III-4 : *Google Sur son chemin vers sémantique.*

Figure IV -1: *Diagramme de cas utilisation de notre système.*

Figure IV -2: *Diagramme de séquence « Initialisation de Crawler».*

Figure IV -3: *Diagramme de séquence « Crawler».*

Figure IV -4 : *Diagramme de séquence « Réinitialialisationde Crawler».*

Figure IV-5 : *Diagramme de séquence « Système».*

Figure IV -6 : *Diagramme de collaboration « Crawler».*

Figure IV -7 : *Diagramme de collaboration « Model de recherche».*

Figure IV -8 : *Diagramme d'activité « Model de recherche».*

- Figure IV-9 :** *Diagramme de classes « Système ».*
- Figure V-1 :** *interface de la plateforme Jade 4.3.0.*
- Figure V-2 :** *l'interface de l'IDE Netbeans 7.1.2.*
- Figure V-3 :** *l'interface de Protégé 3.4.4.*
- Figure V-4 :** *l'interface Apache Tomcat Server 7.0.40*
- Figure V-5 :** *L'architecture générale de notre application " iNeed ".*
- Figure V-6:***La tâche de crawling en cours d'exécution.*
- Figure V-7 :***Les résultats de Crawling.*
- Figure V-8 :***le contenu de fichier indexé.*
- Figure V-9 :***La tâche de crawling en cours d'exécution.*
- Figure V-10 :***L'interface utilisateur principale d'iNeed.*
- Figure V-11 :***L'interface utilisateur du résultat d'iNeed.*
- Figure V-12 :***L'interface A propos d'iNeed.*
- Figure V-13 :***résultat d'une recherche.*
- Figure V-14 :***la recherche sémantique*
- Figure V-15 :***le rédacteur des propositions*
- Figure V-16 :***la correcteur automatique de iNeed*
- Figure V-17 :***la recherche par type.*
- Figure V-18 :***lancement d'un téléchargement.*

Liste des Tableaux

- Tableau II-1.***Comparaison entre les deux agents.*
- Tableau IV-1.***Les cas d'utilisation du l'utilisateur et du l'administrateur.*

Introduction Générale

Introduction générale

I. CENTRES D'INTERET

Le volume de données présentes sur le web est en augmentation extrêmement rapide, et en faisant une moyenne des estimations trouvées dans des articles de presse ou des articles scientifiques, on peut dire que le nombre de documents accessibles est de l'ordre de la dizaine de milliards. Ces documents peuvent être de nature scientifique, industrielle ou commerciale. Pour des entreprises mais aussi pour des particuliers, ces documents peuvent contenir à un moment donné des informations stratégiques nécessaires à la résolution d'un problème. Cette taille colossale et la demande des utilisateurs posent un défi à la communauté scientifique qui doit être en mesure de proposer des outils efficaces de recherche d'information.

Des moteurs de recherche sont donc apparus depuis des dizaines d'années. À partir d'une requête simple, ils parcourent leur mémoire pour trouver le plus rapidement possible les documents correspondant aux souhaits de l'utilisateur. Google est certainement la dernière réussite américaine en ce domaine puisqu'il possède en mémoire l'image de plusieurs milliards de documents. Il fournit souvent en réponse à une requête des milliers de documents en un temps inférieur à la seconde. Ce sont là précisément les points forts et les points faibles de ce type de méthode : l'utilisateur est submergé par des milliers de réponses présentées "en vrac", dont une partie ne correspond pas à la requête, une autre correspond à des pages qui n'existent plus, sans parler de nombreux types de documents qui ne sont pas pris en compte dans la mémoire du moteur. Dans bien des cas, l'utilisateur passe un temps assez long à analyser les résultats du moteur, sans garantie de résultats.

II. CONTEXTE ET PROBLEMATIQUE

Il y a plusieurs façons d'accéder à l'information sur internet, il y a aussi plusieurs façons de perdre son temps à effectuer cette opération. La méthode la plus commune pour chercher de l'information est d'utiliser un moteur de recherche tels que **Google** (<http://www.google.com>) ou **Altavista** (<http://www.altavista.com>). Par contre, ces moteurs n'utilisent pas le sens des textes continus dans les pages trouvées, mais plutôt l'occurrence des mots dans ces pages ainsi que les liens vers ces pages. Quoique relativement efficaces, ces moteurs s'ils ne sont pas utilisés convenablement, peuvent noyer l'internaute dans des millions des pages qui contiennent les dits mots de la recherche.

C'est notamment pour corriger ces problèmes que le web sémantique mis de l'avant. Les moteurs de recherche intelligents appartiennent à la cadre de web sémantique permis entre autres de spécifier le sens des mots. Une recherche pourra alors être située dans un contexte sémantique avant d'être lancée, afin d'apporter des réponses plus précises aux requêtes demandées. Présentement, les pages qui incorporent de telles informations sont plutôt rares.

De plus, pour utiliser automatiquement cette information, un agent devra être programmé en conséquence, car à moins qu'il puisse comprendre le texte en langage naturel, il y a peu de chance qu'il déduise lui-même le sens d'une propriété. Un agent dans ce contexte est un programme qui perçoit et agit sur son environnement et il sera décrit plus en détail dans l'état de l'art.

Les ontologies aussi forment un modèle utilisé pour la représentation sémantique des services Web. Elles permettent d'établir des relations sémantiques entre les différents concepts d'un domaine et d'étendre la description sémantique des services web.

Alors, notre objectif de ce mémoire c'est de créer un moteur de recherche intelligent à la base de ces deux techniques : les agents intelligents et l'ontologie, qui va résoudre la difficulté de trouver les besoins exacte de les internautes, en partant du principe très général suivant : plutôt que de donner une réponse très rapide et très pauvre en information, il serait envisageable pour de nombreuses requêtes de fournir des résultats beaucoup plus riches pour l'utilisateur mais en un temps plus long. ce nouveau moteur va pouvoir explorer de nouvelles régions d'Internet avec une stratégie efficace sélectionnant les meilleures pages à explorer à un moment donné.

III. ORGANISATION DE MEMOIRE

L'ensemble de chapitres composant ce thème sont organisés en deux grandes parties : La première partie est un état de l'art présentant respectivement: les outils de recherches sur le web et les agents intelligents dans les moteurs de recherche et enfin, les ontologie pour la recherche sur le web. Dans la deuxième partie, nous présentons en détail nos travail pratique, on commençons par la modélisation du travail et on terminons par l'implémentation et les résultats.

Plus précisément les chapitres de ce mémoire se présentent comme suit : Dans la première partie nous avons abordé l'état de l'art, cette partie comprend trois chapitres

Le premier chapitre, présente les outils de recherche sur le web, nous présentons la définition, architecture, exemples, avantages et inconvénients de chaque outil.

Dans le deuxième chapitre, nous présentons les agents et les agents intelligents en informatique. Nous citerons les notions de base, les caractéristiques, l'environnement, l'interaction entre les agents et en fin, nous discutons des plates-formes d'agents intelligents existantes de nos jours.

Dans le troisième chapitre, nous présentons les ontologies, leurs utilisations en informatique, ainsi que leurs utilisations dans les moteurs de recherche.

Dans la deuxième partie nous avons décrit notre réalisation pratique, cette partie comprend deux chapitres, le quatrième et le cinquième chapitre.

Dans le quatrième chapitre, nous décrivait notre contribution, donc c'est la proposition d'une approche basée sur l'utilisation de la technologie agents et ontologies pour la recherche d'information dans le web.

Dans le cinquième chapitre, on discute la réalisation et l'implémentation de l'architecture décrite dans le quatrième chapitre. Toute on décrivait les outils, langages et technologies utilisés lors du processus d'implémentation.

Enfin, nous terminons ce mémoire par une conclusion générale, qui récapitule les travaux réalisés et fait le point sur un ensemble de perspectives envisagé

Première partie :

Etat de l'art

Chapitre I

LES OUTILS DE RECHERCHE

SUR LE WEB

Dans ce chapitre :

- ❖ Introduction
- ❖ Outils de recherche sur le web
- ❖ Moteurs de recherche intelligents
- ❖ Comparaison entre les outils de recherche
- ❖ Conclusion

1. INTRODUCTION

Lorsqu'un nouveau protocole naît sur l'Internet et commence à être abondamment exploité, des outils de recherche appropriés voient le jour. Le web a connu en ces dernières années un développement inouï.

Selon les statistiques de NetCraft (Entreprise Anglaise spécialisée dans les technologies Internet), le web compte plus de 631.521.198 sites, en Mars 2013. **(site1)**

Le nombre actuel de pages web est estimé au moins 14,6 billion, auxquelles viennent se rajouter quotidiennement milliards de nouvelles pages. **(site2)**

Cette quantité d'informations en tout genre rend parfois difficile la navigation et la recherche rapide. C'est pourquoi il est impératif d'introduire la notion d'outils de recherche permettant de lancer une requête et de trouver ainsi les documents susceptibles de renfermer l'information désirée par l'utilisateur. La recherche d'information par l'intermédiaire de ces outils spécifiques permet un gain de temps considérable.

Nous distinguons trois catégories d'outils pour la recherche d'information sur le web : les moteurs de recherche, les annuaires et les méta-moteurs. Cette distinction qui repose également sur le mode d'indexation reste essentielle, car elle induit des usages et des technologies très différentes. Ainsi un annuaire thématique va-t-il référencer des sites Web, là où un moteur indexera toutes les pages d'un site ? En effet, l'annuaire facilitera le défrichage, le premier repérage des ressources dans un domaine ou un secteur défini par l'organisation arborescente proposée, alors qu'un moteur de recherche permettra de trouver un document très précis. Enfin les méta-moteurs permettent d'interroger en une seule fois différents outils de recherche, qu'ils soient de type annuaire ou de type moteur. Nous présentons dans ce qui suit ces trois catégories d'outils pour la recherche sur le web. (BOURAMOUL, 2011)

2. OUTILS DE RECHERCHE SUR LE WEB :

Sur le web, Il existe à l'heure actuelle trois grandes familles d'outils de recherche :

- Les annuaires ou répertoires thématiques.
- Les moteurs de recherche.
- Les méta-moteurs.

2.1. Les annuaires ou répertoires thématiques :

2.1.1. Qu'est-ce qu'un annuaire ?

Un annuaire ou répertoire est un outil de recherche dont les sites proposés sont vérifiés par des humains et classés selon des catégories appropriées. L'inscription d'un site dans un annuaire pour figurer dans la base de données se fait au moyen d'un formulaire où doivent être inscrites plusieurs informations dont une description du site, l'adresse (URL), le titre ainsi que la catégorie dans laquelle le propriétaire du site désire qu'il figure.

Les éléments requis lors d'une demande d'inscription sont très importants car la recherche au moyen de mots clés est basée sur ces éléments (adresse Web, titre, description) et non sur le contenu des pages du site en question. Il ne s'agit donc pas d'une indexation automatique effectuée par un Robot (programme informatique qui scrute le réseau afin de localiser les documents et les indexer dans une base de données), mais d'un référencement humain et volontaire sollicité par le titulaire du site lui-même (R.ZAIANE, 1999)

Les principaux répertoires qui existent sur Internet sont : **La Toile du Québec, Yahoo, Open Directory Project, Looksmart, Galaxy, NetGuide.**

Les annuaires sont donc des outils basés sur le recensement humain de l'information. Ils signalent des sites et des ressources de l'Internet comme un catalogue de bibliothèque signale des livres ou bien encore comme les pages jaunes signalent des entreprises. On distingue dans ce contexte deux catégories d'annuaires On distingue dans ce contexte deux catégories d'annuaires :

A) *Les annuaires commerciaux*

Ils se financent grâce à la publicité. Ils ont en principe une couverture dit "générale" (ils couvrent toutes les disciplines). Ils peuvent concerner le monde ou une zone régionale, nous citons parmi eux :

- Annuaires généralistes internationaux : le plus connu est sans doute 'Yahoo Directory', mais il existe aussi 'DMIZ' de l'Open Directory Project et l'annuaire de 'Lycos'.
- Annuaires régionaux commerciaux : ce sont les annuaires qui recensent des sites en fonction de leur langue. Dans le cas de des annuaires francophones nous citons la version française de 'Yahoo Directory' ou encore l'annuaire 'Francité'.
- Les annuaires qui recensent d'autre pays ou parties du monde : comme l'annuaire 'Wohaa' pour l'Afrique et l'annuaire russe 'Yandex'.

B) *Les annuaires non commerciaux*

Sont des annuaires élaborés par des individus de façon bénévole ou bien par des institutions. Ils sont soit généraux soit spécialisés. Leur préoccupation consiste toujours à identifier les ressources et les sites en tenant comptes de leur qualité :

- Annuaires à couverture (généraliste) : comme le 'Vlib' (Virtual Library) et l'annuaire 'Resource Discovery network'.
- Annuaires à couverture thématique ou spécialisée : comme le répertoire en sciences humaines 'Voice of the Shuttle' et le répertoire de ressources juridiques 'Findlaw'. (BOURAMOUL, 2011)

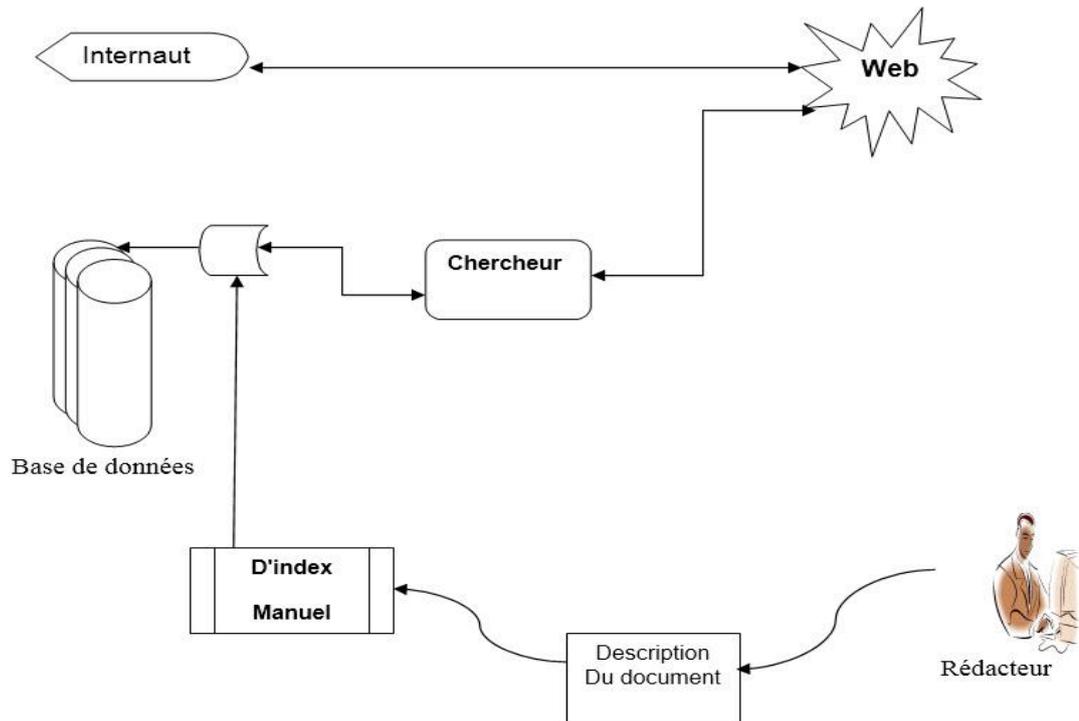


Figure I-1 : Les annuaires ou répertoires thématiques.

2.1.2. Avantages attribués aux annuaires :

- 👉 Lors d'une recherche, une sélection de sites correspondant à une catégorie précise est rapidement obtenue.
- 👉 Comme le contenu des sites a été vérifié par des humains, il y'a moins de chance d'obtenir des résultats erronés.
- 👉 Une qualité significative des sites répertoriés.

2.1.3. Inconvénients attribués aux annuaires :

- 👉 Les sites répertoriés doivent être inscrits.
- 👉 La mise à jour de la base de données est plus longue.
- 👉 Classement des résultats de la recherche par ordre alphabétique et non par pertinence des documents trouvés.
- 👉 Inscrire son site à un annuaire ne garantit aucunement son acceptation car les critères d'acceptations sont souvent sévères.

2.2. Les moteurs de recherche :

2.2.1. Qu'est-ce qu'un moteur de recherche ?

Un Moteur de recherche d'informations sur Internet est un outil permettant à un utilisateur, aussi bien novice que très expérimenté, d'accéder de manière simplifiée à des données dont la localisation lui est inconnue, ou dont les différentes parties sont disséminées sur le web.

Ceci est réalisable par la simple opération qui consiste à donner au Moteur les mots importants concernant le sujet des informations recherchées, dits mots-clés, ou la description même du document recherché, et ce Moteur a pour effet de retourner comme résultat de ses recherches la liste de toutes les pages web relatives à ces mots-clés, ou bien celles correspondant à la description passée en paramètre.

De façon scientifique. Un moteur de recherche est une immense base de données continuellement mise à jour par des "Robots" ou "Spiders" qui scrutent le Web en allant de page en page et qui les sauvegardent dans son index. Son fonctionnement est donc totalement automatisé.

Le rafraîchissement de la base se fait selon certains délais. Une fois les pages repérées, le moteur de recherche classe les pages par ordre de pertinence, selon un ordre et un algorithme basé sur des critères de tri qui lui sont spécifiques. C'est à la pertinence des résultats obtenus que l'on juge la qualité d'un moteur de recherche.

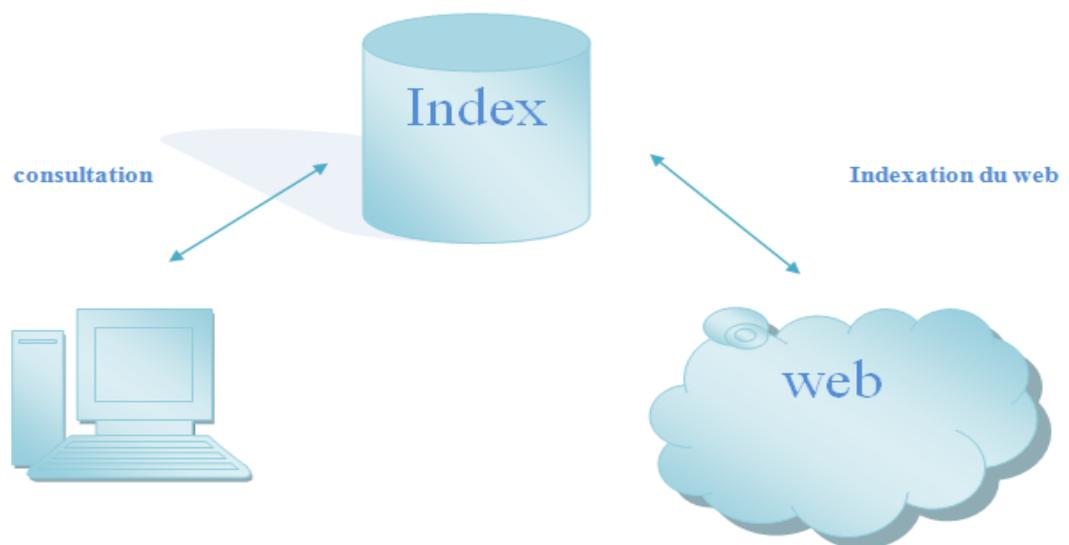


Figure I-2 :Moteur de recherche

2.2.2. Principes de base :

Les moteurs de recherche sont constitués de « robots », encore appelés bots, spiders, crawlers ou agents qui parcourent les sites à intervalles réguliers et de façon automatique (sans intervention humaine, ce qui les distingue des annuaires) pour découvrir de nouvelles adresses (URL). Ils suivent les liens hypertextes (qui relient les pages les unes aux autres) rencontrés sur chaque page atteinte. Chaque page identifiée est alors indexée dans une base de données, accessible ensuite par les internautes à partir de mots-clés.

Les moteurs de recherche ne s'appliquent pas qu'à Internet : certains moteurs sont des logiciels installés sur un ordinateur personnel. Ce sont des moteurs dits desktop qui combinent la recherche parmi les fichiers stockés sur le PC et la recherche parmi les sites Web — on peut citer par exemple Exalead Desktop, Google Desktop et Copernic Desktop Search, etc.

Des modules complémentaires sont souvent utilisés en association avec les trois briques de bases du moteur de recherche. Les plus connus sont les suivants :

1. Le correcteur orthographique : il permet de corriger les erreurs introduites dans les mots de la requête, et s'assurer que la pertinence d'un mot sera bien prise en compte sous sa forme canonique.
2. Le lemmatiseur : il permet de réduire les mots recherchés à leur lemme et ainsi d'étendre leur portée de recherche.
3. L'anti dictionnaire : utilisé pour supprimer à la fois dans l'index et dans les requêtes tous les mots "vides" (tels que "de", "le", "la") qui sont non discriminants et perturbent le score de recherche en introduisant du bruit.

En ce qui concerne les caractéristiques, les moteurs de recherche ont un fonctionnement commun, mais diffèrent par un certain nombre de critères. Pour ce qui est de commun, rappelons simplement qu'ils procèdent tous des mêmes étapes :

- D'abord l'exploration du web, durant laquelle ils vont collecter les informations sur chaque page rencontrée.
- Puis l'indexation, durant laquelle ils vont enregistrer dans une base de données les informations collectées.
- Enfin la recherche, durant laquelle ils vont rechercher les données collectées en fonction des mots clés. (BOURAMOUL, 2011)

2.2.3. Structure d'un moteur de recherche :

Afin d'assurer le rôle d'un moteur de recherche et couvrir tous les aspects liés à la recherche documentaire, il est principalement constitué de trois parties :

- ✓ Le robot (appelé aussi spider)
- ✓ L'index ou la base de données.
- ✓ Le logiciel/interface d'interrogation.

❖ *Le robot*

Le robot ou encore "spider" est la partie la plus importante du moteur de recherche, car celui-ci effectue la recherche directement sur le Web pour en extraire le plus grand nombre d'informations relatives aux documents présents sur le Web et les indexer au sein de sa base de données.

Un moteur de recherche utilise un robot qui balaie la structure hypertexte du Web par suivi récursif des liens pour en archiver intégralement son contenu. Il assure ainsi la lecture des données des pages Web et le repérage des liens pointant vers d'autres pages afin de constituer l'index.

❖ *L'index*

L'index est le lieu de stockage et d'indexation des pages Web visitées par le robot, c'est la phase de structuration et de classification de l'information rapatriée. En général, les informations répertoriées sont : l'adresse (URL), le titre des pages, les mots clés voir même l'intégralité des pages. L'entrée de pages Web dans l'index est également rendue possible par la visite du robot pages soumises volontairement par leurs créateurs. Afin d'actualiser le contenu des pages Web dans l'index, le robot se rend à intervalle défini sur celles-ci, pour mettre à jour leur contenu dans l'index.

Les méthodes d'indexations diffèrent d'un moteur à l'autre influençant ainsi les résultats obtenus suite à une requête. Une fois les documents collectés par les robots, les moteurs d'indexations utilisent différentes méthodes pour indexer le contenu des pages, parmi les plus courantes :

- Indexation complète du texte (Full-Text)
- Indexation Manuelle par l'intermédiaire d'un fichier de soumission à compléter par tout opérateur de site désirant faire figurer ses pages Web dans la base de données du moteur.
- Indexation de balises spécifiques (les balises Méta, Title des pages HTML)
- Indexation statistique consistant à supprimer les mots de sens vide et à ne retenir que ceux qui dépassent un seuil de fréquence d'apparition (Scoring).

❖ L'interface

L'interface permet à l'utilisateur de saisir sa requête en utilisant un ou plusieurs termes significatifs qui seront ensuite recherchés dans l'index. L'interface sélectionne parmi les milliers de documents enregistrés dans l'index ceux qui satisfont cette requête et les propose sous forme d'une liste de pages Web énumérées selon un ordre de pertinence décroissant, pages sur lesquelles l'utilisateur peut ensuite se rendre via un lien hypertexte.

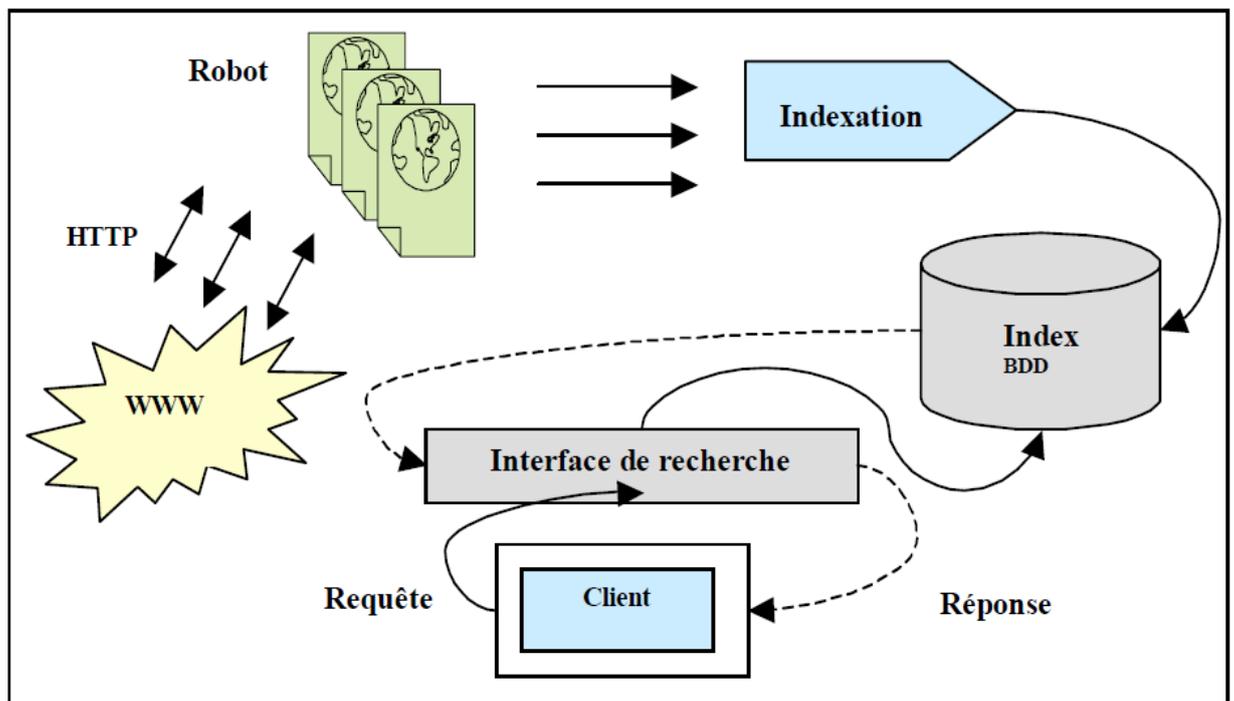


Figure I-3 : Architecture générale d'un moteur de recherche.

2.2.4. Fonctionnement d'un moteur de recherche :

Le mode de fonctionnement d'un moteur de recherche peut-être schématisé de la manière suivante :

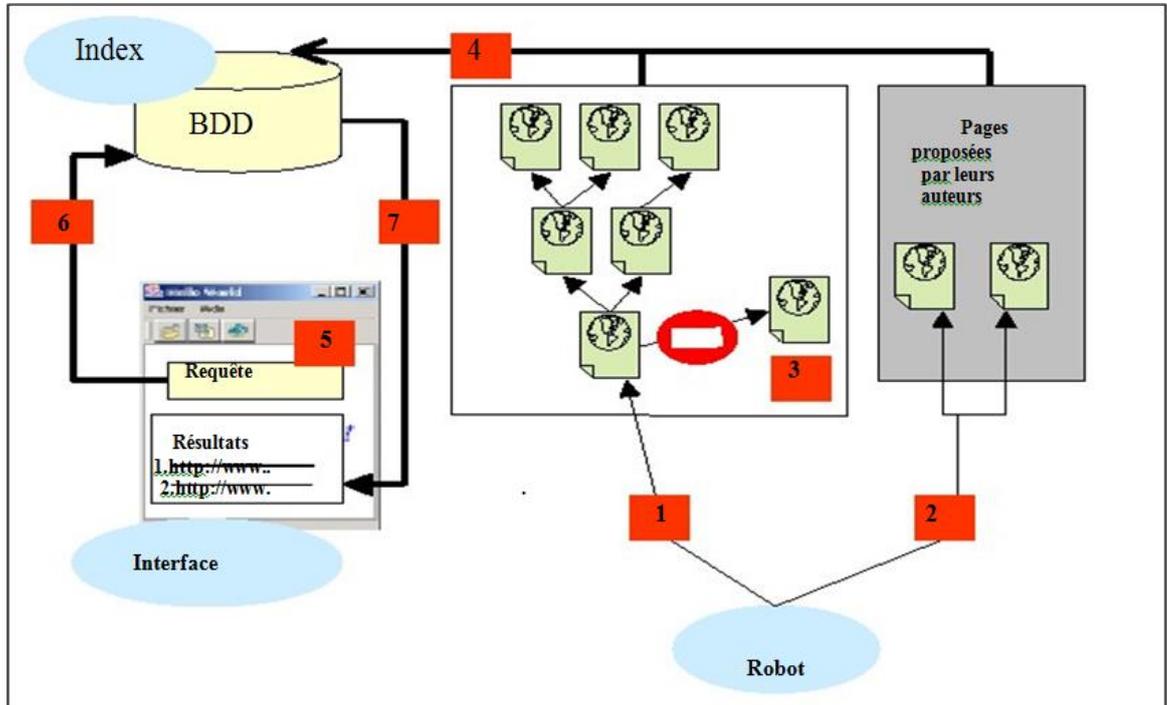


Figure I-4 : *Fonctionnement d'un moteur de recherche.*

Le robot repère les pages Web par suivi récursif des liens appartenant aux pages présentes sur le réseau (1) ou proposées par les auteurs (2). Toutefois, les pages à accès réservé ne seront pas indexées (3) alors que les autres seront indexés au sein de la base de données (4). La requête est émise par l'utilisateur à travers l'interface de recherche (5) dont les termes seront recherchés dans l'index (la base de données) (6). Les résultats sont ensuite affichés à travers l'interface sous forme de liens hypertextes (7).

2.2.5. Le Robot (Spider)

Le "spider" ou encore le robot est la partie la plus importante du moteur de recherche, car celui-ci effectue la recherche directement sur le réseau Intranet pour en extraire le plus grand nombre d'informations relatives aux documents présents et les indexer au sein de sa base de données.

Un robot est un programme qui traverse la structure hypertexte du réseau Intranet pour retrouver un nombre maximum de documents, par un suivi récursif des liens, en utilisant des

protocoles standards du Web et tout en faisant abstraction des sites qui n'autorisent pas la visite automatique des spiders.

Afin d'appréhender le robot, nous le présentons comme suit :

- La composition du robot.
- Le fonctionnement général du robot.

2.2.5.1. Composition du robot :

Un robot est généralement composé de trois modules effectuant chacun une fonction déterminée :

•Crawler

Le crawler a pour rôle de télécharger des documents à partir d'adresses URL en utilisant plusieurs techniques pour télécharger le plus grand nombre de pages sur une période de temps minimale.

•Analyseur

L'analyseur a pour tâche l'extraction des liens hypertextes présents dans la page, ainsi que l'extraction des mots nécessaires à une bonne indexation selon le type de document à indexer (HTML, PDF, DOC...).

•Indexeur

L'indexeur finit par récupérer l'URL du document ainsi que le résultat de son analyse intégrale (ensemble de mots) afin de les insérer dans la base de données.

Le résultat final de notre robot sera une base de données qui peut être consultée par les utilisateurs à travers une interface d'interrogation.

2.2.5.2. Fonctionnement générale du robot

D'après les trois fonctions citées précédemment, une conception générale du robot peut-être donnée dans la figure suivante :

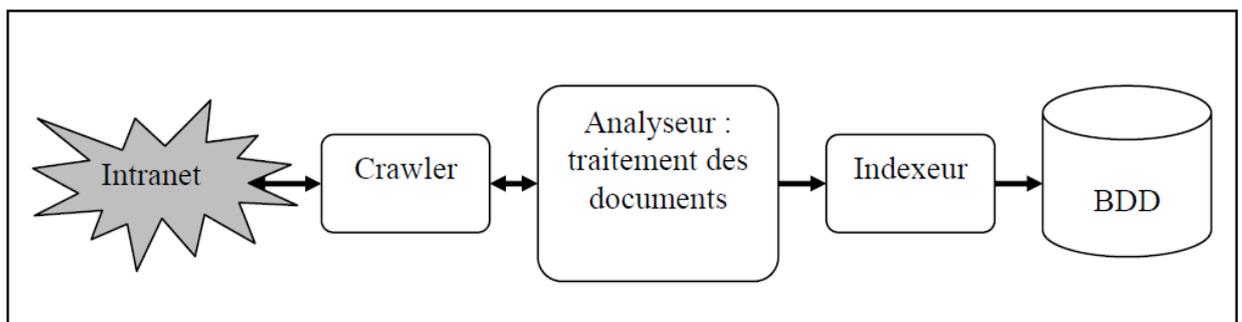


Figure I-5 : Fonctionnement générale du robot.

2.2.6. Architectures des moteurs de recherche :

2.2.6.1. Architecture générale des premiers moteurs de recherche

L'architecture originale utilisée par Altavista représente la première catégorie de systèmes. Il s'agit d'une architecture très simple qui se divise en deux parties distinctes. On retrouve d'une part un crawler et d'autre part l'interface d'interrogation du moteur de recherche et le système d'analyse des requêtes proposés par les utilisateurs du système.

Le cœur du système repose sur un index inversé permettant d'associer des mots à un ou plusieurs documents. La demande de l'utilisateur est traitée en interrogeant l'index inversé pour connaître les documents dans lesquels apparaissent le plus souvent les mots de la requête. (PICAROUGNE, 2004)

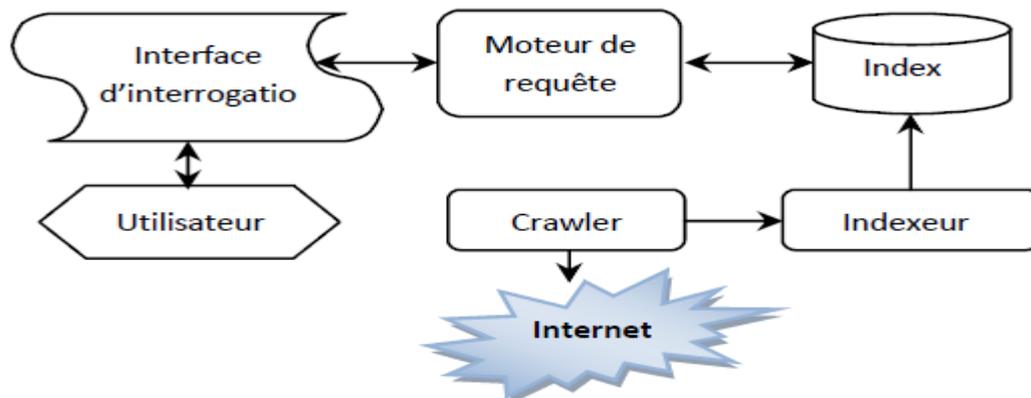


Figure I-6 : Architecture originale du moteur de recherche *Altavista*. (PICAROUGNE, 2004)

2.2.6.2. Architecture distribuée et adaptative

Des variantes de l'architecture précédente, basées sur le modèle indexeur-crawler, ont été imaginées pour gommer les défauts inhérents à sa conception. L'une d'entre elle, appelée Harvest s'est révélée très innovante en matière de distribution des ressources.

Le récolteur : est chargé de collecter et d'extraire périodiquement des informations d'indexation -textes, images - depuis plusieurs sites Web.

Le broker : quant à lui, fournit le mécanisme d'indexation et l'interface d'interrogation sur les données amassées par le récolteur.

On retrouve ici, le mécanisme indexeur-crawler identifié dans la section précédente. Cependant, plusieurs brokers et plusieurs récolteurs peuvent communiquer ensemble, chacun se spécialisant dans un domaine précis. Lorsqu'une requête est émise sur un broker dont le

domaine traité ne correspond pas à ses capacités, celui-ci transmet la requête à une autre entité capable de la gérer.

C'est un système totalement adaptatif dans lequel il est possible de configurer les brokers et les récolteurs de manière à répartir le besoin en ressources sur un ou plusieurs domaines particuliers. (PICAROUGNE, 2004)

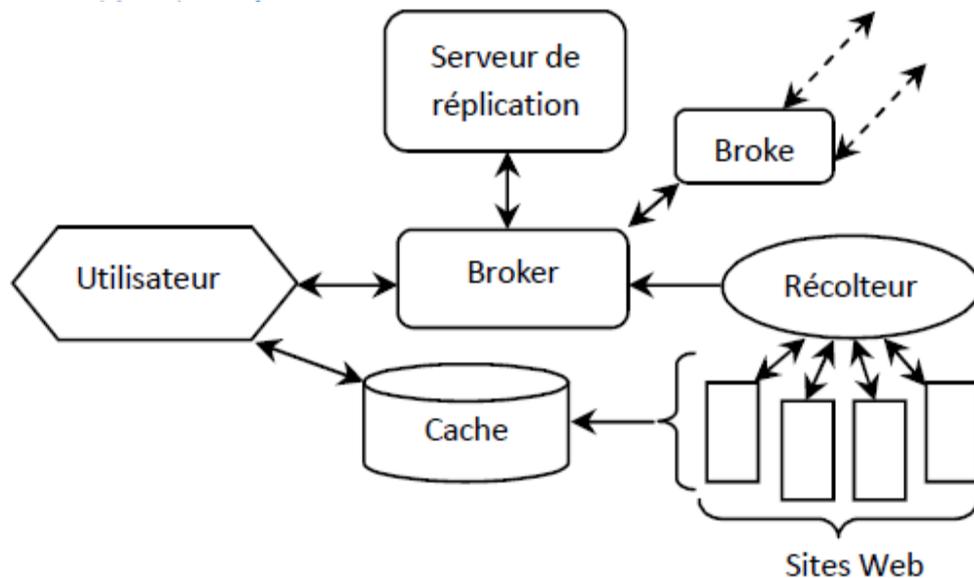


Figure I-7 : Architecture du système *Harvest*. (PICAROUGNE, 2004)

2.2.6.3. Architecture moderne d'un moteur de recherche

L'architecture du moteur de recherche Google est certainement une des plus efficaces actuellement. Elle ne repose pas sur un système monolithique mais sur un grand nombre de machines classiques coopérant ensemble. Ce système peut se décomposer en plusieurs parties comprenant :

- Un sous-système d'exploration d'Internet
- Un indexeur
- Un analyseur de la topologie d'Internet formée par les liens hypertextes : et un sous-système de présentation et d'exécution de requêtes.
- Un serveur d'URL garde la mémoire des liens des pages à visiter. Des robots chargés d'explorer le Web récupèrent ces liens afin de télécharger les documents correspondant et les stocker dans une base de données recensant la totalité des pages indexées. Cette opération est réalisée continuellement et alimente et met à jour en permanence la base de documents du moteur. Périodiquement, cette base est analysée pour réaliser un index inversé reliant des termes aux documents les contenant.

D'autres informations sur les termes sont extraites comme leur position dans le document, la taille de la police utilisée ou sa fonte.

Cette analyse permet également d'extraire tous les liens hypertextes des documents rencontrés afin d'alimenter le serveur d'URL. Cette base de liens est utilisée afin de calculer le PageRank permettant de trier les documents de l'index par pertinence décroissante. (PICAROUGNE, 2004)

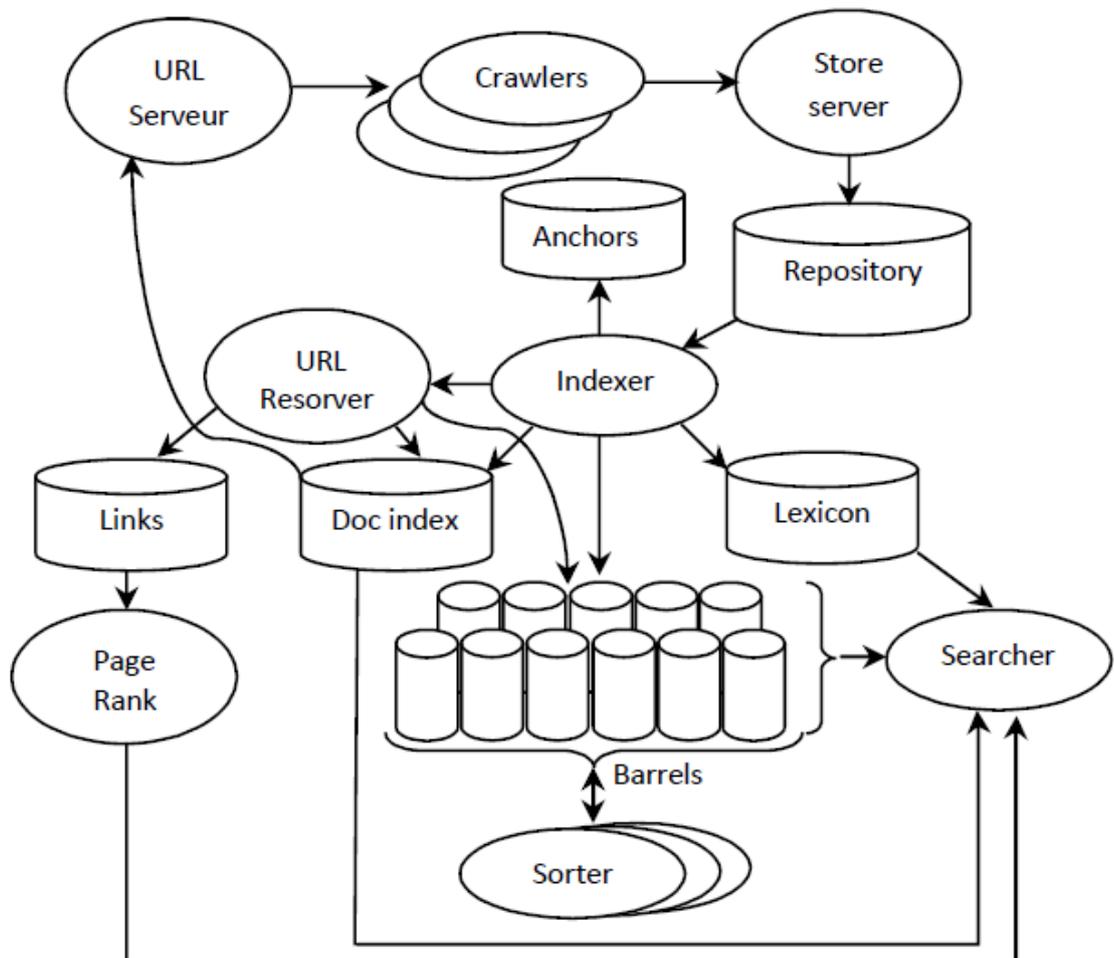


Figure I-8 :Architecture du moteur de recherche *Google*. (PICAROUGNE, 2004)

2.2.7. Domaines d'utilisations des moteurs de recherche :

Les moteurs de recherche sont utilisés dans toutes les applications où il est nécessaire d'accéder au contenu de l'information tant pour la recherche que pour le filtrage ou la diffusion sélective. Parmi ces applications, nous pouvons citer celles de gestion de bibliothèques et de centres de documentation, la recherche dans un ensemble de sources d'informations qui s'intéressent à un domaine précis, par exemple un petit moteur de recherche fournis avec un site web spécialisé (informatique, médecine, théories d'algèbre

...etc.), qui cherche dans le lot des documents appartenant au site un sujet précis que demande l'utilisateur.

2.2.8. Les restrictions d'accès à l'information :

Une autre fonctionnalité en plus de celles décrites plus haut, est la confidentialité des informations retournées. Il faut, cependant, garder à l'esprit que toutes les informations ne doivent pas être systématiquement accessibles à tous. Certaines informations doivent même rester confidentielles ou être accessibles à un nombre restreint d'utilisateurs. D'une façon générale, l'entreprise établit des critères d'accessibilité à chaque information ou ensemble d'informations, tels que des documents, des bases de données... etc. La gestion de l'information sous forme électronique doit également tenir compte de ces paramètres.

2.2.9. Avantages attribués aux moteurs de recherche :

- ☞ Les recherches donnent plus de résultats car la base de données d'un moteur de recherche est très importante (par exemple La base de données de Google est stockée dans dix serveurs ou Datacenter).
- ☞ La base de données est mise à jour fréquemment à intervalle régulier.
- ☞ Le classement des résultats des recherches est effectué par pertinence et non par ordre alphabétique.

2.2.10. Inconvénients attribués aux moteurs de recherche

- ☞ Les recherches peuvent générer une grande masse de résultats (bruit)
- ☞ Comme le contenu des sites n'est pas examiné par des humains, la qualité des résultats peut être moindre.
- ☞ Les moteurs de recherche n'interrogent pas directement le Web mais leurs bases de données contenant les termes décrivant chaque page Web ce qui inclut que les moteurs peuvent difficilement suivre les mises à jour des sites où l'information change quotidiennement et donc ne peuvent assurer une indexation totalement fiable d'un environnement dynamique comme celui du Web

2.3. Les méta-moteurs :

2.3.1. Qu'est-ce qu'un méta-moteur :

Les méta-moteurs sont des outils créés plus récemment que les précédents.

Les méta-moteurs sont des outils de recherche qui interrogent non leur propre base de données, mais celles de plusieurs moteurs de recherche simultanément et affichent à l'internaute une synthèse pertinente.

Deux catégories de méta-moteurs : ceux en ligne et ceux consistant en un "logiciel client" à installer sur son ordinateur (le plus connu : COPERNIC).

Le principe de fonctionnement des méta-moteurs est différent, Certains indexent l'information contenue dans différents annuaires et moteurs, d'autres les interrogent simultanément de façon dynamique. Certains de ces méta-moteurs retraitent plus au moins les réponses (tri, déboullonnage). Ils permettent ainsi de rechercher de façon plus large sur le Web. Toutefois, cela peut également générer du "bruit" (réponses non pertinentes).

La parade mise en œuvre par certains méta-moteurs consiste à limiter le nombre de réponses de chaque outil interrogé (ce qui est indispensable et permet ainsi d'obtenir les réponses en principe les plus pertinentes). (LARGOUET, 2005)

2.3.2. Principes de base :

Le principe d'un méta-moteur est de permettre l'interrogation simultanée de plusieurs indexes de moteurs de recherche différents. La saisie de la requête s'effectue à travers une interface unique qui peut être accessible via un site Web ou via un logiciel qu'il est nécessaire d'installer sur un poste client. La requête est alors soumise aux différents moteurs de recherche interrogés dont le nombre varie de quelques-uns à plusieurs dizaines selon les méta-moteurs considérés. Les réponses provenant de ces différents moteurs subissent le plus souvent une élimination des doublons et sont présentées par ordre de pertinence à l'utilisateur, ce dernier peut se rendre sur chaque page proposée, via un lien hypertexte.

Les principaux méta-moteurs actuels sont : **infind**, **debriefing**, **dogpile**, **eo**...etc.

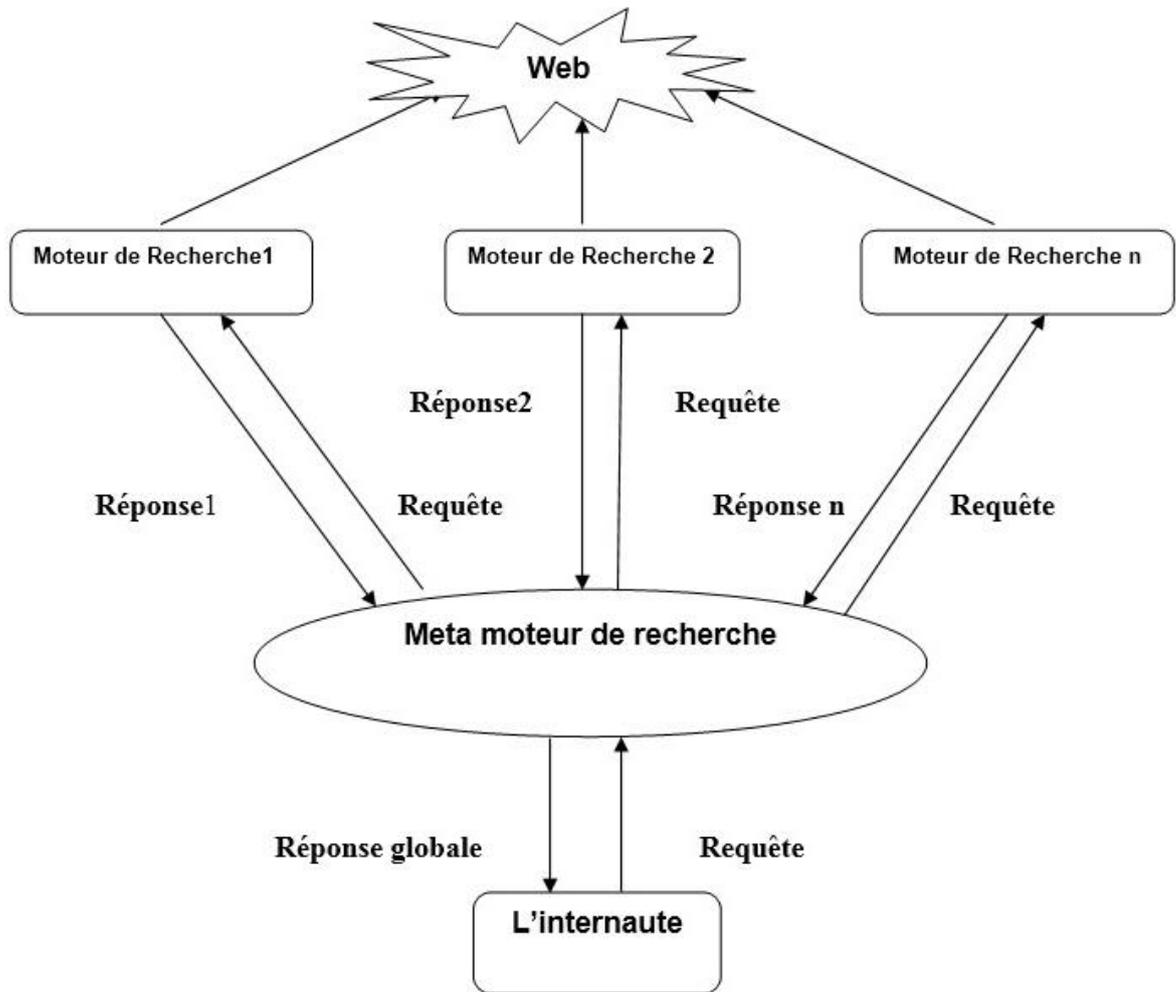


Figure I-9 : Méta-moteur de recherche

2.3.3. Quelques méta-moteurs :

❖ **Kartoo** (<http://www.kartoo.com/>)

Ce méta-moteur intelligent puise ses résultats dans une quinzaine de moteurs français et anglais (Yahoo, Google, Excite, Nomade...) et affiche les résultats sur une carte thématique. Il propose des fonctionnalités innovantes telles que l'interrogation en langage naturel, l'interprétation booléenne automatique, l'envoi des résultats à un tiers, ...

❖ **Ixquick** (<http://www.ixquick.com/>)

Ixquick connaît les moteurs de recherche qui peuvent s'occuper des expressions, de la logique booléenne, des caractères jokers et d'autres commandes de recherche. Ixquick traduira puis expédiera votre recherche, uniquement sur les moteurs qui peuvent gérer la complexité de votre recherche.

❖ **Mamma** (<http://www.mamma.com/>)

Mamma interroge simultanément dix des principaux moteurs de recherche après leur avoir adapté le format des mots et la syntaxe. Mamma crée ensuite une base de données virtuelle, organise les résultats, les met dans un format uniforme et les présente selon leur pertinence et leur source.

❖ **MapStan Search** (<http://search.mapstan.net/>)

Ce méta-moteur de capitalisation des recherches vous offre : une vision synthétique des résultats, des suggestions complétant les résultats, un catalogue dynamique des sites WEB

2.3.4. Avantages attribués aux méta-moteurs :

- ☞ Augmentation de la taille de l'index interrogé.
- ☞ Résultats de recherches beaucoup plus importants

2.3.5. Inconvénients attribués aux méta-moteurs :

- ☞ Même si la plupart des méta-moteurs assurent une traduction de la requête pour l'adapter à la syntaxe de chacun des moteurs interrogés, l'utilisation de requêtes complexes entraîne le plus souvent des réponses très bruitées.
- ☞ Le temps de recherche est plus long.
- ☞ Difficulté d'intégration des résultats retournés par différents moteurs utilisant des méthodes et des techniques différentes.

2.4. L'indexation :

L'objectif d'indexation est de trouver les concepts ou représentants les plus importants dans les documents et de créer une représentation interne en utilisant ces concepts, en pratique ces concepts peuvent être des mots simples ou composés (groupe de mots), L'idée d'utiliser des termes simples comme des représentants de concepts est assez naturelle. (Karima AMROUCHE, 2008)

L'indexation consiste à identifier l'information contenue dans tout texte et à la représenter au moyen d'un ensemble d'entités appelé index pour faciliter la comparaison entre la représentation d'un document et d'une requête. (Roussey C, 2001)

Donc l'indexation consiste à analyser les documents afin d'extraire un ensemble des mots clés servant comme descripteurs des documents. Elle peut être :

- ✓ Manuelle : la représentation du document se fait par un spécialiste (documentaliste).
- ✓ Automatique : la représentation du document est totalement automatisée.
- ✓ Semi-automatique : l'extraction des descripteurs s'effectue par le système et le choix des descripteurs est laissé au spécialiste.

L'indexation peut être basée sur un langage contrôlé (lexique, ontologie, réseau sémantique, thesaurus) ou libre (les termes sont pris directement à partir des documents). L'indexation à base de langage contrôlé permet une recherche par concepts (par sujets, par thème) plus intéressante que la recherche par terme. Le processus d'indexation se compose d'un ensemble de traitements : l'analyse lexicale, l'élimination des mots vides, la lemmatisation, la pondération et enfin la création de l'index (Bal Kamal, 2010)

2.4.1. Analyse lexicale

C'est la tâche consistant à décomposer le contenu de document en mot simple ou composé, afin de trouver l'ensemble des termes appartenant à un document, cette étape se fait dépend fortement de la langue des documents à indexer. Cette extraction est effectuée en tenant compte des espaces, des chiffres et des ponctuations. Un terme peut être un mot simple ou composé, mais en RI on utilise souvent les mots simples.

2.4.2. Elimination des mots vides

Les mots vides sont des mots trop fréquents peu significatifs et porteurs de peu de sens, augmentant ainsi la taille de l'index et rendant la recherche plus lente. L'élimination de ces mots permet de réduire l'index, on gagne alors en espace mémoire.

On distingue deux techniques pour éliminer les mots vides :

- L'utilisation d'une liste prédéfinie de mots vides (aussi appelée anti-dictionnaire ou stop list), par exemple cette liste pourra contenir les termes (the, or, a, you, I, us, of, in...) pour l'anglais, (le, la, de, des, je, tu...) pour le français.
- L'élimination des mots dépassant un certain nombre d'occurrences dans le document.

2.4.3. Lemmatisation

Un mot peut avoir plusieurs formes dans un texte dont le sens est presque similaire. La lemmatisation est une technique qui permet de ramener un mot à sa racine. Par exemple, programmes et programme, programmer et programmation, programmeurs et programmés font tous référence à la racine 'programme'. Elle désigne l'analyse lexicale du contenu textuel regroupant les mots d'une même famille afin de réduire les mots à leurs racines grammaticales.

L'algorithme de Porter est sans doute le plus connu dans ce domaine, elle est un procédé pour éliminer les terminaisons plus communes morphologiques et flexionnelles des mots en

anglais. Son utilisation principale est dans le cadre d'un processus de normalisation terme qui se fait habituellement lors de la mise en place des systèmes de recherche d'information. (Porter, 2006)

2.4.4. Pondération

Dans un document, certains termes sont plus représentatifs du contenu et de la sémantique du document par rapport l'autres. L'objectif de la pondération est de trouver les termes qui représentent le mieux le contenu d'un document. La pondération des termes permet de mesurer l'importance d'un terme dans un document, cette importance est souvent calculée à partir de considérations et d'interprétations statistiques (ou parfois linguistiques). Les termes importants doivent avoir un poids fort.

2.4.5. Création de l'index

Les informations sélectionnées lors du processus d'indexation sont mémorisées et enregistrées dans une structure appelée index. Cette structure permet de sélectionner pour n'importe quel terme, tous les documents le contenant. Il y a deux solutions plus utilisée actuellement fichiers inverses et fichiers maitres.

- Les fichiers inverses : sont composés de deux éléments principaux : le dictionnaire et le fichier posting. Le dictionnaire consiste en une liste de tous les mots distincts de la collection. Pour chaque mot est assigné l'ensemble des documents dans lesquels ce dernier apparaît (posting).
- Les fichiers maitres : Au lieu de donner pour chaque document les mots et les fréquences qui le constituent, on donne pour chaque mot les documents où il apparaît et sa fréquence dans chacun de ces documents.

2.4.6. Les langages d'indexation

- Langage contrôlé : il s'agit d'un lexique figé de descripteurs. L'indexation est alors le plus souvent manuelle, parfois semi-automatique, un professionnel choisit un ou plusieurs descripteurs pour représenter le document.
- Langage libre : les descripteurs sont extraits automatiquement des documents, ou de la requête de l'utilisateur. Ici, un document est le plus souvent indexé par la liste des mots qui le composent

.

3. MOTEURS DE RECHERCHE INTELLIGENTS :

Pour dire qu'un moteur de recherche est intelligent, il s'agit de comprendre les questions de l'utilisateur et lui donner des réponses spécifiques, sur mesure, comme le ferait un être humain. Mais il apparaît très vite que cette notion est assez difficile à définir. En effet, tous les jours, les gens utilisent le contexte dans leurs prises de décision. Prenant l'exemple de la phrase suivante "S'il pleut, je vais prendre un parapluie pour me rendre à l'université". Le fait de prendre un parapluie n'a rien à voir avec le fait d'aller à l'université mais pourtant cela contraint la manière d'exécuter la tâche d'aller à l'université. La mise en évidence du contexte est particulièrement visible quand il y a plusieurs méthodes pour accomplir une tâche. Dans ce cas, chaque personne choisit sa méthode en fonction de ses connaissances et des informations contextuelles qu'il possède. Il s'avère donc que le traitement des données contextuelles joue un rôle dans tous les domaines où le raisonnement intervient. (BOURAMOUL, 2011)

Alors il est très important pour un moteur de recherche intelligent d'être capable de traiter ces langages naturels. Et ça se base sur la notion de Web sémantique. Aussi, les modules complémentaires (ensembles des agents intelligents intégrés) est un moyen d'ajouter l'intelligence au moteur de recherche.

Un moteur de recherche intelligents généralement contient les caractéristiques suivantes :

- ✓ Suggérer des mots proches : Certains moteurs de recherche proposent à l'utilisateur des termes proches de ceux de sa requête, ce qui permet de la préciser. Ces termes proches sont identifiés par des méthodes statistiques ou bien en se basant sur des dictionnaires ou ontologies.
- ✓ Les connaissances plutôt que les informations
 - Tentatives d'appréhender non plus les informations brutes (présence de tel mot dans telle page d'un site Internet) mais des informations qualifiées ou connaissances.
 - Nécessite des pages de sites où les informations sont qualifiées.
 - Recherches sur des informations qualifiées.
- ✓ Exploitation du sens sémantique des contenus.
- ✓ Standardisé et formalisé par Ontologie (OWL), Description des ressources, Métadonnées, annotation, URI, triplets
- ✓ machines exploitent, interprètent et combinent les ressources.
- ✓ Vaste espace d'échange de ressources entre machines et utilisateurs.

3.1. Exemple d'un moteur de recherche intelligent :

3.1. WolframAlpha :

Wolfram Alpha est un outil de calcul en langage naturel développé par la société internationale Wolfram Research. Il s'agit d'un service internet qui répond directement à la saisie de questions factuelles en anglais par le calcul de la réponse à partir d'une base de données, au lieu de procurer une liste de documents ou de pages web pouvant contenir la réponse. Son lancement a été annoncé en mars 2009 par le physicien et mathématicien britannique Stephen Wolfram et il a été lancé le 16 mai 2009 à 3 h 00 du matin. Wolfram Alpha contient environ 10 milliards d'informations, plus de 50 000 types d'algorithmes et de modèles, et des capacités linguistiques pour plus de 1 000 domaines. **(site 3)**

Les utilisateurs saisissent une question ou une demande de calcul. Le service calcule les réponses et les visualisations correspondantes à partir d'une base de connaissance.

Grâce à l'utilisation de l'outil Mathematica, Wolfram|Alpha est capable de répondre à des questions mathématiques. La réponse est généralement présentée sous une forme lisible par un être humain.

Exemple : $\lim_{x \rightarrow 0} x/\sin x$ fournit la réponse attendue, 1, ainsi qu'une façon de l'obtenir en utilisant la règle de L'Hôpital.

Wolfram|Alpha est aussi capable de répondre à des questions factuelles posées en anglais naturel, telles que « Where was Ségolène Royal born? » (« Où Ségolène Royal est-elle née ? »), ou des questions plus complexes telles que « How old was Nicolas Sarkozy in 1981? » (« Quel âge avait Nicolas Sarkozy en 1981 ? »). Wolfram|Alpha affiche son interprétation de la question saisie (« Input interprétation ») à l'aide de phrases standardisées, par exemple « Ségolène Royal | place of birth » ou « age | of Nicolas Sarkozy (politician) | »

Wolfram|Alpha analyse des données issues de disciplines très variées, dont notamment les mathématiques, statistiques, analyse de données, physique, chimie, science des matériaux, ingénierie, astronomie, sciences de la vie et de la Terre (géologie), nouvelles technologies, dates et heures, lieux et géographie, données socioéconomique, météorologie, santé et médecine, alimentation et nutrition, linguistique, culture, médias, personnalités, histoire, éducation, organisations diverses, jeux et sports, musique, couleurs, ...etc.

The screenshot shows the WolframAlpha interface. At the top, there's a navigation bar with 'HOME', 'EXAMPLES', 'PRODUCTS', 'BLOG', and 'ABOUT'. Below that, a promotional banner for 'Wolfram|Alpha Pro' is visible. The main search area contains the query 'what's the name of the second Algerian president?' in the input field. Below the input, the 'Input interpretation' shows 'Algeria President 2nd'. The 'Result' section displays 'Houari Boumedienne'. To the right of the result, there's a table of 'Basic information' with the following data:

official position	President
country	Algeria
start date	19/06/1965 (48 years 1 day ago)
end date	27/12/1978 (34 years 5 months 24 days ago)
duration of leadership	13 years 6 months 8 days

On the left side, there are navigation options like 'Favorites', 'History', 'Preferences', 'Downloads', 'Uploads', and 'Account'. A 'Related Queries' section lists items like '3rd president of AL...', 'prime minister of A...', 'bordering countries...', and 'Gini index of Alger...'. On the right, there are social media sharing options and a 'Hey, teacher! Join us on' banner.

Figure I-9 :Exemple d'un question dans Wolfram Alpha

4. COMPARAISON ENTRE LES OUTILS DE RECHERCHE

A travers les avantages et les inconvénients énumérés pour chacun des outils de recherche cités précédemment, il devient aisé d'établir une comparaison et de définir les principales différences entre ces outils. Ces différences ont des incidences sur les résultats de la recherche, ainsi, la connaissance de ces outils est nécessaire pour l'obtention de meilleurs résultats lors d'une recherche sur le Web (Internet ou Intranet).

Les annuaires sont généralement utilisés pour effectuer une recherche sur un sujet général, c'est-à-dire au début du processus de recherche, pour cerner le sujet et retrouver les sites de référence sur ce dernier. Les annuaires sont alors associés à une recherche thématique.

A la différence des annuaires, les moteurs de recherche indexent des pages Web, selon une approche totalement automatisée et sont généralement utilisés pour trouver des informations sur un sujet précis dont les termes clés sont déjà connus, on parle alors de recherche par interrogation via l'utilisation de mots clés. Chaque moteur de recherche possède un système de classement différent. Les résultats affichés par les moteurs sont des pages Web tandis que ceux provenant des annuaires sont des sites.

En fait, certains outils de recherche présentent à la fois une fonction de répertoire et de recherche automatique. De plus en plus, certains répertoires se sont associés avec un moteur de recherche qui fournit des résultats non trouvés dans la base de données (Le moteur de recherche Google intègre le répertoire Dmoz depuis mars 2000). (FORESTIER, 2003)

Les moteurs de recherche sont des outils très importants qui permettent de faciliter la recherche d'informations sur le Web, cependant il est nécessaire de souligner que les moteurs ne permettent pas un recouvrement total du Web, et donc ne s'adaptent pas à l'évolution du Web.

Les méta-moteurs permettent d'adresser simultanément une même requête à différents moteurs de recherche pour augmenter le nombre de résultats. Les méta-moteurs sélectionnent les réponses en fonction de leur pertinence. Ils peuvent être considérés comme une évolution des moteurs de recherche, cependant ils souffrent d'un temps d'interrogation plus long et d'une difficulté d'interrogation des résultats provenant de sources d'information hétérogènes.

Nous nous intéressons dans notre étude tout particulièrement aux moteurs de recherche qui peuvent être facilement intégrés dans un contexte Intranet, en les dotant d'une interface d'indexation automatique ainsi qu'une autre interface pour l'indexation manuelle des documents fournis par leurs auteurs afin de les référencer auprès de notre moteur de recherche. Dans un intranet, vu que la densité de l'information circulante est moins intense qu'Internet et le nombre des utilisateurs et des documents est beaucoup plus réduit, il est préférable de fournir un outil hybride qui profite des avantages de toutes les technologies de recherches existantes.

5. CONCLUSION :

Internet nous donne accès à une masse considérable d'informations, des dizaines de millions de pages. Les documents publiés embrassent à peu près tous les sujets imaginables et sont diffusés aussi bien par des institutions que par des associations, des entreprises ou des individus. Ils sont de présentation et de qualités très inégales.

Attendez-vous donc à rencontrer le pire et le meilleur sur Internet. Pour y trouver de l'information intéressante et valable, vous devrez faire preuve d'ingéniosité dans vos recherches et de beaucoup de sens critique pour sélectionner les documents. Il est donc intéressant, non seulement de savoir comment rechercher de l'information, mais de savoir aussi comment la sélectionner. C'est pourquoi il est impératif d'introduire la notion d'outils de recherche.

Dans ce chapitre, nous avons présenté les différents outils de la recherche sur le Web, les annuaires, les moteurs de recherche et les méta-moteurs. Avec une présentation de leurs principes de base, leurs fonctionnements, avantages, inconvénients et quelques exemples. Ainsi, nous avons présenté une comparaison entre ces outils de recherche.

Dans les chapitres suivants, on va découvrir deux techniques utilisés pour l'amélioration de performance de ces outils de recherche.

Chapitre II

LES AGENTS

Dans ce chapitre :

- ❖ Introduction
- ❖ Généralité sur les agents
- ❖ Les systèmes Multi-Agents (SMA)
- ❖ Les agents intelligents
- ❖ Conclusion

1. INTRODUCTION

Depuis 1993 et le démarrage du World Wide Web, Internet fascine. A la fois, les techniciens, les hommes d'affaires, les banquiers. Même les hommes politiques et les mannequins s'y précipitent. Et depuis cette période, le nombre de sites, le nombre de pages, le nombre d'utilisateurs augmentent sans arrêt. Cet engouement ne tarit pas. Pour les américains, si friands de nouvelles frontières, le WEB est la dernière en date. Pour les bourses, la nouvelle économie fait rêver. Mais quelques problèmes sont apparus, qu'il faudra régler afin de ne pas effrayer les investisseurs : les temps de réponse catastrophiques, les recherches difficiles dans cette montagne d'information, la sécurité des sites, la confidentialité des informations, la criminalité naissante, la protection de l'ensemble du réseau et son administration. La technologie des Agents arrive à point nommé pour aider à gérer cette «gigantesque pagaille».

Dans ce chapitre en va présenter la définition des quelques termes importants. On définit les agents intelligents et leurs caractéristiques, leurs types, classification et domaines d'application.

2. GENERALITE SUR LES AGENTS :

2.1.Définitions :

En réalité, le terme agent est une métaphore. Ce mot vient du verbe latin "**agere**" qui signifie conduire ou agir pour quelqu'un d'autre par délégation. Le programme, donc, agit comme un humain à qui on a confié une tâche ou une mission. (John Richard, 2005)

Dans la littérature, on trouve plusieurs définitions d'agents qui se ressemblent mais qui diffèrent selon le type d'application pour lequel l'agent est conçu. D'après Ferber (J. Ferber, 1995), on appelle agent une entité physique ou virtuelle qui est capable d'agir dans un environnement, communiquer directement avec d'autres agents, qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser), possède des ressources propres, capable de percevoir (mais de manière limitée) son environnement, ne dispose que d'une représentation partielle de

cet environnement (et éventuellement aucune), possède des compétences et offre des services, peut éventuellement se reproduire, et dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

Sycara et Wooldridge (Chaib-draa, 1999) ont proposé la définition suivante pour un agent : un agent est un système informatique, situé dans un environnement, qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.

Jean-Pierre Briot et Yves Demazeau (Jean-Pierre et Yves, 2001) : « un agent est une entité logicielle ou physique à qui est attribuée une certaine mission qu'elle est capable d'accomplir de manière autonome et en coopération avec d'autres agents ».

H. Jens et all (LAWRENCE & GILES, 1998) ont donné la définition suivante : « Un agent est un morceau de logiciel qui peut accomplir une tâche prédéfinie spécifique de façon autonome (au nom d'un utilisateur ou d'une application) ».

En général, un agent représente un composant logiciel réutilisable qui fournit un accès contrôlé à des services et des ressources. Figure II.1 représente un agent dans son environnement, l'agent est activé en entrée par les capteurs de l'environnement et produit en sortie des actions.

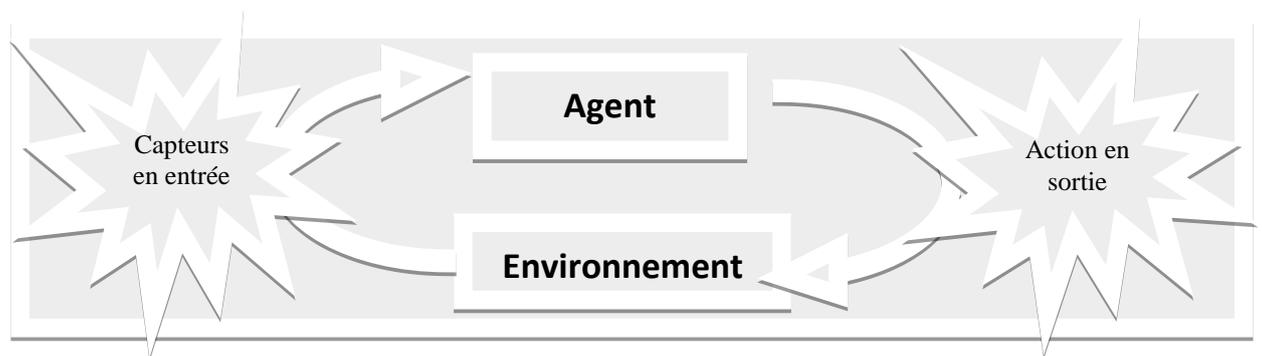


Figure II-1. *L'agent dans son environnement.*

2.1. Caractéristiques des agents

Un agent doit posséder les caractéristiques suivantes :

Autonomie

- Un agent a un certain degré d'autonomie.
- Il possède certains états (non-accessibles aux autres agents et composants du système).
- Il peut prendre certaines décisions par rapport à ses états (sans intervention externe directe).

Situé

- Un agent est situé dans son environnement (physique ou virtuel).
- Il a une représentation de son environnement.

Réactif

- Un agent peut percevoir son environnement via des senseurs.
- Il peut agir sur son environnement via des effecteurs.

Social

- Un agent est capable d'interagir et de communiquer avec les autres agents (par des langages de communication).
- Il est capable de coopérer pour résoudre des problèmes ou effectuer des tâches.

Proactif

- Un agent est capable de « prendre de l'initiative » pour atteindre son but ou effectuer des tâches (et d'adopter les comportements appropriés).

Actif

- Un agent est toujours actif. Il s'exécute donc nécessairement dans un *thread* ou un *process* indépendants.

Apprentissage

- Un agent est capable d'apprendre et d'évoluer en fonction de cet apprentissage.
- Il est capable de changer de comportement (en fonction des expériences passées).

2.2. Typologie des agents

Il existe deux grandes écoles de pensée dans la communauté des agents : l'école cognitive qui conçoit les agents comme des entités intelligentes et l'école réactive qui conçoit les agents comme des entités très simples réagissant directement aux modifications de l'environnement

2.2.1. Agents réactifs

Comme son nom l'indique, un agent réactif ne fait que réagir aux changements qui surviennent dans l'environnement. Autrement dit, un tel agent ne fait ni délibération ni planification, il se contente simplement d'acquiescer des perceptions et de réagir à celles-ci en appliquant certaines règles prédéfinies. Étant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement (B. Chaib-draa, 2010).

2.2.2. Agents cognitifs (délibératifs)

Les agents délibératifs sont des agents qui effectuent une certaine délibération pour choisir leurs actions. Une telle délibération peut se faire en se basant sur les buts de l'agent ou sur une certaine fonction d'utilité. Elle peut prendre la forme d'un plan qui reflète la suite d'actions que l'agent doit effectuer en vue de réaliser son but (B. Chaib-draa, 2010).

Comparaison entre les deux types d'agents :

Agent cognitifs	Agent réactif
Représentation explicite de l'environnement	Pas de représentation explicite de l'environnement
Peut tenir compte de son passé	Pas de mémoire de son historique
Agent complexe	Fonction stimulus/action
Petit nombre d'agent	Grand nombre d'agent

Tableau II-1. Comparaison entre les deux agents.

2.3.L'interaction entre les agents

Il existe différents types d'interactions que les agents peuvent utiliser comme la coordination, la coopération et la communication...

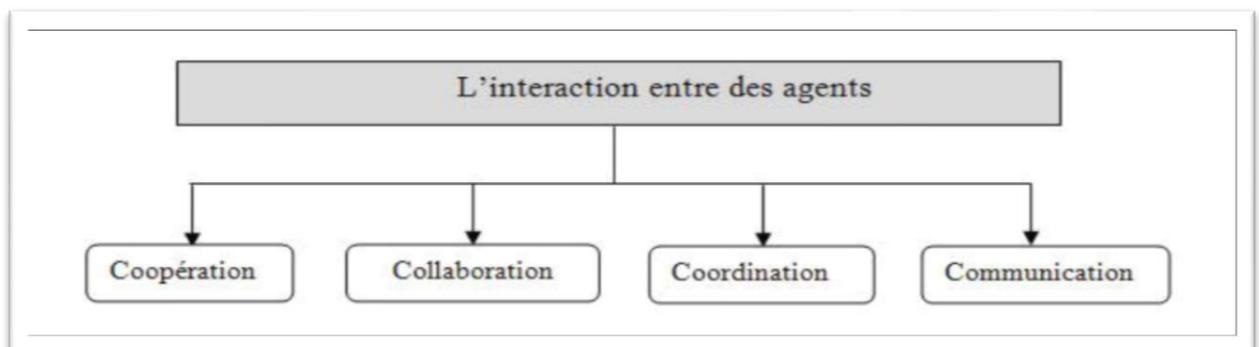


Figure II-2 : différents types d'interactions entre les agents

Nous présentons ci-dessous, quelques formes et situations importantes connues et exploitées pour l'interaction (Nadia BEN SEGHIR, 2009).

2.3.1. La Coopération

La coopération est nécessaire quand un agent ne peut pas atteindre ses buts sans l'aide des autres agents. Cette situation est fréquente même chez des espèces primitives, édification d'une fourmilière par exemple. Souvent les buts nécessitant la coopération sondes buts sociaux, ils assurent la survie du groupe ou de l'espèce. Quelquefois, ce sondes buts individuels, un agent qui en aide un autre peut attendre une aide en retour Ouse faire payer son travail. Un agent peut avoir besoin d'un autre agent parce que cet agent a des compétences qu'il n'a pas, ou parce qu'il faut être plusieurs pour réaliser la tâche(pousser un objet trop lourd). La coopération n'est pas forcément consciente, elle peut résulter d'un comportement automatique (la construction des ruches ou des termitières).

2.3.2. La Collaboration

La coopération, qui s'exprime par le travail collectif d'une équipe d'agents pour satisfaire un objectif global, nécessite la décomposition de la tâche globale en dessous tâches complémentaires, suivie par leur répartition et leur allocation aux agents, tout en considérant les compétences des différents agents et les ressources disponibles dans le système. Le concept de collaboration couvre ces deux aspects : décomposition et répartition. Elle donne réponses aux questions : *qui doit faire quoi ? Avec quel moyen ?*

Les études sur la collaboration dans les systèmes multi-agents se focalisent sur l'aspect répartition (allocation). La phase de décomposition, à cause de sa complexité et des limites des SMA actuels, est assurée préalablement par le développeur du SMA.

2.3.3. La Coordination

La coexistence des agents dans un environnement où les ressources sont limitées, et où les objectifs locaux des différents agents peuvent se contredire provoquant l'apparition de situations de conflits qui influencent sur le rendement global de tout le système, et ainsi diminuent les avantages de coopération. Pour vaincre ces situations de conflits et d'en sortir ou d'en éviter, les agents sont amenés à exécuter des actions supplémentaires (or que les actions productives), ces dernières sont dites actions coordinatrices. Le concept décooordination regroupe l'ensemble d'outils et méthodes qui peuvent être employés pour

résoudre des conflits (dus aux ressources partagées limitées ou aux objectifs incompatibles) ou pour optimiser des comportements (éliminer des actions redondantes et inutiles) et plus généralement pour assurer un tout cohérent.

2.3.4. La Négociation

La négociation joue un rôle fondamental dans les activités de coopération. En général, une négociation intervient lorsque des agents interagissent pour prendre des décisions communes, alors qu'ils poursuivent des buts différents. Plus précisément, l'objectif de la négociation est de résoudre des conflits qui pourraient mettre en péril des comportements coopératifs. Il y a d'abord un échange de points de vue, généralement divergents, puis un ajustement réciproque (concessions ou propositions alternatives) pour obtenir le compromis attendu. Le processus de négociation permet d'améliorer les accords (en réduisant les inconsistances et l'incertitude) sur des points de vue communs ou des plans d'action grâce à l'échange structuré d'informations pertinentes (François Bourdon et Patrice Enjalbert, 2000)

2.3.5. La Communication

La communication est l'un des concepts pertinents dans les systèmes multi-agents. Comme chez les humains, la communication est à la base de l'interaction et de l'organisation sociale, elle permet aux agents de coopérer, négocier, échanger des informations et effectuer des tâches en commun. Sans communication, l'agent n'est qu'un individu isolé sourd et muet aux autres agents. On distingue essentiellement deux modèles de communication :

- Communication par partage d'information.
- Communication par envoi de messages.

• Communication par partage d'information :

La communication entre les différents agents du système est réalisée par partage d'information lorsque ceux-ci disposent d'une zone de données commune dans laquelle ils rangent les conclusions qu'ils ont pu tirer. Outre ces résultats partiels, cette zone renferme les données du problème initial. Les agents peuvent ainsi y puiser les informations dont ils ont besoin pour résoudre une partie du problème globale.

Cette zone est le moyen d'échange d'information entre les agents. Ils la consultent pour se faire une idée de l'état d'avancement de la construction de la solution et voir ainsi si

la situation leur est favorable, si leur raisonnement est opportun ou pas. Les agents n'ont alors pas besoin de se connaître mutuellement.

Ce type de communication correspond à ce que la littérature désigne communément sous le nom de modèle du blackboard, Le black board correspond donc à la zone de données commune. La figure présenter structure d'un système de tableau noir :

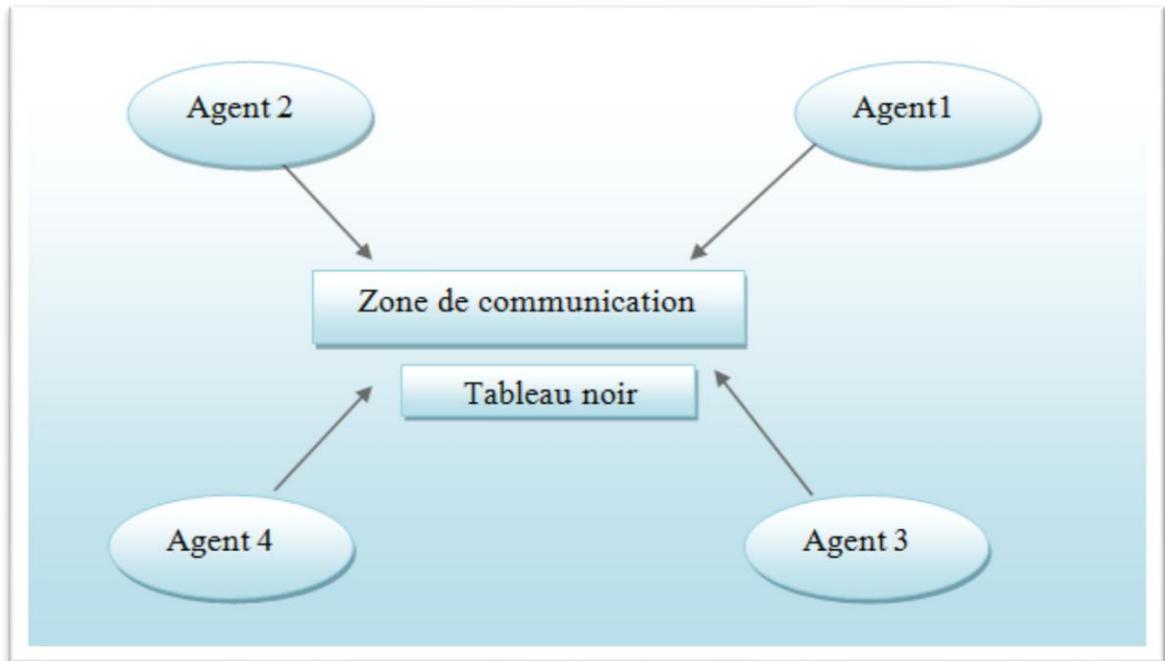


Figure II-3 : *La communication par tableau noir.*

• **Communication par envoi de message :**

- Une autre organisation possible des agents peut consister à ce que chacun ait sa propre base de faits locale, où ses propres résultats, représentant une partie de la solution courante au problème global, sont stockés. Un agent peut alors, lorsqu'il le juge utile, entrer en communication directe avec un autre agent en lui transmettront de l'information via des messages de la même façon que des échanges postaux peuvent se faire. Ces messages peuvent comporter des demandes de renseignements ou encore fournière, spontanément ou suite à une demande antérieure, des informations, il y a échangé des résultats partiels.

- Ce type de communication suppose l'utilisation d'un protocole pour les échanges. Il est utilisé dans les modèles d'acteurs ou de société de spécialistes (Darmont J, 2000).

Nous avons utilisé cette type de communication, pour envoyer et recevoir les requêtes entre les agents de même type.

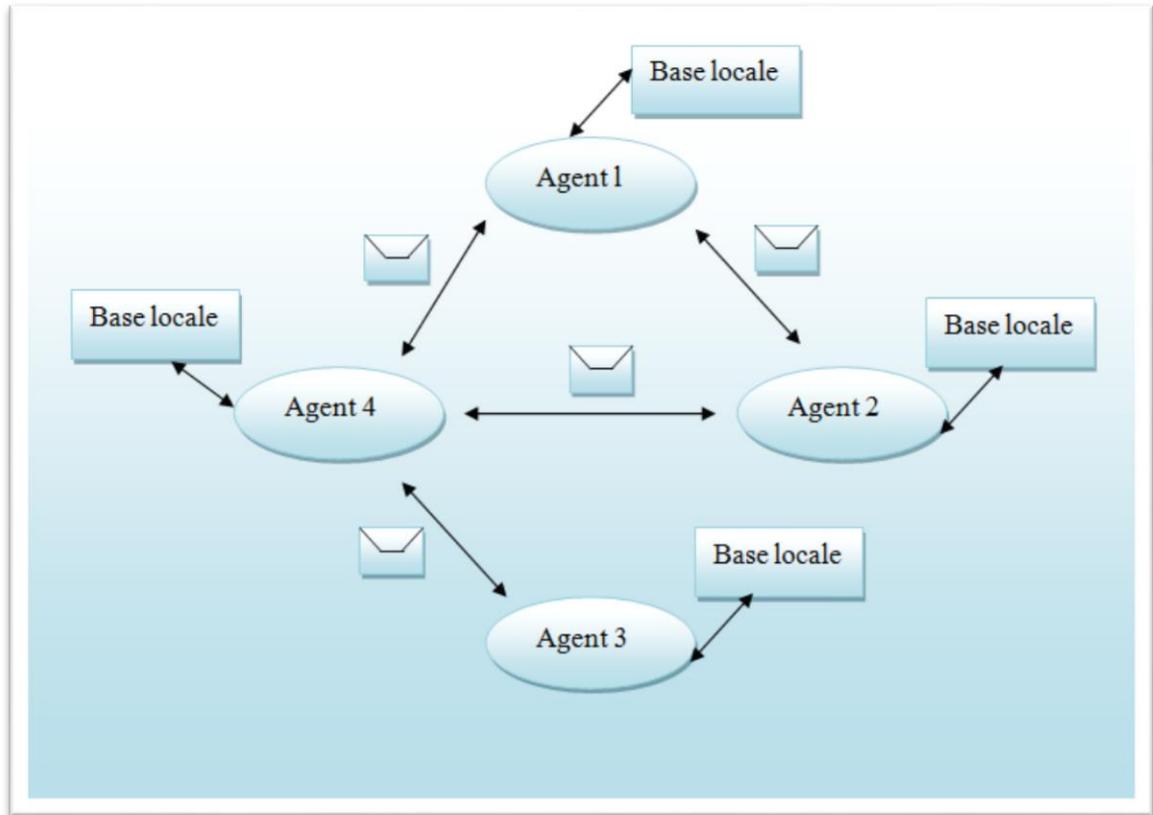


Figure II-4 : *La communication par envoi de message.*

3. LES SYSTEMES MULTI-AGENTS (SMA):

La définition d'un système multi-agents (SMA) est plus immédiate : « un système multi-agent est un ensemble organisé d'agents ». Cela signifie que dans un système multi-agent, il existe une ou plusieurs organisations qui structurent les règles de cohabitation et de travail collectif entre agents (définition des différents rôles, partages de ressources, dépendances entre tâches, protocoles de coordination, ...). Dans un même système, il existe en général plusieurs organisations et un même agent peut appartenir à plusieurs simultanément. Des exemples d'organisations d'agents dans le monde réel sont une organisation économique telle qu'une entreprise, mais aussi une organisation animale telle qu'une fourmilière. Suivant les cas, les comportements des agents sont plus ou moins complexes et rationnels et l'organisation est plus ou moins adaptative.

Les agents sont en général situés dans un environnement (par exemple, topologique) contenant également des entités passives, manipulées par les agents (par exemple, des ressources, des données, des objets physiques), et communément appelées objets. Chaque agent n'a qu'une connaissance partielle de son environnement et des autres agents. Un système multi-agent est donc intrinsèquement décentralisé (Yves, 2001).

Les systèmes multi-agents sont à l'intersection de plusieurs domaines scientifiques : informatique répartie et génie logiciel, intelligence artificielle, vie artificielle. Ils s'inspirent également d'études issues d'autres disciplines connexe notamment la sociologie, la psychologie sociale, les sciences cognitives et bien d'autres. C'est ainsi qu'on les trouve parfois à la base des (Boissier, 2004)

- Systèmes distribués;
- Bases de données et bases de connaissances distribuées coopératives;
- Protocoles de communication et réseaux de télécommunications;
- Programmation orientée agents et génie logiciel;
- Robotique cognitive et coopération entre robots;
- Applications distribuées comme le web, l'Internet, le contrôle de trafic routier, le contrôle aérien, les réseaux d'énergie, etc.

4. LES AGENTS INTELLIGENTS :

4.1.Définition :

Le terme « agent intelligent » désigne un certain nombre d'applications fonctionnant à la fois dans un environnement Internet et sur d'autres environnements comme les systèmes d'exploitation. La définition du terme reste très générale. Elle se réfère à une partie d'un système informatique (programme, code, crawler, spider) ou au système complexe lui-même (méta-moteur en ligne, comparateur de prix, logiciel).

Si on recherche des définitions aujourd'hui sur Internet, on peut se servir de Google. Le moteur nous présente une liste dont les entrées mettent l'accent sur la récupération de documents. C'est cette fonction qui semble être la plus courante. Cependant le terme couvre un plus grand champ d'applications (John Richard, 2005) .

L'association française de normalisation (AFNOR) définit un agent intelligent ainsi:

« Objet utilisant les techniques de l'intelligence artificielle : il adapte son comportement à son environnement et en mémorisant ses expériences, se comporte comme un sous-système capable d'apprentissage : il enrichit le système qui l'utilise en ajoutant, au cours du temps, des fonctions automatiques de traitement, de contrôle, de mémorisation ou de transfert d'informations. » (John Richard, 2005).

Une deuxième définition pour le terme agents intelligents :

«Programme utilitaire qui appartient à une classe d'application et qui fonctionne de façon indépendante, ou encore, qui sillonne un réseau pour en faciliter l'exploitation» (Lambert & Médaille & Welsh, 2006)

Et voici la troisième pour mieux comprendre :

«Un agent est dit intelligent parce qu'il intègre les techniques de l'intelligence artificielle et que, en théorie, il est en mesure d'adapter son comportement à son environnement, grâce à ses capacités d'apprentissage et de mémorisation de ses expériences. Capable d'assumer des fonctions automatiques de contrôle, de mémorisation ou de transfert d'informations, son utilisation est de plus en plus répandue, dans des applications fort variées allant de la simple recherche d'informations à la simulation, en passant par l'aide à la décision ,le routage d'informations dans les réseaux, le courrier électronique, etc. Dans les faits, la plupart des agents actuellement sur le marché ont des capacités beaucoup plus réduites et sont d'avantage capables d'actions répétées que de véritables raisonnements. » (Lambert & Médaille & Welsh, 2006).

4.2.Les agents et le web :

On peut se demander pourquoi l'utilisateur a besoin d'agents intelligents. La croissance phénoménale d'Internet en nombre de sites et de documents nécessite le développement de programmes pour gérer le flux des informations et la recherche documentaire. Au début d'Internet, le moyen le plus utilisé pour trouver un document était le surf, c'est-à-dire une forme de navigation qui consiste à explorer Internet par le biais d'hyperliens. Or cette méthode s'avère de plus en plus inadéquate étant donné la quantité d'informations disponibles sur le Web. D'où la nécessité d'utiliser des agents d'Internet. Or, après avoir observé les usagers, on constate que très peu parmi eux se servent d'agents logiciels téléchargés. La majorité préfère passer par un moteur de recherche ou un portail. Cependant, les fonctionnalités intégrées dans certains agents logiciels réduisent le bruit en filtrant les résultats proposés, ce qui allège en partie la surcharge informationnelle que l'utilisateur doit gérer. Si en amont, les moteurs de recherche semblent avoir réussi à réduire le bruit et à présenter une liste de résultats satisfaisants après quelques itérations, la gestion de l'information sur le PC de l'utilisateur n'est pas au point pour l'instant. A notre avis l'un des enjeux économiques consiste à faire installer sur l'ordinateur de l'utilisateur un logiciel qui fonctionne en étroite liaison avec un moteur de recherche. Ainsi deux des trois points stratégiques (PC, Moteur, site ciblé) seront liés en permanence. L'utilisateur dispose de quatre

grandes classes d'outils de recherche documentaire sur Internet. Il peut les choisir librement pour l'instant car la plupart sont gratuits. Le premier choix des internautes est de loin le moteur de recherche. L'annuaire paraît moins utilisé en tant que tel, mais il peut contenir un moteur de recherche fourni par un partenaire, comme l'atteste l'exemple de Yahoo. Le méta-moteur en ligne (considéré par certains comme un agent intelligent en ligne) peut être consulté. Les agents logiciels téléchargés sur le disque dur de l'internaute demeurent moins connus par le grand public et constituent des outils destinés essentiellement aux professionnels de la veille. L'agent est soit un logiciel, soit un méta-moteur en ligne, soit un programme intégré dans un système de recherche. L'usage a donné à ce terme une polysémie regrettable. On peut constater que les agents se connectent à l'ensemble des ressources qui structurent l'organisation de l'information, de la connaissance et du savoir sur Internet. Plus ils sont perfectionnés, plus ils comportent des langages permettant d'interroger avec précision les autres outils et sources d'information. Cependant, utiliser un agent, le paramétrer et en exploiter les fonctionnalités représente un coût en termes de temps et d'efforts d'apprentissage.

4.3. Les caractéristiques des agents intelligents :

Dans la section définition nous avons posé que l'intelligence était basée sur un agrégat de caractéristiques (la capacité d'apprendre, la capacité sociale et une haute autonomie) chacun pouvant avoir des degrés divers.

- ✓ La capacité d'apprendre : Le petit Larousse définit «apprendre» (du latin appréhendée, saisir) comme le fait «d'acquérir la connaissance, l'information, l'habitude ». Un Agent aura la capacité d'apprendre s'il sait acquérir de la connaissance, de l'information ou des habitudes.

Exemple : Un Agent grâce à sa capacité de réactivité, doit se déclencher à une certaine heure. Mais l'utilisateur l'arrête dans sa tâche (qui pourrait être une sauvegarde) car il ralentit le travail de l'utilisateur. L'Agent va apprendre à différer son exécution pour éviter de gêner l'utilisateur.

- ✓ La capacité sociale : Les Agents interagissent avec les autres Agents (et éventuellement des êtres humains) grâce à des langages de communication entre Agents. Cette capacité sera la base pour la coopération entre les Agents

Exemple 1 : Notre Agent de sauvegarde rencontre un autre Agent de sauvegarde sur un réseau. Ces deux Agents peuvent se mettre d'accord pour se partager le travail afin que la tâche soit achevée plus vite.

Exemple 2 : L'Agent ne peut exécuter sa tâche pour des raisons techniques diverses (problèmes de droits d'accès, par exemple). L'Agent va communiquer ce fait à l'utilisateur, en lui apportant la démarche à suivre dans un langage naturel compréhensible. L'utilisateur pourra ensuite, en utilisant le même langage, indiquer à l'Agent comment résoudre le problème.

- ✓ Haut degré d'autonomie : L'Agent fonctionne sans intervention directe humaine ou autre autonomie) et en plus il a une forme de contrôle sur ses actions et sur leur état interne (haut degré).

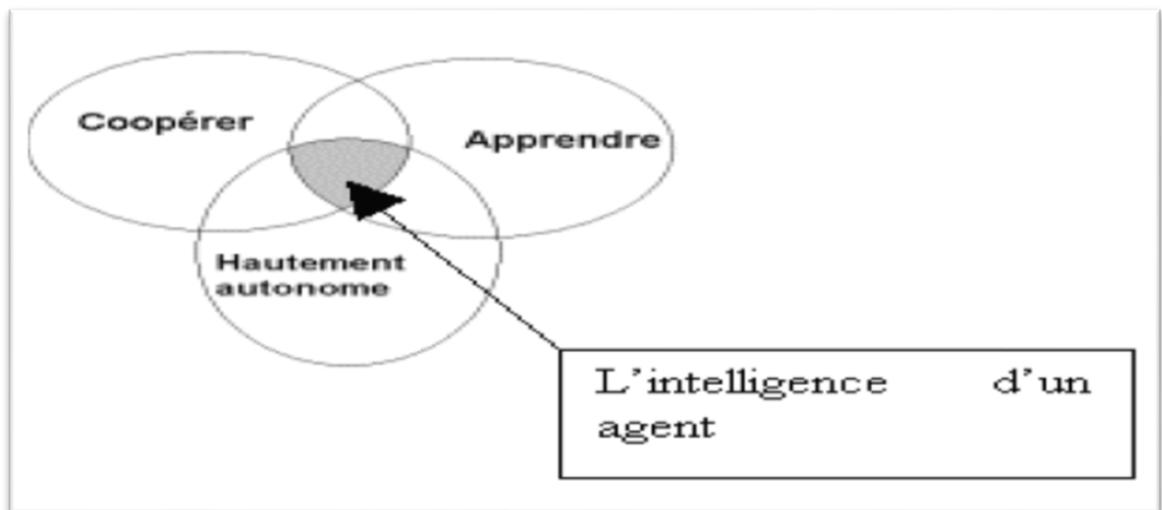


Figure II-5 : Schéma de la représentation de l'intelligence pour un Agent.

- ✓ Un peu plus d'intelligence : La pro-activité est une caractéristique qui est intégrée dans les Agents hautement intelligents (très rares actuellement). Les Agents n'agissent pas seulement en réponse à leur environnement, mais ils sont capables d'avoir un comportement guidé par un but, en ayant la possibilité de prendre l'initiative.

Exemple : Un Agent réseau peut décider, de lui-même, pendant un temps où il est non actif, de faire des statistiques sur les routeurs pour améliorer son activité future (DUTIL, 2009).

4.4. Les Caractéristiques additionnels :

Les caractéristiques suivantes peuvent être ajoutées aux caractéristiques de base pour permettre à l'Agent d'exécuter sa tâche. Il s'agit ici d'une liste non exhaustive, d'autres caractéristiques peuvent s'ajouter et se combiner.

- ✓ **Coordination** : L'Agent est capable de coordonner ses actions par rapport à un utilisateur ou un autre Agent.

Exemple : Un Agent de sauvegarde sauve les données qui viennent d'être mises à jour par un utilisateur ou par un autre Agent.

- ✓ **Compétitif** : L'Agent est capable d'agir dans un environnement où d'autres Agents interviennent. Le but est le même pour tous les Agents présents, mais un seul l'atteindra. Les autres échoueront et, forcément, tous les coups sont permis.

Exemple : Un Agent commercial peut chercher les meilleurs prix et les meilleurs services pour un produit donné. En négociant avec les fournisseurs, d'une manière plus rapide et plus optimale qu'un autre Agent ou utilisateur. Un Agent très compétitif pourra induire ses «adversaires» sur de fausses pistes.

- ✓ **Mobilité** : Les agents doivent pouvoir être multi-plate-forme et multi-architecture. Ils doivent pouvoir se déplacer sur le réseau où ils accomplissent des tâches sans que l'utilisateur ait le moindre contrôle sur celles-ci.

Afin d'éclaircir la définition de mobilité et dissiper nombre de confusions, nous devons prendre en compte deux types de mobilités :

- ✓ **Mobilité relative, ou par requêtes** : Dans ce cas il n'y a pas un réel déplacement de l'Agent. Celui-ci lance une succession de requêtes à destination de différents serveurs. C'est le cas des Agents de recherche, tel que Copernic, qui interroge différents moteurs de recherche afin de fournir à son utilisateur une synthèse des résultats.

- ✓ **Mobilité réelle de l'Agent** : Le processus Agent se déplace d'un serveur à un autre, sur le réseau. Le code de l'objet est transporté et ses données, aussi. Ensuite, il continue son exécution sur la nouvelle machine.

Exemple : Un Agent de sauvegarde peut se déplacer sur plusieurs serveurs dans le but de faire des restaurations de fichiers (DUTIL, 2009).

Quand nous parlerons d'Agent mobile, c'est ce dernier cas qui sera pris en compte.

4.5. Classification des agents intelligents :

On peut regrouper les agents intelligents en fonction de l'environnement, de la dichotomie PUSH/PULL, ou CLIENT/SERVEUR, ou par rapport à leurs tâches. Celles-ci sont définies par leurs concepteurs.

4.5.1. Classification selon la plate-forme d'informations :

L'environnement désigne la plate-forme sur laquelle les agents sont installés. On énumère ainsi trois grandes classes d'agents intelligents: ceux du PC, d'Internet et les agents d'intranet. Les agents du PC (desktop agents) s'imbriquent dans le système d'exploitation de l'ordinateur (Windows, Mac OS, etc.). Ils sont connus sous le nom d'assistants. Leur fonction est d'aider l'utilisateur dans l'exécution d'une tâche automatisée comme installer un nouveau logiciel ou un nouveau périphérique en (plug and play). Les agents d'intranet (Intranet agents) permettent de récupérer des ressources sur les serveurs de l'entreprise. Les agents d'Internet (Internet agents) aident l'utilisateur à rechercher des documents sur Internet. Les moteurs et méta-moteur font partie de cette catégorie.

4.5.2. Classification PUSH/PULL :

Une seconde distinction se fait entre les agents du type PULL et ceux du type PUSH. Le verbe anglais pull signifie tirer vers soi, tandis que le mot push veut dire envoyer vers quelqu'un d'autre. L'utilisateur initie la requête en paramétrant un agent à partir de mots-clés ou d'expressions plus complexes. Il délègue la tâche à un agent PULL comme un moteur ou méta-moteur. Cependant, l'utilisateur interagit avec le programme en transformant la requête en fonction des résultats retournés.

Au contraire, avec la technologie PUSH, l'utilisateur est passif. Les logiciels de type PUSH permettent d'accéder à des chaînes d'information thématiques ou d'actualité, comme les chaînes de télévision ou de journaux. Les informations sont envoyées régulièrement aux abonnés en fonction de leur profil ou de leurs centres d'intérêts définis au préalable ou appris grâce à la vigilance du programme informatique. PointCast, Marimba et BlackWeb constituent des exemples de logiciels de PUSH. Le moteur Google (news alerts) et certains méta-moteurs comme Copernic ont intégré ce type d'activité.

Les agents de diffusion sélective font partie de la technologie PUSH. Leur but est de trouver pour l'utilisateur les informations susceptibles de l'intéresser sans qu'il en ait fait la demande. Le système fonctionne de la manière suivante. L'utilisateur souscrit un contrat

avec un fournisseur d'informations qui lui envoie un choix de documents en fonction de ses goûts et intérêts. Un agent intelligent choisit automatiquement les documents et établit une liaison entre l'agent installé sur l'ordinateur de l'utilisateur et celui du serveur du fournisseur. Les journaux américains comme The New York Times offrent ce type de service.

Les agents d'alerte et de veille surveillent une source d'information ou un thème pour prévenir l'utilisateur en fonction d'une requête prédéfinie. Ils envoient des messages par courrier électronique lors d'un changement de contenu. Parmi ce type d'agent, nous rencontrons Url-Minder et NetMindHighlighter. Nous avons constaté que les moteurs de recherche intègrent aussi ces fonctions. D'autres agents PULL surveillent les archives des forums de discussion et les listes de diffusion.

4.5.3. Classification Client / Serveur:

La dichotomie client/serveur permet également une classification. Certains agents peuvent être téléchargés et installés sur le disque dur de l'utilisateur. D'autres, au contraire, opèrent à partir du serveur du propriétaire de l'agent, et fonctionnent en mode client-serveur. L'internaute se connecte et utilise les services de la technologie mise à sa disposition gratuitement. L'avantage des agents du côté client réside dans leur paramétrage

Plus poussé par rapport aux agents localisés sur un serveur à distance. Notons que si les premiers sont téléchargeables gratuitement dans un premier temps, les derniers restent toujours disponibles et gratuits. Le financement du site et du développement se fait par le biais d'autres types de modèles économiques. Nous avons observé que les moteurs de recherche proposent un dispositif du type agent client (sur le disque dur) pour étendre progressivement leur action vers l'utilisateur. Ainsi on combine les avantages des deux types d'interactions : sur le serveur et chez le client.

4.5.4. Classification par Tâches :

Le dernier mode de classement décrit les usages prévus par les éditeurs de logiciels : les tâches accomplies par l'agent. Passons en revue ces divers types de programme en précisant que cette liste n'est pas exhaustive.

- ❖ Un agent de filtrage : est conçu pour examiner des courriers reçus et détruire les e-mails non désirés sur la messagerie, éliminer les informations non pertinentes d'une

requête, chercher et préparer des informations à partir de diverses sources. (Quelques exemples : NewsHound, ZDNetpersonalView ou NewsPage Direct.)

- ❖ Un agent aspirateur : (retrieval agent) télécharge un site entier sur le disque dur, facilitant ainsi l'analyse de son contenu hors-ligne.
- ❖ Les agents avertisseurs : ou d'alerte hors-ligne (notifiers) préviennent l'utilisateur lorsqu'un site change, lorsqu'une information importante arrive ou qu'un événement important se produit.
- ❖ Les agents de recherche : (search agents) identifient des informations pertinentes sur Internet en relation avec un ensemble de moteurs et en fonction des préférences des usagers. Ils peuvent intégrer un module d'apprentissage. Ce que l'on nomme ici agent fait l'objet d'un ensemble de fonctionnalités introduites dans les progiciels d'agents intelligents du type méta-moteur.
- ❖ Des agents livreurs d'informations : hors-ligne : (delivery agents) qui, comme leur nom l'indique, envoient des informations personnalisées aux usagers sur leur disque dur. La connexion n'est pas établie pendant la lecture ou la consultation des documents reçus, ce qui permettait auparavant de diminuer la consommation de bande passante. Aujourd'hui, ce problème est moins important grâce au haut débit et aux abonnements illimités.
- ❖ Les agents d'achat : ou comparateurs de prix (shopping agents) facilitent la recherche des meilleurs prix pour un produit donné. Ils ne sont pas forcément appréciés par les propriétaires des sites Web commerçants et leur entrée dans un site peut être interdite. La concurrence pure et parfaite sur Internet ne fait pas l'unanimité. Ce type de technologie intègre progressivement les sites des moteurs de recherche ou des portails.
- ❖ Les agents de bavardage : (chatterbots) s'avèrent capables de s'entretenir avec un usager. Le premier de ce type, Eliza, se comportait comme une psychanalyste qui posait des questions à son patient pour le faire parler. Depuis, on conçoit les chatterbots pour introduire une forme de dialogue et d'interactivité sur certains sites Web commerciaux. Le bot peut répondre à des questions posées par l'internaute sur un produit ou un service ou envoyer des fiches techniques en fonction des mots-clés repérés dans l'énoncé du visiteur.

- ❖ Les agents de petites annonces : (classified agents) examinent les offres de produit dans des bases de données en fonction du sujet ou du domaine spécifié par l'internaute. Ils envoient les résultats par e-mail.
- ❖ L'agent pense-bête :(announcement agent) a pour mission de rappeler à l'utilisateur les événements ou les rendez-vous importants. Ce type d'agent peut s'installer en ligne ou sur le disque dur d'un PC.
- ❖ D'autres types d'agent : existent, certains spécialisés dans les livres (book agents) et qui cherchent les nouveautés dans le monde de la publication en fonction des préférences des usagers. Les agents de suivi du monde des affaires (business information monitoring agents) filtrent l'actualité économique, les publications et rapports mis sur Internet par les entreprises. Les agents de services financiers personnalisés (financial service agents) apportent des informations financières en fonction du portefeuille personnel de l'utilisateur. Enfin, les agents de recrutement (job agents) cherchent les profils d'éventuels candidats pour un emploi en fonction des CV mis en ligne. (John Richard, 2005).

4.6.Le fonctionnement d' un Agent intelligent :

Nous avons posé dans la section définition qu'un agent était intelligent, s'il avait l'ensemble de caractéristiques suivantes : une capacité sociale, une capacité à apprendre et un haut degré d'autonomie. Pour permettre cela, il faut intégrer au programme agent les modules suivant :

-Un module permettant de mémoriser le savoir de l'agent qui sera le plus souvent fait par la réalisation d'un système de base de données (SGBD). Ce module sera nommé « base de connaissance ».

-Un module permettant de connaître l'ontologie de la « base de connaissance ». Il sera appelé « Le contrôleur d'accès » qui peut reposer sur un SGBD, dit « intelligent ».

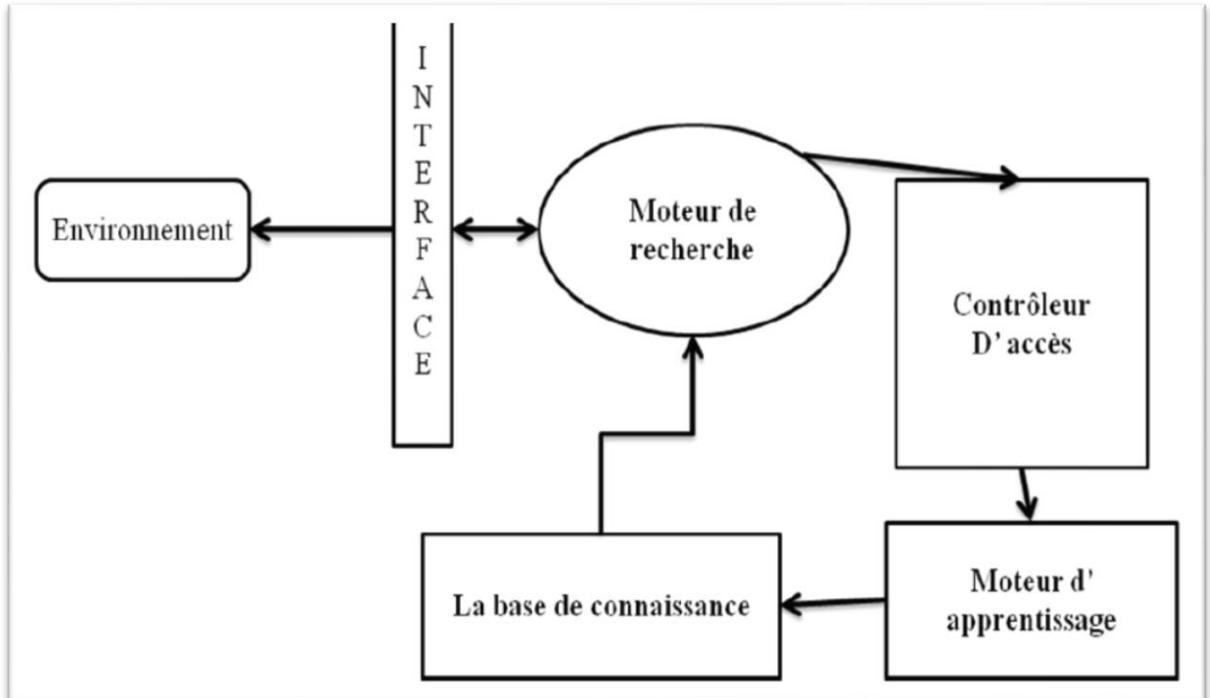


Figure II-6 : Schématisation du mécanisme de l'intelligence chez un agent.

Nous pouvons dès lors, faire la distinction entre quatre groupes :

- Le contenu représente les données employées par les mécanismes dans le raisonnement de l'apprentissage (représenté sur le schéma par la base de connaissance).
- Les mécanismes de l'agent contenant les moteurs d'apprentissage et de réflexion.
- L'accès qui permet de percevoir le contenu des connaissances de l'agent.
- L'interface permettant une perception de l'environnement de l'agent.

4.7. Quelques agents intelligents sur le web :

Parmi les agents intelligents de RI sur internet on peut citer quelque uns (WebSeeker ,FirstStopWebSearch , Internet Search ActiveX 1.0 , EasySeeker, Nick's Pics, Search Master, PR infoFinder , HtmlSearch Lite , SharpewareWebStorm, LanHunter...etc.)

4.7.1. WebSeeker :

WebSeeker combine les résultats de nombreux moteurs de recherche. Il présente une liste claire des résultats qui peut être sauvegardée et consultée hors ligne. La liste peut être organisée facilement et mise à jour automatiquement.

4.7.2. Quelques Fonctions :

- ☞ Utilise plus de 100 moteurs de recherche (Yahoo, InfoSeek, Lycos, Alta Vista etc.).
- ☞ Supprime les liens cassés et les doublons.
- ☞ Accepte les requêtes booléennes et les phrases.

- 👉 Ajout de moteurs de recherche.
- 👉 Surveille les changements des pages Internet.
- 👉 Programmation de recherche automatique.
- 👉 Mise à jour automatique à travers Internet (Agentland, 2009)

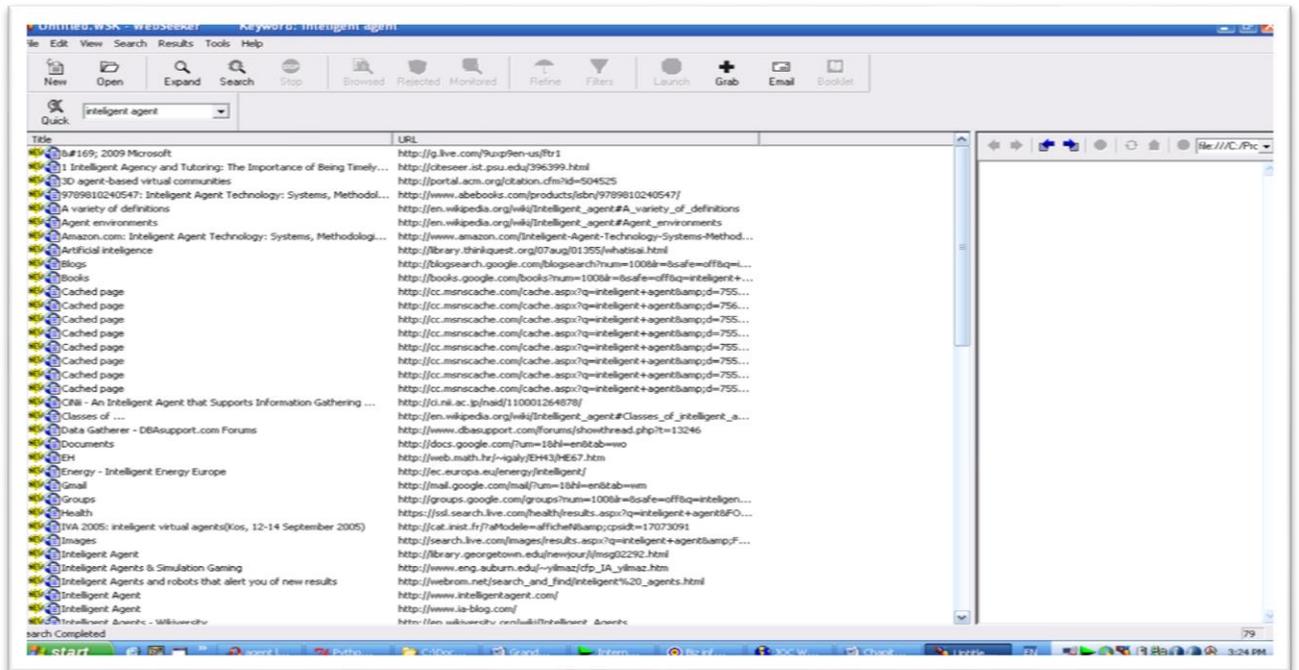


Figure II-7 : Capture d'écran de l'interface du webseeker

5. CONCLUSION :

Nous avons présenté dans ce chapitre en premier lieu une vision générale sur les agents et les systèmes multi-agents, ces systèmes qui ont porté résolution aux problèmes de l'IA classique sont organisés dans des sociétés d'agents qui interagissent, communiquent et coopèrent entre eux pour accomplir une tâche bien déterminée.

Après ces notions du base précédentes, nous avons passé à l'étude d'une nouvelle technologie, c'est la technologie d'agents intelligents. Les agents intelligents sont des systèmes informatiques intégrés à un environnement complexe et dynamique. Il analyse et agit en fonction de l'environnement et des objets à atteindre. En effet, un agent intelligent modifie son comportement en fonction de l'environnement et il est capable d'anticiper, autrement dit il est proactif.

Dans le chapitre suivant, on va voir une autre nouvelle notion, c'est les ontologies, qui sont aussi utilisées pour l'amélioration des moteurs de recherche.

Chapitre III

LES ONTOLOGIES ET

LES MOTEURS DE RECHERCHE

Dans ce chapitre :

- ❖ Introduction
- ❖ L'ontologie
- ❖ Le Web Sémantique
- ❖ Les ontologies et le web sémantique
- ❖ Conclusion

1. INTRODUCTION

Nées des besoins de représentation des connaissances, les ontologies sont à l'heure actuelle au cœur des travaux menés en Ingénierie des Connaissances (IC). Le terme « ontologie » est utilisé depuis le début des années 1990, et son champ d'application s'élargit considérablement. Un des plus grands projets basés sur l'utilisation d'ontologies consiste à ajouter au Web une véritable couche de connaissances permettant des recherches d'informations au niveau sémantique. A terme, il est prévu que des applications internet pourront mener des raisonnements en utilisant les connaissances stockées sur la Toile.

Au fur et à mesure des expérimentations, des méthodologies de construction d'ontologies et des outils de développement adéquats sont apparus. L'enjeu de l'effort engagé est de rendre les machines suffisamment sophistiquées pour qu'elles puissent intégrer le sens des informations(**R.Lekhchine, 2009**).

Dans ce chapitre, nous relèverons les différentes définitions qui ont été attribuées à la notion d'ontologie. Nous montrerons aussi l'objectif et la topologie de l'utilisation des ontologies dans le domaine du Web sémantique ainsi que leur place dans les systèmes à bases de connaissances. Ensuite, nous montrerons la construction d'une ontologie et le cycle de vie d'une ontologie et la notion web sémantique et la relation des ontologies avec le web sémantique. Finalement, nous présenterons la recherche d'information guidée par les ontologies dans le web sémantique et les ontologies dans les moteurs de recherche sémantique.

2. L'ONTOLOGIE :

2.1. Notion d'ontologie

Introduit en Intelligence Artificielle (IA) il y a 23 ans, le terme d'ontologie est cependant usité en philosophie depuis le XIX^{ème} siècle. Dans ce domaine, l'ontologie est une étude de l'être en tant qu'être, c'est-à-dire, une étude des propriétés générales de ce qui existe. C'est à l'occasion de l'émergence de l'Ingénierie des Connaissances que les ontologies sont apparues en IA, comme réponses aux problématiques de représentation et de manipulation des connaissances au sein des systèmes informatiques (R.Lekhchine, 2009).

2.2. Définitions

Le terme « ontologie » est employé dans des contextes très différents touchant la philosophie, la linguistique ou l'IA. De nombreuses définitions ont été offertes pour donner un éclaircissement sur ce terme, mais aucune de ces définitions ne s'est explicitement imposée. Les définitions de ce terme ne sont pas toujours consistantes et cela dépend des domaines spécifiques (McGuinness, 2001). Pour ne pas dévier de notre propos, nous avons recensé les définitions suivantes :

Définition1 « une ontologie définit les termes et les relations de base du vocabulaire d'un domaine ainsi que les règles qui indiquent comment combiner les termes et les relations de façon à pouvoir étendre le vocabulaire » (R. Neches, 1991).

Définition2 « une ontologie est une spécification explicite d'une conceptualisation » La définition de Gruber est la plus utilisée dans la littérature. Elle a été légèrement modifiée (Gruber T. , 1993).

Définition3 « une ontologie est une spécification explicite et formelle d'une conceptualisation partagée » Le terme « conceptualisation » réfère dans cette définition à une abstraction d'un phénomène du monde, obtenue en identifiant les concepts appropriés à ce phénomène. Le terme « Formelle » indique que les ontologies sont interprétables par la machine. Cependant, « Spécification explicite » signifie que les concepts de l'ontologie et les contraintes liées à leur usage sont définis de façon déclarative. Enfin, le terme « partagé » signifie que l'ontologie capture la connaissance consensuelle. Mais cette définition laisse la porte ouverte à de nombreuses définitions. (Borst, 1997).

Définition5 « Les ontologies sont des spécifications partielles et formelles d'une conceptualisation commune » (Guarino, Understanding, building, and using ontologies,

1997.).En 1997, Guarino accentue l'ambiguïté du terme conceptualisation qui doit être pris dans son sens intuitif. La spécification des ontologies est partielle, car une conceptualisation ne peut pas toujours être entièrement formalisée dans un cadre logique, du fait d'ambiguïtés ou du fait qu'aucune représentation de leur sémantique n'existe dans le langage de représentation d'ontologies choisi. « Commune » renvoie à l'idée qu'une ontologie rend compte d'un savoir consensuel, c'est-à-dire qu'elle n'est pas l'objet d'un individu, mais qu'elle est reconnue par un groupe (Furst, 02-07-2002).

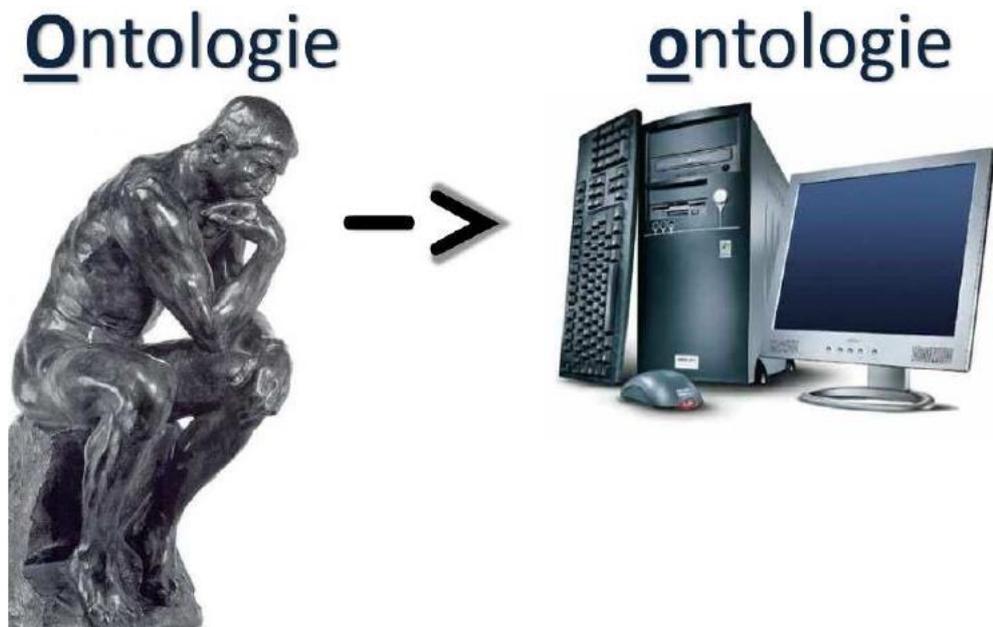


Figure III-1: *Historique de L'ontologie (informatique).*

2.3. L'objectif d'ontologie

L'objectif premier d'une ontologie est de modéliser un ensemble de connaissances dans un domaine donné, qui peut être réel ou imaginaire. Les ontologies sont employées dans l'intelligence artificielle, le Web sémantique, le génie logiciel, l'informatique biomédicale et l'architecture de l'information comme une forme de représentation de la connaissance au sujet d'un monde ou d'une certaine partie de ce monde (Y. Askane, Y. El-ouchi, 2009/2010).

2.4. Que représente-t-on dans une ontologie ?

Les ontologies produisent un vocabulaire commun d'un domaine et définissent, de façon plus ou moins formelle, la signification des termes et des relations entre eux. Les connaissances intégrées dans les ontologies sont formalisées en mettant en jeu cinq types de composants : concepts, relations, fonctions, axiomes, instances (Gomez Pérez A., 1999.).

2.4.1. Concepts : Ils sont appelés aussi termes ou classes de l'ontologie. Un concept est un constituant de la pensée (un principe, une idée, une notion abstraite) sémantiquement évaluable et communicable. L'ensemble des propriétés d'un concept constitue sa compréhension ou son intention et l'ensemble des êtres qu'il englobe, son extension. Selon (Bachimont, 2004) un concept se définit à trois niveaux : Un concept est une signification. Sa place dans un système de significations permet de le comprendre, de le distinguer et de le différencier par rapport à d'autres concepts.

Un concept est une construction. Selon (Gomez Pérez A., 1999.), ces concepts: peuvent être classifiés selon plusieurs dimensions :

- ✓ Niveau d'abstraction (concret ou abstrait).
- ✓ Atomicité (élémentaire ou composée).
- ✓ Niveau de réalité (réel ou fictif).

En résumé, un concept peut être tout ce qui peut être évoqué et peut consister en la description d'une tâche, d'une fonction, d'une action, d'une stratégie ou d'un processus de raisonnement, etc.

2.4.2. Relations : Représentent un type d'interaction, ou bien des associations existant entre les concepts d'un domaine. Elles se définissent formellement à partir d'un produit de concepts : $R : C_1 \times C_2 \times \dots \times C_{n-1} \times C_n$. sous-classe-de (Spécialisation, généralisation), partie-de (agrégation ou composition), associée-à, instance-de sont des exemples de relations binaires. Voici quelques relations les plus courantes dans la littérature : L'équivalence: une relation R est une relation d'équivalence si et seulement si : R est symétrique, réflexive et transitive. On écrit :

$(R \text{ est une relation d'équivalence}) \iff ((R \text{ symétrique}) \wedge (R \text{ réflexive}) \wedge (R \text{ transitive}))$

la cardinalité: c'est le nombre possible de relations de ce type entre les mêmes concepts (ou instances de concept). Les relations portant une cardinalité représentent souvent des attributs. Exemple : une pièce a au moins une porte, un humain a entre zéro et deux jambes.

L'incompatibilité: Deux relations sont incompatibles si elles ne peuvent lier les mêmes instances de concepts. Exemple : les relations «être rouge » et «être vert » sont incompatibles.

L'inverse: Deux relations binaires sont inverses l'une de l'autre si, quand l'une lie deux instances I1 et I2, l'autre lie I2 et I1.

Exemple : les relations « a pour père » et « a pour enfant » sont inverses l'une de l'autre .

L'exclusivité: Deux relations sont exclusives si, quand l'une lie des instances de concepts, l'autre ne lie pas ces instances, et vice-versa. L'exclusivité entraîne l'incompatibilité. Exemple : l'appartenance et la non appartenance sont exclusives. Et bien d'autres relations...

2.4.3. Fonctions : ce sont des cas particuliers de relations dans lesquelles le Nième élément de la relation est défini de manière unique à partir des n-1 premiers. Formellement, les fonctions sont définies ainsi : $F: C_1 \times C_2 \times \dots \times C_{n-1} \rightarrow C_n$. Comme exemple de fonctions binaires, nous avons la fonction mère de et le carré, et comme exemple de fonction ternaire, le prix d'une voiture usagée sur lequel on peut se baser pour calculer le prix d'une voiture d'occasion en fonction de son modèle, de sa date de construction et de son kilométrage.

2.4.4. Axiomes : constituent des assertions, acceptées comme vraies, à propos des abstractions du domaine, traduites par l'ontologie. Ils ont pour objectif de représenter des concepts et des relations dans un langage logique permettant de représenter leur sémantique. Ils représentent les intentions des concepts et des relations du domaine et, de manière générale, les connaissances n'ayant pas un caractère strictement terminologique (S. Staab, 2000). L'utilisation des axiomes sert à définir le sens des entités, mettre des restrictions sur la valeur des attributs, examiner la conformité des informations spécifiées ou en déduire de nouvelles.

2.4.5. Instances: elles constituent la définition extensionnelle de l'ontologie; ces objets véhiculent les (G. Van Heijst, 1997)Connaissances (statiques, factuelles) à propos du domaine du problème.

2.5. Typologie des ontologies

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

2.5.1. Les ontologies de représentation (G. Van Heijst, 1997)

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, 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.

2.5.2. Les ontologies génériques (R. Mizoguchi, 1997) (N. Guarino, 1995)

Sont aussi appelée 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 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.

2.5.3. Les ontologies de domaine (G. Van Heijst, 1997)

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 dans le domaine médical. Entreprise est un autre exemple décrivant le domaine de l'entreprise.

2.5.4. Les ontologies de tâches (Guarino, Formal ontology in information systems, 1998)

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.

2.5.5. Les ontologies d'application (G. Van Heijst, 1997)

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

2.6. Utilisation des ontologies

Même si le besoin de développer une ontologie est très varié et dépend du domaine d'application, on peut facilement énumérer un certain nombre d'utilités, notamment:

2.6.1. La connaissance du domaine : Les ontologies permettent la modélisation des connaissances dans un domaine particulier, dans lequel opère le système à développer.

2.6.2. La communication: les ontologies assurent une communication fiable et hétérogène entre personnes et machines (agents logiciels ou organisations) du fait qu'elle permet de mettre en place un langage ou un vocabulaire conceptuel commun.

2.6.3. L'interopérabilité : La représentation explicite des connaissances dans un

domaine donné sous forme d'une ontologie, permet à son tour une plus grande réutilisation, un partage plus large et une interopérabilité plus étendue.

2.6.4. L'aide à la spécification des systèmes: La représentation conceptuelle des éléments du domaine, permet aux systèmes de réaliser des raisonnements logiques qu'on appelle inférences, et de sortir avec des conclusions capables d'aider l'utilisateur ou le gestionnaire dans ses décisions.

2.6.5. L'indexation et la recherche d'information: Dans le web sémantique, d'une façon générale, dans certains cas en particulier, les ontologies sont utilisées pour indexer et décrire les ressources utilisées. Cela permet une plus grande précision dans les résultats des recherches ou d'assignation des ressources. (S.Bouarroudj,2009)

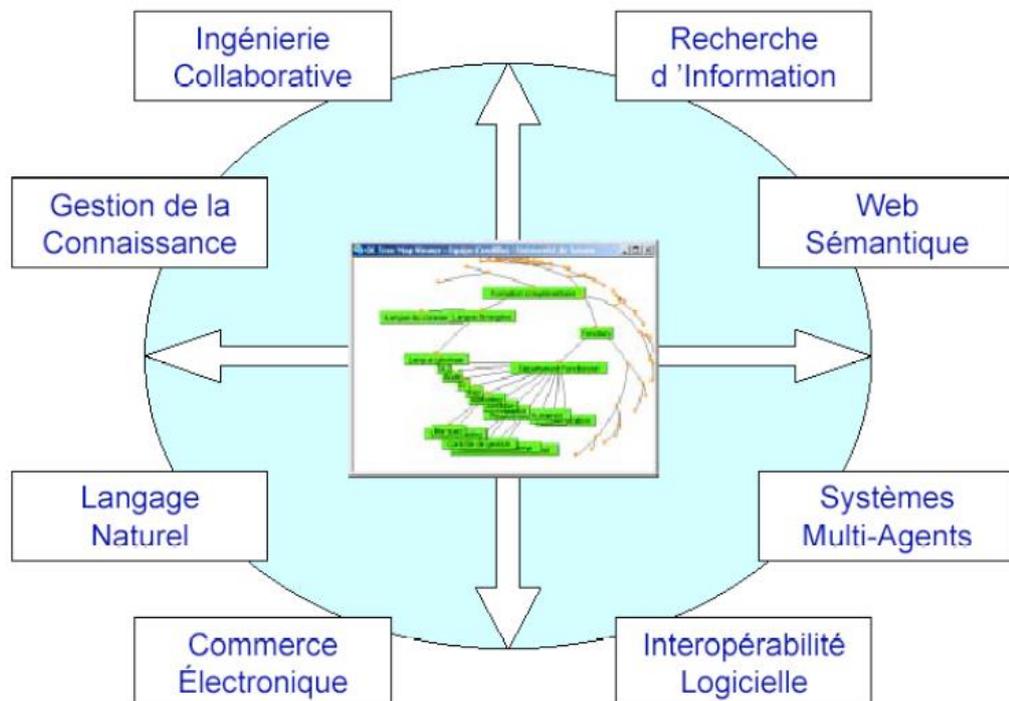


Figure III-2 : Quelque utilisation des ontologies. (ESPINASSE, 2010)

2.7. Construction d'une ontologie

Le processus de construction d'une ontologie doit respecter certains principes de bases qui permettent d'obtenir une ontologie susceptible de répondre aux objectifs de l'ontologie. Le constructeur de l'ontologie, se doit donc de garder à l'esprit ces principaux critères tout au long du cycle de développement de son ontologie (Gruber T. , 1993).

La clarté et l'objectivité : l'ontologie doit fournir le sens des termes définis en offrant des

définitions objectives ainsi que de la documentation associée en langage naturel.

L'exhaustivité: une définition exprimée par une condition nécessaire et suffisante est préférable à une définition exprimée seulement par une condition nécessaire ou par une condition suffisante. La cohérence : afin de pouvoir formuler des inférences cohérentes avec les définitions. L'extensibilité monotone maximale: Les nouveaux termes, qu'ils relèvent de la langue générale ou d'une langue de spécialité, devraient être inclus dans l'ontologie sans entraîner de modifications dans les définitions existantes.

Nous allons suivre le processus de construction spécifié dans (M.Hemam, 2005). Ce processus est inspiré des différentes phases proposées par la méthode METHONTOLOGY.

-Chaque domaine est caractérisé par une liste de services, activités et tâches. Ces services sont liés aux concepts existants dans l'ontologie du domaine. Un concept contient un ensemble de propriétés. Par exemple dans notre ontologie de domaine, le concept « hôtel » a les propriétés « Name », « Star Numéro » et « Adresse ». Chaque concept peut avoir des sous concepts, comme « room » et « suite » pour le concept « hôtel ». Chaque concept de l'ontologie du domaine est le sujet d'un ou plusieurs services, activités ou tâches. Par exemple, le service «Lodging_hotel » est associé au concept « hôtel ». Cette relation donne la possibilité d'améliorer la recherche et aider les utilisateurs à mieux exprimer leurs besoins. La construction de cette ontologie des services, ainsi que la mise en correspondance avec l'ontologie de domaine, sont pour le moment réalisées manuellement. **(Ben Mustapha, N., BaazaouiZghal, H. and Aufaure, MA, 2007)**

2.8. Quelques méthodologies de construction d'ontologie

Les méthodologies peuvent porter sur l'ensemble du processus et guider l'otologiste dans toutes les étapes de la construction. Bien qu'aucune méthodologie générale n'ait pour l'instant Réussi à s'imposer, de nombreux critères de construction d'ontologies ont été proposés pour des méthodologies. ENTERPRISE, TOVE (Toronto Virtual Enterprise) et METHONTOLOGY sont les méthodologies les plus représentatives pour construire des ontologies.

2.9. Cycle de vie d'une ontologie

Puisque les ontologies sont destinées à être utilisées comme des composants logiciels dans des systèmes répondant à des objectifs opérationnels différents, leur développement doit s'appuyer sur les mêmes principes que ceux appliqués en génie logiciel (Gruber T.-R. , 1995). Ainsi, les ontologies doivent être considérées comme des objets techniques évolutifs et possédants un cycle de vie qui nécessite d'être précisé. Dans ce contexte, les activités liées

aux ontologies sont, d'une part, des activités de gestion de projet (planification, contrôle, assurance qualité), et, d'autre part, des activités de développement (spécification, conceptualisation, formalisation). Un cycle de vie d'une ontologie comprend une étape initiale d'évaluation des besoins, une étape de construction, une étape de diffusion et une étape d'utilisation. La phase de construction peut être décomposée en 3 étapes: conceptualisation, ontologisation, opérationnalisation.

3. WEB SEMANTIQUE :

Le Web sémantique 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, grâce à un système de métadonnées formelles, utilisant notamment la famille de langages développés par le W3C (World Wide Web Consortium). (Y. Askane, Y. El-ouchi, S. Elmes)

A 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".

Le Web sémantique (T. Berners-Lee, 2001) 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 grâce à un système de métadonnées formelles utilisant notamment la famille de langages développés par le W3C.

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.

Le web sémantique se veut un web de demain dont le contenu peut être appréhendé et exploité par des machines. Résumons les objectifs du web sémantique dans les points suivants:

- ✓ Créer un espace d'échange de ressource entre machines permettant l'exploitation de grand volume d'informations et de services variés.
- ✓ Libérer l'utilisateur d'une bonne partie de son travail de recherche et d'exploitation des résultats.
- ✓ Rendre explicite les relations sémantiques qui existent entre unités d'information.
- ✓ Obtenir des annotations sémantiques plus riches, compatibles avec les standards du Web, tel que le XML, qui fournit un format universel de stockage et d'échange.

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 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.

3.2. Langages du Web sémantique (S.Bouarroudj,2009)

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) et OWL.

Le langage XML(eXtensibleMarkup Langage)est un langage de description et d'échange de documents structurés, issue de SGML (Standard GenralizedMarkupLanguage). 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.

XML est un langage de description de format de document XML permettant de définir la structure d'un document XML. La connaissance de la structure d'un document XML permet notamment de vérifier la validité de ce document. Un fichier de description de structure(XML Schéma Description, ou fichier XSD) est donc lui-même un document XML.

Les espaces de nommage XML offrent une méthode simple pour qualifier les noms des éléments et les attributs utilisés dans des documents XML, en associant ceux-ci avec des espaces de nommage désignés par des références d'URI(Uniform Resource Identifier).

RDF (Resource Description Framework) (L. Lassila, 1999)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. La construction de base en RDF consiste en un triplé d'éléments (Ressource, Propriété, Valeur), qu'on appelle déclaration RDF. Par analogie, un triplé RDF est similaire à la déclaration < sujet — prédicat — objet >

***Ressource** (Sujet): Cela peut être n'importe quel objet référencé par une URI, qu'il concerne le web (Page HTML, document PDF, fichier multimédia...), ou non (Personne, Région, Etc.).

***Propriété** (prédicat): Critère, caractéristique, attribut ou relation qui peut décrire la Ressource (titre, couleur, taille, auteur, etc.).

***Valeur** (objet): C'est la valeur qui sera affectée à la propriété de la ressource. Cette affectation peut être soumise à certaines restrictions.

RDF-S (RDF Schema) est un vocabulaire de base pour décrire les déclarations RDF, au même titre que le XML-S pour le langage XML. Il ajoute à RDF la possibilité de définir des hiérarchies de classes et de définir les genres et les propriétés des ressources, d'assigner des contraintes spécifiques sur la nature des documents et de fournir des informations sur l'interprétation des déclarations RDF. Les schémas RDF permettent donc de garantir qu'un document RDF est sémantiquement consistant. Dans un schéma, de nouvelles ressources peuvent être définies comme des spécialisations d'autres ressources. Les schémas contraignent aussi le contexte d'utilisation des ressources.

Le langage OWL (Ontology Web Language) 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. La Figure III -4 montre que Le OWL peut être défini en trois sous langages complémentaires proposant une expressivité croissante (D. L. McGuinness, 2004), chacun conçu pour des communautés de développeurs et des utilisateurs spécifiques : OWL Lite, OWL DL, OWL Full.

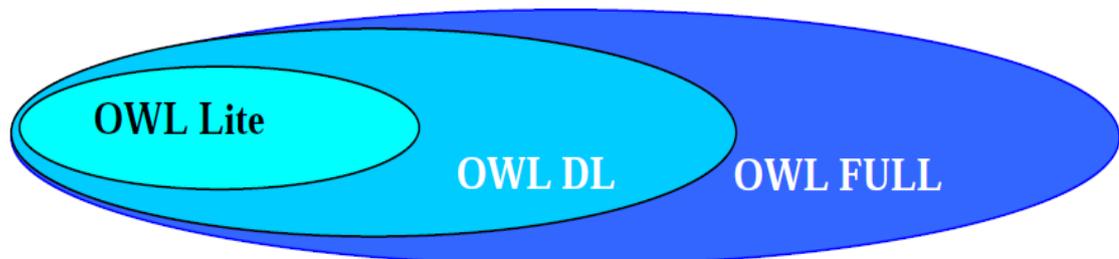


Figure III-3 : Les types de langage OWL (D. L. McGuinness, 2004)

Le langage OWL Lite répond à des besoins de hiérarchie de classification et de fonctionnalités de contrainte simples de cardinalité 0 ou 1. Une cardinalité 0 ou 1 correspond à des relations fonctionnelles, par exemple: une personne possède une adresse. Toutefois, cette personne peut avoir un ou plusieurs prénoms. OWL Lite ne suffit donc pas pour cette situation. Il ne permet pas d'exprimer des contraintes complexes sur les classes ou sur les associations.

Le langage OWL DL (Description Logic) concerne les utilisateurs qui souhaitent une expressivité maximum tout en maintenant la complétude (garantie de calculer toutes les conclusions) et la décidabilité (tous les calculs doivent être achevés en un temps fini). Ce langage inclut toutes les structures OWL avec certaines restrictions, comme la séparation des types : une classe ne peut pas aussi être un individu ou une propriété. Il est nommé DL car il correspond à la logique de description. Le langage OWL Full est destiné aux personnes souhaitant une expressivité maximale, ainsi que la liberté syntaxique de RDF, mais sans aucune garantie de calcul.

En OWL Full, une classe peut également être un individu, il n'y a pas de séparation des types. OWL permet donc à une ontologie d'augmenter le sens du vocabulaire prédéfini.

4. LES ONTOLOGIES ET LE WEB SEMANTIQUE

Le Web actuel est essentiellement syntaxique, la structure des ressources étant bien définie, mais leur contenu restant inaccessible aux traitements machines, seuls les humains étant capables de l'interpréter.

Le Web sémantique (al, 2002) a alors l'ambition de lever cette difficulté en associant aux ressources du Web des entités ontologiques comme références sémantiques, ce qui permettra aux différents agents logiciels d'accéder et d'exploiter directement le contenu des ressources et de raisonner dessus. Ce référencement sémantique peut aussi résoudre les problèmes d'interprétation des ressources informationnelles provenant des applications hétérogènes et réparties et de permettre ainsi à ces applications d'être intégrées sémantiquement (Grüninger, 1996).

L'architecture du Web sémantique repose sur une hiérarchie des langages d'assertion et de description d'ontologies ainsi que sur un ensemble de services pour l'accès aux ressources au moyen de leurs références sémantiques, pour gérer l'évolution des ontologies, pour l'utilisation des moteurs d'inférences capables d'effectuer des raisonnements complexes ainsi que des services pour la vérification de la validité sémantique de ces raisonnements (D. Oberle, 2004).

4.1. La recherche d'information guidée par les ontologies dans le web sémantique

Les systèmes de recherche d'information devraient fournir à l'utilisateur un accès facile à l'information à laquelle il s'intéresse. Cependant, les systèmes traditionnels ont des difficultés à fournir un résultat pertinent.

Actuellement la recherche d'information sur le web est essentiellement basée sur des technologies plein texte comme dans les principaux moteurs de recherche, par exemple Google et Altavista. Deux problèmes nous semblent importants dans la manière dont ces moteurs fonctionnent:

La recherche se fait par terme, donc en prenant en compte uniquement une notion syntaxique. Les problèmes d'ambiguïté et d'hétérogénéité sont donc relativement fréquents et peuvent nuire à la qualité des résultats.

Les moteurs de recherche rendent souvent des centaines de documents pour chaque requête. La tâche la plus lourde revient à l'utilisateur qui doit fouiller dans cette masse de résultats pour sélectionner les documents qui lui seront les plus utiles. Les résultats ne sont pas tous pertinents et l'information retrouvée n'est pas complète.

Un des enjeux actuels de la recherche d'information est de développer des systèmes capables d'intégrer plus de sémantique dans leurs traitements. L'objectif est double : comprendre le contenu des documents et comprendre le besoin de l'utilisateur pour pouvoir les mettre en relation. La recherche sur le web sémantique peut en plus de l'exploitation plein texte comme dans le web actuel, exploiter des annotations des documents et des ontologies (Gruber T. R., 1993).

Ainsi, certains des travaux actuels dans la RI tentent d'améliorer le procédé de récupération avec l'aide des ontologies. Elles peuvent aider l'utilisateur à détecter ses besoins et trouver les mots-clés appropriés qui utilisent les concepts existants dans l'ontologie et leur description. La connaissance qui représente les ontologies peut être utilisée à différents niveaux dans le processus de RI. Elle peut aider à l'indexation des documents, alors appelée indexation sémantique. Les ontologies peuvent également aider à la formulation du besoin de l'utilisateur et à l'accès aux documents. Enfin l'ontologie peut être utilisée dans le modèle lui-même pour réaliser l'appariement entre le besoin et les documents.

La recherche sémantique ou 'guidée par les ontologies' est une inférence exécutée par un raisonneur sur un ensemble de règles et de relations entre les instances modélisées par un langage formel telles que les logiques de descriptions et les règles SWRL. Cette recherche permet d'accéder aux ressources selon leur contenu plutôt que par mot clés. Les annotations des documents et la requête sont exprimés en utilisant le vocabulaire de l'ontologie.

4.2. Exemple de l'utilisation d'ontologie dans la recherche d'information : moteur de recherche sémantique

La recherche sémantique est une inférence exécutée par un raisonneur sur un ensemble de règles et de relations entre les instances. Dans la plupart des cas, les règles sont modélisées par un langage formel tel que les logiques de descriptions (Napoli, 1997) et les règles SWRL (Semantic Web Rule Language) (I. Horrocks, 2004) ou par des graphes conceptuels (GC).

A titre d'exemple, citons le moteur de recherche sémantique *Corese* synonyme de COnceptual REssource moteur de recherche. Il s'agit d'un moteur basé sur RDF Conceptual Graphs (CG). Il permet le traitement de RDF et RDF Schéma déclarés dans le formalisme CG. Les principales fonctionnalités de Corese sont dédiées à extraire des ressources Web annotées dans RDFS, en utilisant un langage de requête basé sur SPARQL et un moteur de règles d'inférence. La recherche est basée sur les annotations sémantiques qui sont des instanciations des schémas RDFS.

4.2.1. Google, vers un moteur de recherche sémantique:

Première vue, rien ne distingue un moteur de recherche « classique », comme Google, d'un moteur de recherche « sémantique ». Même interface sobre, avec au centre de la page une fenêtre dans laquelle l'utilisateur entre sa requête.

En fait, la différence se fait sur le mode de recherche. Un moteur « classique », type Google, fonctionne de la manière suivante : ses robots indexent dans les pages qu'ils parcourent des « mots ». Ceux-ci sont ensuite rangés dans une gigantesque base de données. Les requêtes que font les utilisateurs sont une recherche dans ce « dictionnaire géant », un algorithme permettant de classer et présenter les résultats selon un certain ordre de pertinence.

Actuellement, Google détermine l'importance d'un site web grâce à son algorithme en analysant les mots contenus dans le site ainsi que les liens pointant vers ce dernier. La mise à jour de l'algorithme de Google va apporter une nouvelle donnée. Elle permettra au moteur de recherche de comprendre les requêtes des internautes en s'intéressant d'avantage au sens des mots. La recherche sémantique dans Google n'est pas nouvelle. Le moteur de recherche s'intéresse au sens des mots depuis plusieurs années en proposant directement sur la page de résultats une réponse aux requêtes des internautes comme les définitions, les traductions ou

encore les recherches liées à la géolocalisation comme « météo » ou « cinéma ». Il s'agit donc d'une nouvelle évolution du moteur, ce n'est pas un bouleversement.

Comme ses concurrents, Google souhaite retenir les internautes sur son site. Par exemple, Facebook a intégré les résultats de Bing dans son moteur de recherche pour garder l'internaute sur son site. Google, lui, mise sur une recherche sémantique encore plus performante. Cette mise à jour de l'algorithme de Google pourrait concerner 10 à 20% des résultats de recherches selon le Wall Street Journal. Avec la recherche sémantique, Google va être en concurrence directe avec différents types de sites, comme les sites d'informations tels que Wikipédia ou les sites de questions-réponses comme Yahoo Answers. (site2)



Figure III-4 : Google Sur son chemin vers sémantique. (site2)

5. CONCLUSION

Nous avons présenté dans ce chapitre les notions liées aux ontologies et au Web Sémantique. Cette présentation, bien que n'étant pas exhaustive car ce domaine est assez vaste.

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. Les ontologies contribuent à l'excellent travail dans le développement de moteurs de recherche sémantique.

Dans le chapitre suivant, nous présenterons la modélisation de notre réalisation : un outil de recherche d'information sur le web, par le langage de modélisation UML (UML2)

Deuxième partie :

Modélisation et Implémentation

Chapitre IV

Modélisation

Dans ce chapitre :

- ❖ Introduction
- ❖ Langage UML
- ❖ L'implémentation de notre système
- ❖ Conclusion

1. INTRODUCTION

Les techniques de programmation n'ont cessé de progresser depuis l'époque de la programmation en langage binaire à nos jours. Cette évolution a toujours été dictée par le besoin de concevoir et de maintenir des applications toujours plus complexes.

La taille des applications ne cessant, alors, de croître, la programmation structurée (Pascal en 1970, C en 1972, Modula et Ada en 1979, ...) a également rencontré ses limites, faisant alors place à la programmation orientée objet (Simula 67 en 1967, Smalltalk en 1976, C++ en 1982, Java en 1995, ...). La technologie objet est donc la conséquence ultime de la modularisation dictée par la maîtrise de la conception et de la maintenance d'applications toujours plus complexes. Cette nouvelle technique de programmation a nécessité la conception de nouvelles méthodes de modélisation. (Ahmed, Introduction à UML, 2011)

Notre travail consiste à réaliser un outil de recherche intelligente d'information sur le Web. Et afin de faire une bonne réalisation, il faut faire une bonne modélisation .

Dans notre travail, pour l'étape de modélisation, nous avons choisi la méthode de modélisation UML (UML2).

Notre démarche de travail consisté on premier lieu à la création des diagrammes de cas utilisation, de séquences et du diagramme de collaboration et de classe, puis en suite, pour chaque cas d'utilisation, nous associons un diagramme d'activités.

2. LANGUAGE UML :

2.1. Définition UML

UML (Unified Modeling Language) c'est-à-dire, langage unifié pour la modélisation. C'est une notation graphique (Ce n'est pas une méthode) destinée à la création de modèles orientés objet en vue de l'analyse et de la modélisation de logiciels orientés objet. (Ahmed, balicours, 2011)

- ✓ Un langage de Modélisation (standard).
- ✓ Basé sur une approche solide Objecter.
- ✓ Un support important de stratégies et d'heuristiques.
- ✓ Adapté au développement Objet.
- ✓ Adapté au développement Itératif.
- ✓ Adaptable au processus d'une équipe.
- ✓ Supporté par des outils.

2.2. Pourquoi UML ?

- ✓ Obtenir une modélisation de très haut niveau indépendante des langages et des environnements.
- ✓ Faire collaborer des participants de tous horizons autour d'un même document de synthèse.
- ✓ Faire des simulations avant de construire un système.
- ✓ Exprimer dans un seul modèle tous les aspects statiques, dynamiques, juridiques, spécifications, etc...
- ✓ Documenter un projet.
- ✓ Générer automatiquement la partie logicielle d'un système. (D.Revuz, 2004)

2.3. Les diagrammes UML (Ahmed, Introduction à UML, 2011)

UML 2.0 comporte treize types de diagrammes représentant autant de *vues* distinctes pour représenter des concepts particuliers du système d'information. Ils se répartissent en deux grands groupes :

2.3.1. Diagrammes structurels ou diagrammes statiques (*UML Structure*)

- ✓ diagramme de classes (*Class diagram*)
- ✓ diagramme d'objets (*Object diagram*)

- ✓ diagramme de composants (*Component diagram*)
- ✓ diagramme de déploiement (*Deployment diagram*)
- ✓ diagramme de paquetages (*Package diagram*)
- ✓ diagramme de structures composites (*Composite structure diagram*)

2.3.2. Diagrammes comportementaux ou diagrammes dynamiques (UML Behavior)

- ✓ diagramme de cas d'utilisation (*Use case diagram*)
- ✓ diagramme d'activités (*Activity diagram*)
- ✓ diagramme d'états-transitions (*State machine diagram*)
- ✓ Diagrammes d'interaction (*Interaction diagram*)
 - diagramme de séquence (*Sequencediagram*)
 - diagramme de communication (*Communication diagram*)
 - diagramme global d'interaction (*Interaction overview diagram*)
 - diagramme de temps (*Timing diagram*)

2.3.3. Les diagrammes en détails :

2.3.3.1. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. Il permet de recueillir, d'analyser et d'organiser les besoins.

2.3.3.2. Diagramme de classes

Le diagramme de classes est généralement considéré comme le plus important dans un développement orienté objet. Il montre la structure interne d'un système. Il contient principalement des classes.

2.3.3.3. Diagramme d'objets

Le diagramme d'objets permet d'éclairer un diagramme de classes en l'illustrant par des exemples. Il est, par exemple, utilisé pour vérifier l'adéquation d'un diagramme de classes à différents cas possibles.

2.3.3.4. Diagramme d'états-transitions

Le diagramme d'états-transitions représente la façon dont évoluent (*i.e.* cycle de vie) les objets appartenant à une même classe. La modélisation du cycle de vie est essentielle pour représenter et mettre en forme la dynamique du système.

2.3.3.5. Diagramme d'activités

Le diagramme d'activités n'est autre que la transcription dans UML de la représentation du processus telle qu'elle a été élaborée lors du travail qui a préparé la modélisation : il montre l'enchaînement des activités qui concourent au processus.

2.3.3.6. Diagramme de séquence et de communication

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre. On peut représenter les mêmes opérations par un diagramme de communication, graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) les échanges entre objets. En fait, le diagramme de séquence et le diagramme de communication sont deux vues différentes mais logiquement équivalentes (on peut construire l'une à partir de l'autre) d'une même chronologie. Ce sont des diagrammes d'interaction.

3. L'IMPLEMENTATION DE NOTRE SYSTEME

3.1. Diagramme de cas utilisation :

Le tableau ci-après récapitule l'ensemble des cas d'utilisation qui seront détaillés par la suite, en mettant en exergue les notions suivantes : acteur, diagramme de cas d'utilisation, les enchaînements (scénarios), le diagramme de séquence, le diagramme d'activité.

Cas d'utilisation	Acteurs
Recherche Consultation	L'utilisateur
Initialisation Réinitialisation	L'administrateur

Tableau IV-1. Les cas d'utilisation du l'utilisateur et du l'administrateur.

Notre diagramme de cas d'utilisation est représenté dans la figure Suivante:

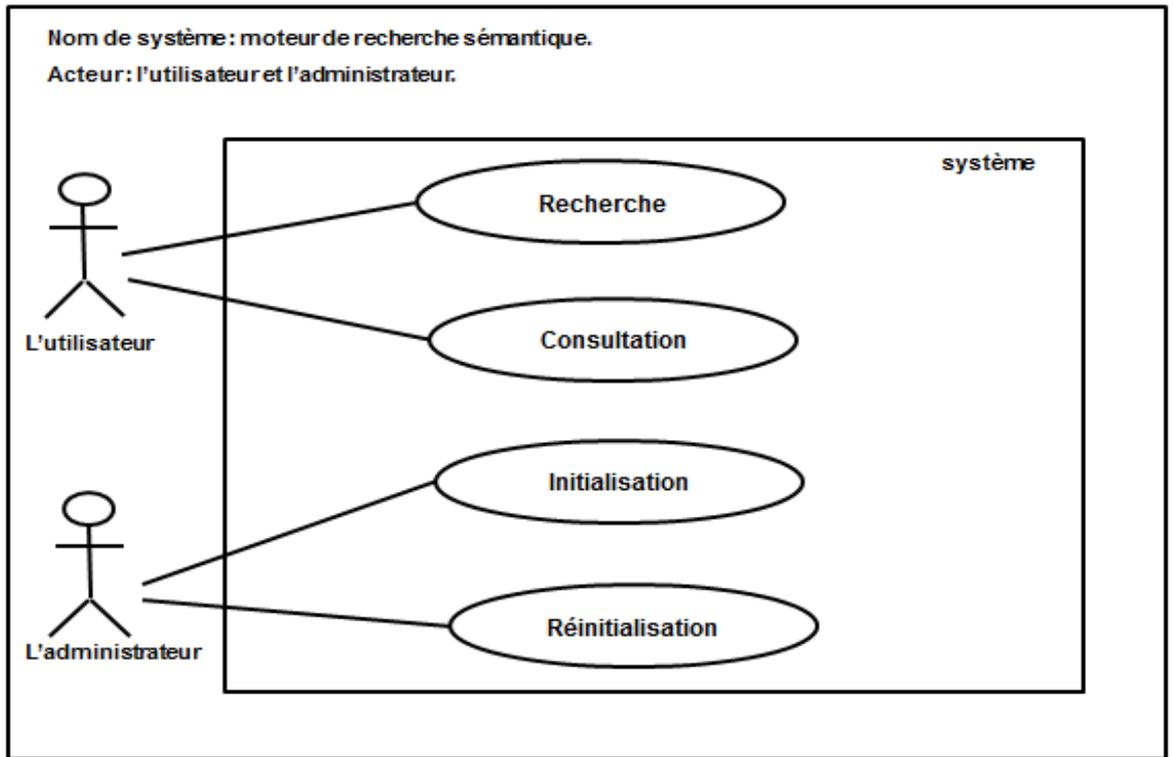


Figure IV -1 : Diagramme de cas utilisation de notre système.

3.2. Diagramme de séquence :

3.2.1. Diagramme de séquence (Initialisation de Crawler)

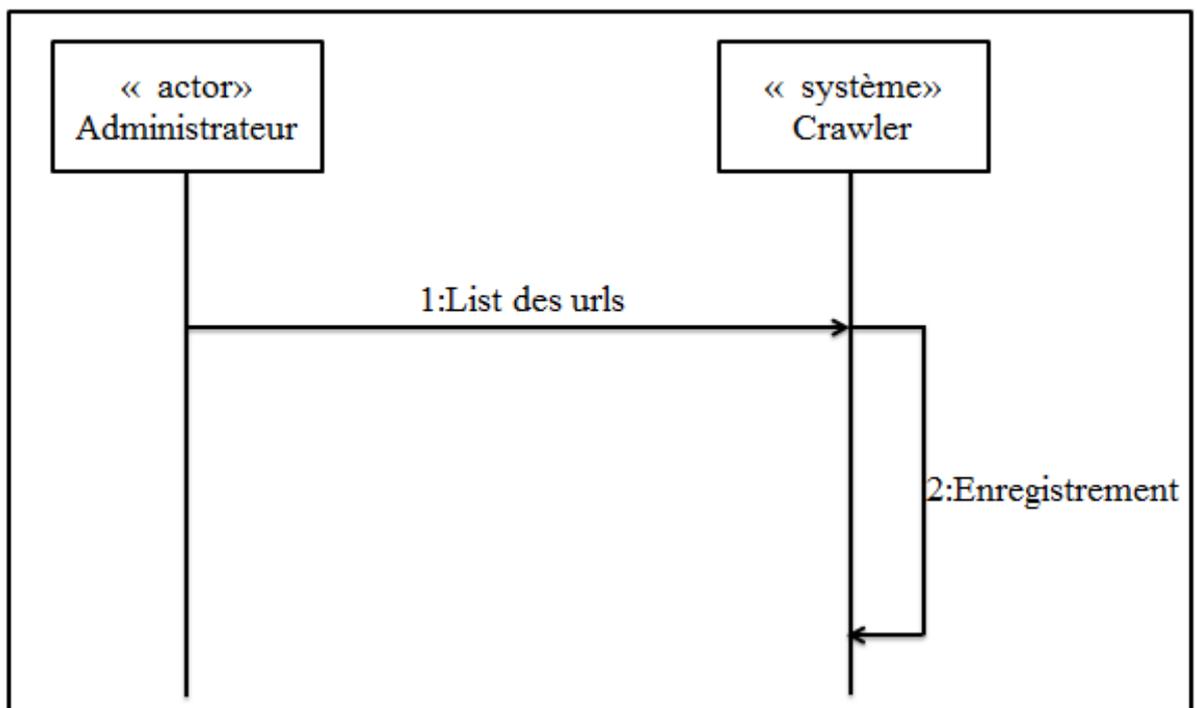


Figure IV-2 : Diagramme de séquence « Initialisation de Crawler ».

Scénario:

Quand le système se lance, l'administrateur peut initialiser la liste adresses (urls) qui on doit explorer, ou enrichie la liste existent.

3.2.2. Diagramme de séquence(Crawler)

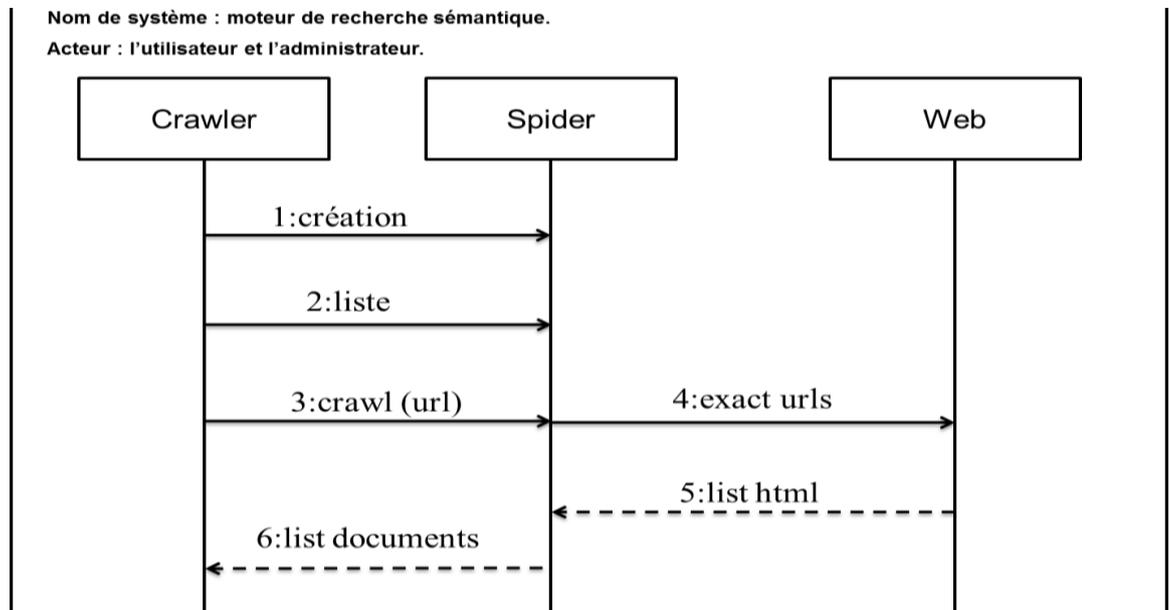


Figure IV-3: Diagramme de séquence « Crawler».

Scénario:

Le crawler va créer un ensemble des spiders qui doivent explorer les pages web.

3.2.3. Diagramme de séquence(Réinitialisation de Crawler)

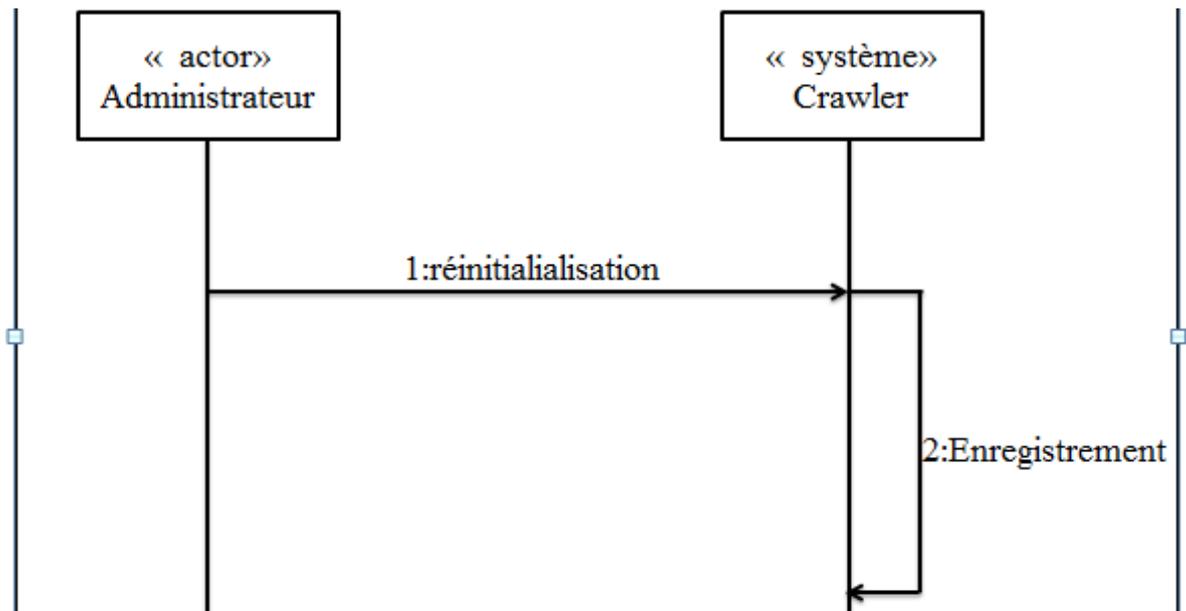


Figure IV-4 : Diagramme de séquence « Réinitialisation de Crawler».

Scénario:

Si l'administrateur envoyer un message de réinitialisation, le système sauvegarde tout la base des segments.

3.2.4. Diagramme de séquence(Système)

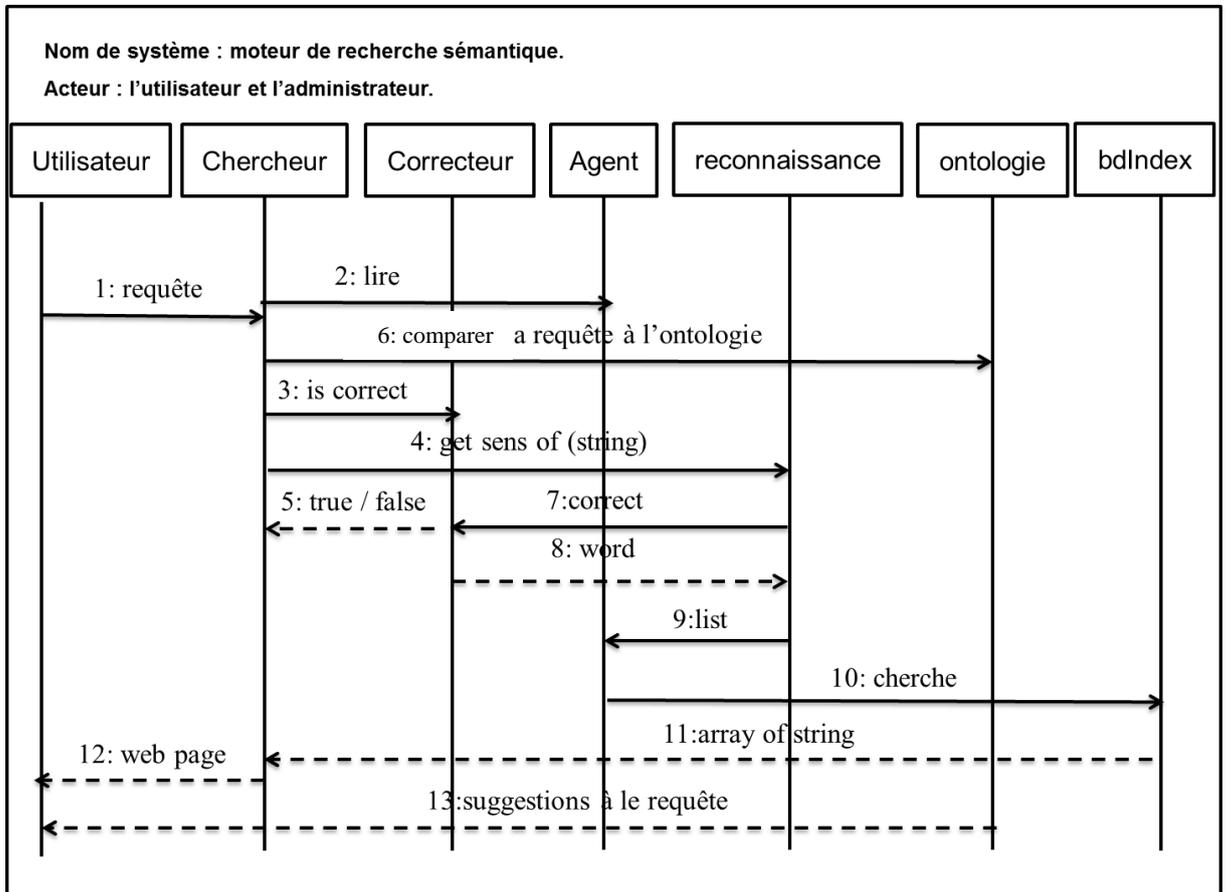


Figure IV-5 : *Diagramme de séquence « Système».*

3.3. Diagramme de collaboration :

3.3.1. Diagramme de collaboration (Crawler):

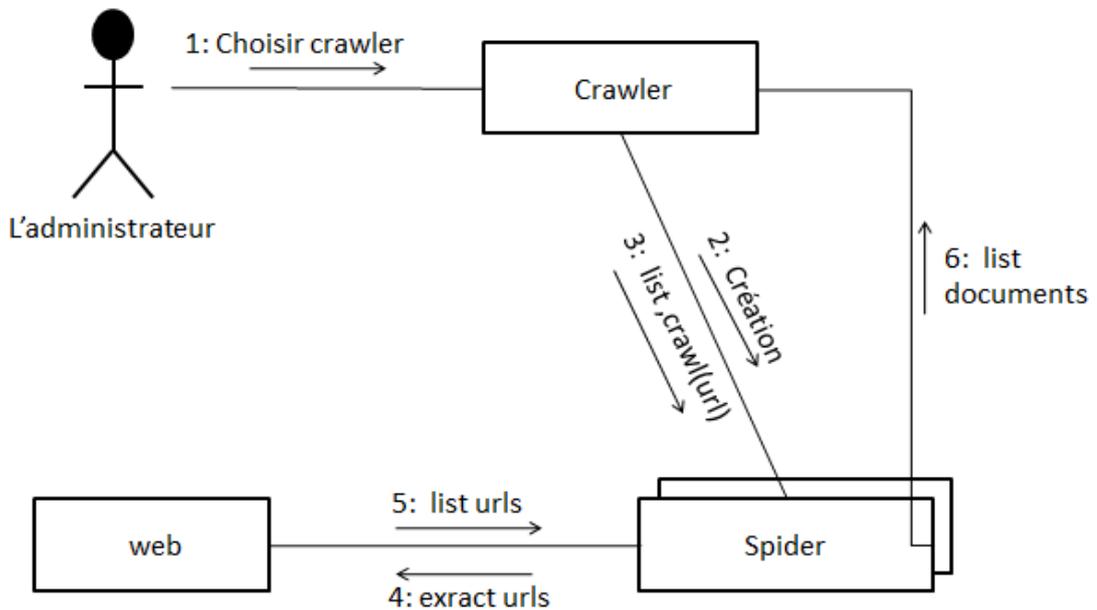


Figure IV-6 : *Diagramme de collaboration «Crawler».*

Scénario:

Le crawler va créer un ensemble des spiders qui doivent explorer les pages web.

3.3.2. Diagramme de collaboration (Model de recherche):

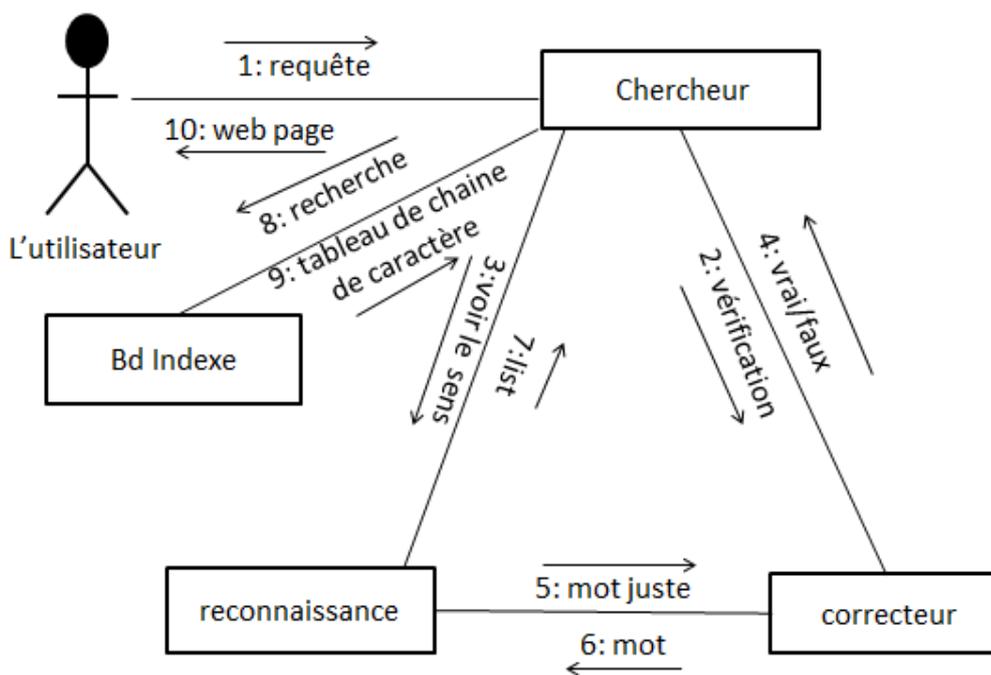


Figure IV-7 : *Diagramme de collaboration «Model de recherche».*

Scénario:

L'utilisateur envoie une requête au chercheur.

3.4. Diagramme d'activité:**Scénario:**

L'utilisateur envoie une requête au chercheur.

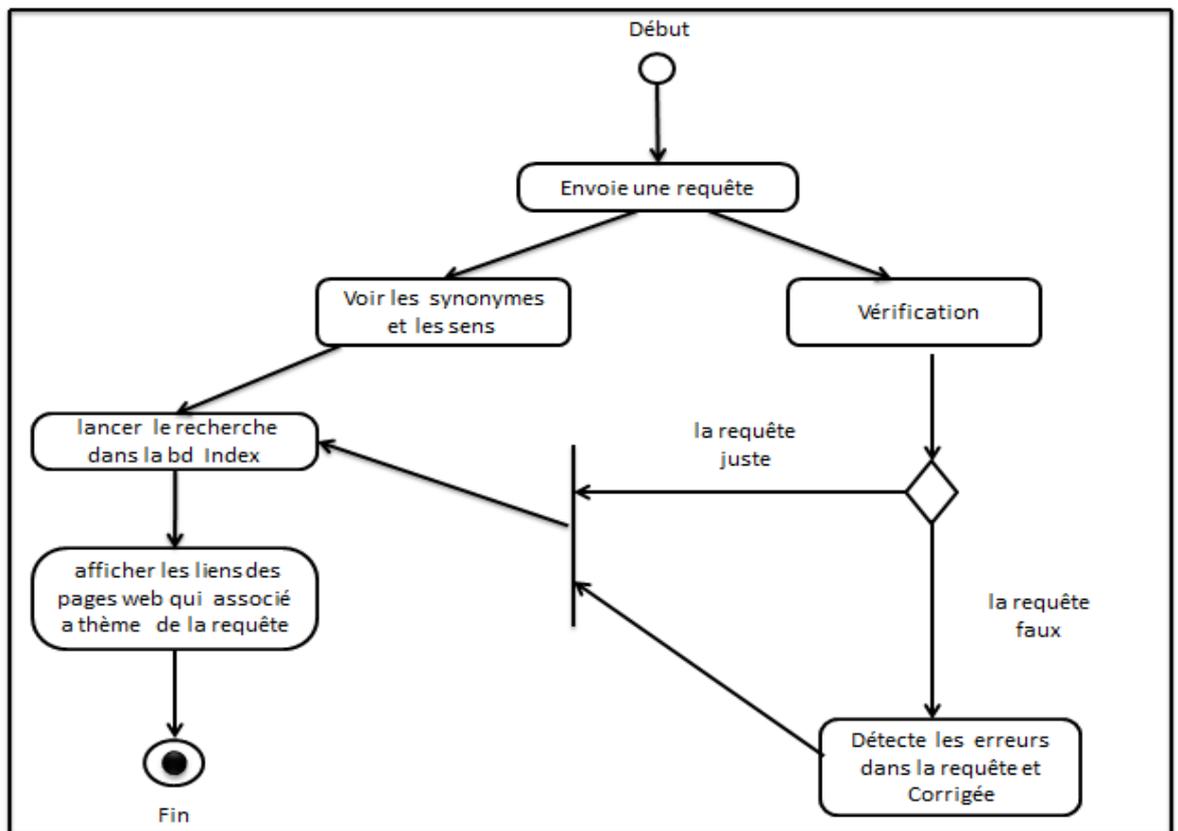


Figure IV-8 : *Diagramme d'activité « Model de recherche ».*

3.5. Diagramme de Classes (Système)

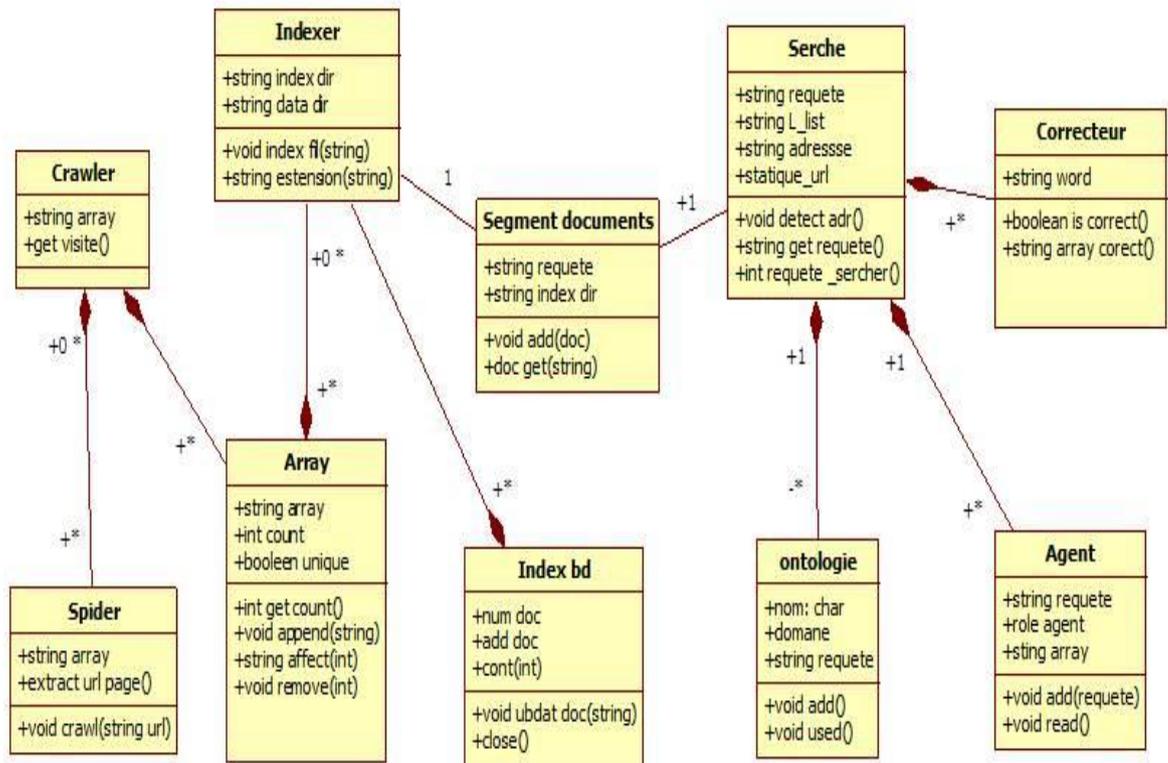


Figure IV-9 : Diagramme de classes « Système ».

4. Conclusion

Dans le but de faciliter l'implémentation de notre outil de recherche, dans ce chapitre, nous avons présenté notre modélisation par des diagrammes en UML afin de cerner les besoins des utilisateurs et les fonctions de l'administrateur du système.

Les résultats de ce chapitre seront enrichis par des détails d'implémentation dans le chapitre suivant pour réaliser notre système.

Chapitre V

Implémentation

Dans ce chapitre :

- ❖ Introduction
- ❖ Environnement du travail
- ❖ Architecture générale du système
- ❖ Implémentation des composants du système
- ❖ L'intelligence de notre outil
- ❖ Conclusion

1. INTRODUCTION

Dans les précédents chapitres nous avons défini les outils de recherches sur le web et les agents dans les moteurs de recherche et enfin, les ontologies pour la recherche d'information sur le web. Ce qui servira de base pour ce chapitre qui va être consacré à la mise en œuvre de l'acquis.

Afin de concrétiser un moteur de recherche qui utilise le paradigme agent et qui profite de l'apport de l'utilisation des ontologies, ce chapitre est essentiellement consacré aux langages et outils exploités pour créer les différentes parties de ce travail. Des captures d'écran seront aussi présentées.

2. ENVIRONNEMENT DU TRAVAIL :

2.1. Choix du langage de programmation :

2.1.1. Java:

Java est un langage de programmation orienté objet, développé par Sun Microsystems. Il fut présenté officiellement en 1995. Selon les développeurs de Sun, Java (qui signifie café en argot américain) est un langage : simple, orienté-objet, distribué, interprété, robuste sécurisé, neutre vis à vis de l'architecture, portable, à haute performance, multi-threaded et dynamique [SIL].

Le langage Java était à la base un langage pour Internet, pour pouvoir rendre plus dynamiques les pages (tout comme le JavaScript aujourd'hui). Mais le Java a beaucoup évolué et est devenu un langage de programmation très puissant permettant de presque tout faire. Contrairement à la plupart des autres langages (sauf la plateforme .Net), Java met à la disposition du développeur une API très riche lui permettant de faire de très nombreuses choses.

2.1.2. Java Script :

JavaScript est un langage de programmation de scripts principalement utilisé dans les pages web interactives mais aussi côté serveur. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés.

Le langage a été créé en 1995 par **Brendan Eich** (Brendan Eich étant membre du conseil d'administration de la fondation **Mozilla**) pour le compte de **Netscape** Communications Corporation. Le langage, actuellement à la version 1.8.2 est une implémentation de la 3^{ème} version de la norme ECMA-262 qui intègre également des éléments inspirés du langage Python. La version 1.8.5 du langage est prévue pour intégrer la 5^{ème} version du standard ECMA.

2.2.Choix de la plate-forme :

2.2.1. Jade 4.3.0 :

Le meilleur moyen pour construire un système multi-agent(SMA) est d'utiliser une plate-forme multi-agent. Une plate-forme multi-agent est un ensemble d'outils nécessaire à la construction et à la mise en service d'agents au sein d'un environnement spécifique. Ces outils peuvent servir également à l'analyse et au test du SMA ainsi créé. Ces outils peuvent être sous la forme d'environnement de programmation (API) et d'applications permettant d'aider le développeur.

JADE est un cadre pour développer des systèmes multi-agents en conformité avec les spécifications FIPA. JADE a passé avec succès le 1er test d'interopérabilité du FIPA à Séoul (janvier 99) et le test d'interopérabilité deuxième FIPA à Londres (avril 01) [BEF, 10]. Elle est implémentée en JAVA et fourni des classes qui implémentent « JESS » pour la définition du comportement des agents. JADE possède trois modules principaux (nécessaire aux normes FIPA):

- ❖ DF « Director Facilitator » fournit un service de « pages jaunes » à la plate-forme.
- ❖ ACC «Agent Communication Channel » gère la communication entre les agents .
- ❖ AMS « Agent Management System » supervise l'enregistrement des agents, leur authentification, leur accès et l'utilisation du système.

Ces trois modules sont activés à chaque démarrage de la plate-forme.

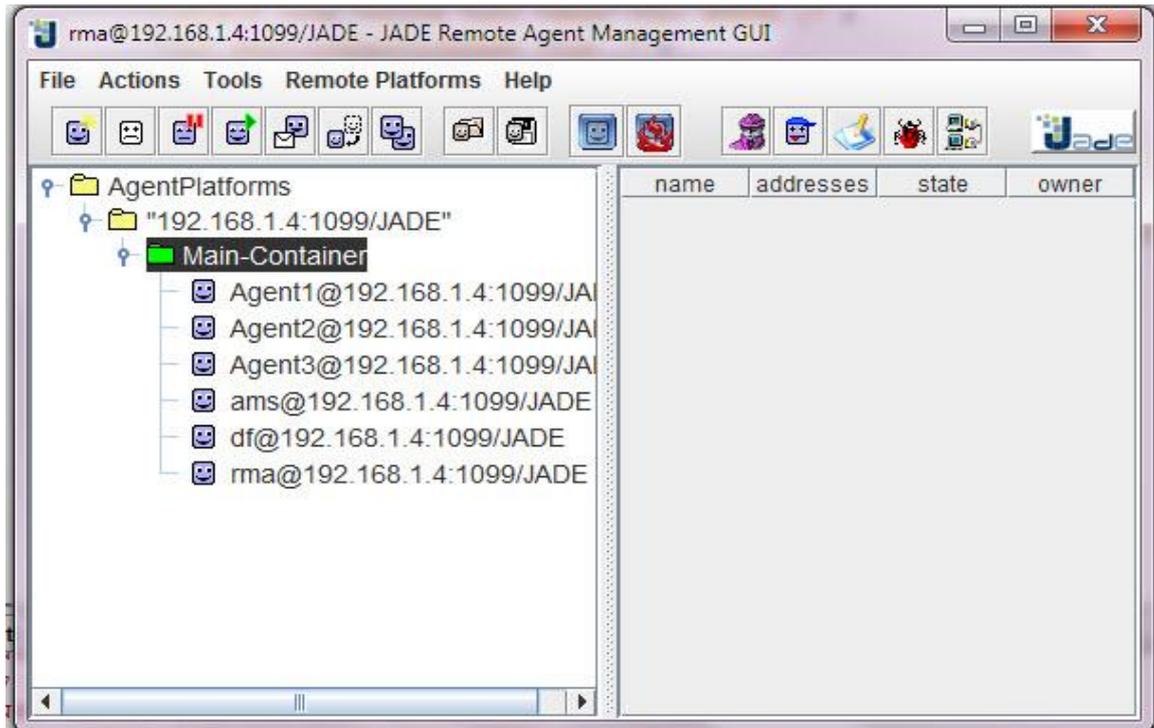


Figure V-1 : interface de la plateforme Jade 4.3.0 .

2.3.Choix des éditeurs :

2.3.1. NetBeans IDE 7.2.1 :

NetBeans est à l'origine un EDI Java qui fut développé par une équipe d'étudiants à Prague racheté ensuite par Sun Microsystems. En 2002 Sun a décidé de rendre NetBeans open-source. NetBeans n'est pas uniquement un EDI Java, c'est également une plateforme qui permet d'écrire des applications Swing ou C++, Python ou autres langages en lui incluant les plugins adéquats. Sa conception est complètement modulaire ce qui fait de lui une boîte à outils facilement améliorable ou modifiable.

La License de NetBeans permet de l'utiliser gratuitement à des fins commerciales ou non. Elle permet de développer tous types d'applications basées sur la plateforme NetBeans.

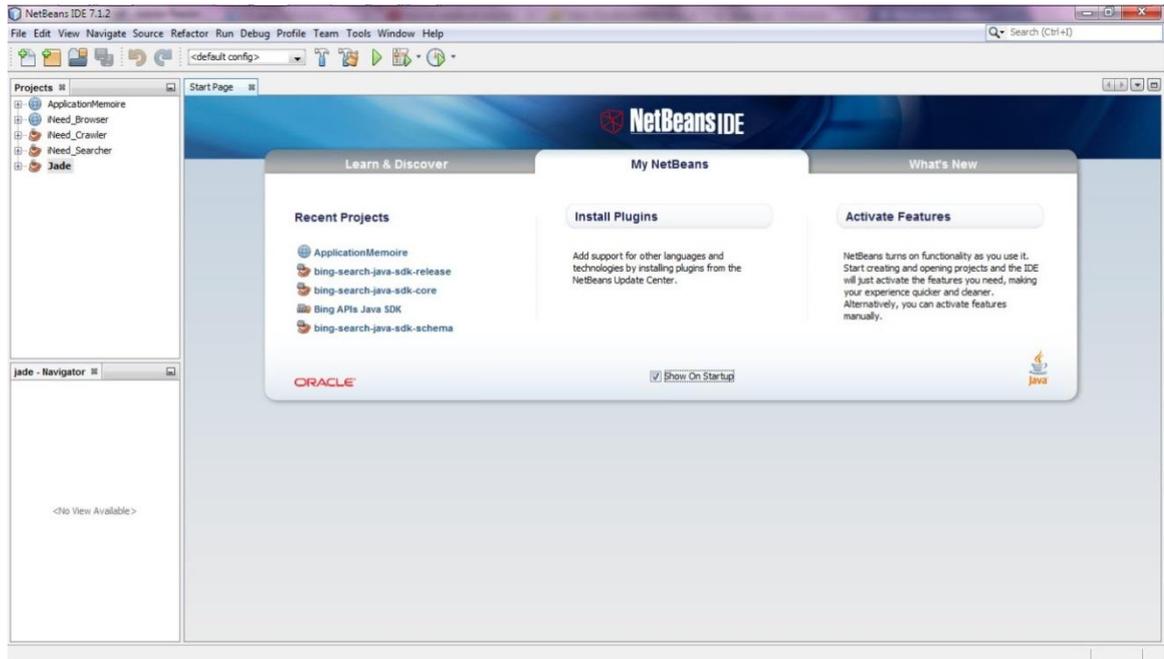


Figure V-2 : l'interface de l'IDE Netbeans 7.1.2

2.3.2. Protégé 3.4.4 :

Protégé est un éditeur qui permet de construire une ontologie pour un domaine donné, de définir des formulaires d'entrée de données et d'acquérir des données à l'aide de ces formulaires sous forme d'instances de cette ontologie. Protégé est également une librairie Java qui peut être étendue pour créer de véritables applications à bases de connaissances en utilisant un moteur d'inférence pour raisonner et déduire de nouveaux faits par application de règles d'inférence aux instances de l'ontologie et à l'ontologie elle-même (méta-raisonnement).

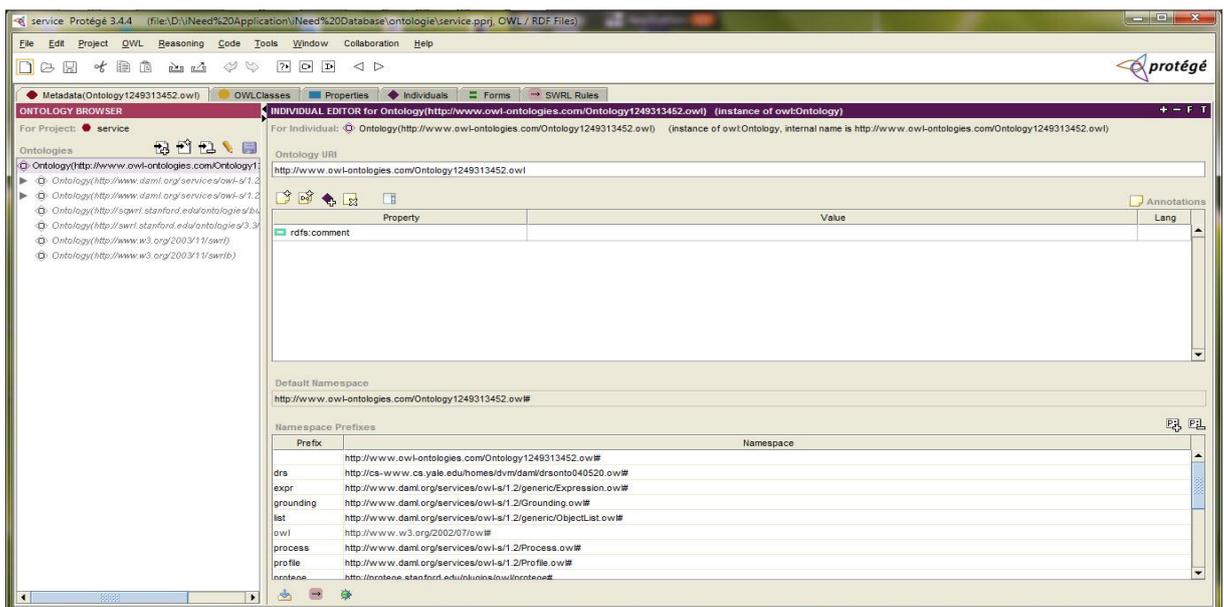


Figure V-3 : l'interface de Protégé 3.4.4

2.4. Choix de Serveur :

2.4.1. Apache Tomcat Server 7.0.40 :

Apache Tomcat est un conteneur libre de servlets et JSP Java EE. Issu du projet Jakarta, c'est un projet principal de l'Apache Software Foundation. Il implémente les spécifications des servlets et des JSP du Java Community Process, est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.

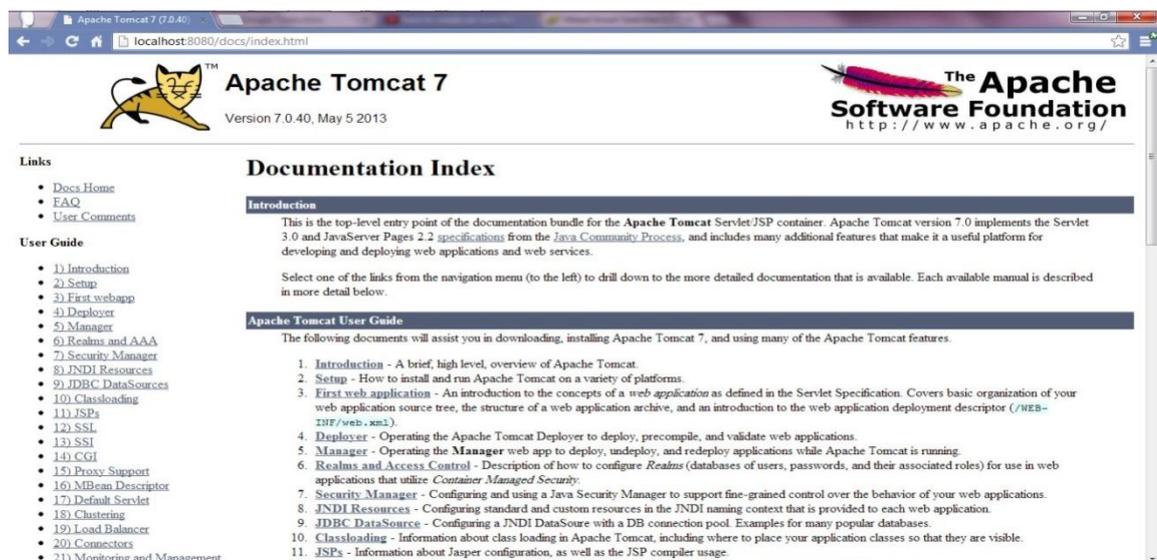


Figure V-4 : l'interface Apache Tomcat Server 7.0.40

2.5. Choix des outils et des technologies supplémentaires :

2.5.1. SPARQL :

SPARQL est l'équivalent de SQL car comme en SQL, on accède aux données d'une base de données via ce langage de requête alors qu'avec SPARQL, on accède aux données du Web des données. Cela signifie qu'en théorie, on pourrait accéder à toutes les données du Web avec ce standard. L'ambition du W3C est d'offrir une interopérabilité non pas seulement aux niveaux des services, comme avec les services Web, mais aussi aux niveaux des données structurées ou non qui sont disponible à travers l'Internet Ce standard a été créé par le groupe de travail DAWG du W3C, SPARQL est considéré comme l'une des technologies clés du Web sémantique et le 15 Janvier 2008, la version 1.0 est devenue une recommandation officielle du W3C. La version 1.1 permettra d'enregistrer des données et de fusionner des données de sources différentes. Le dernier appel à contribution a eu lieu donc la version 1.1 sera bientôt candidat à la recommandation pour devenir un nouveau standard du Web.

2.5.2. SKOS :

SKOS (Simple Knowledge Organization System) Système simple d'organisation des connaissances: est une famille de langages formels permettant une représentation standard des thésaurus, classifications ou tout autre type de vocabulaire contrôlé et structuré. Construit sur la base du modèle de données standard RDF, son principal objectif est de permettre la publication facile de vocabulaires structurés pour leur utilisation dans le cadre du Web sémantique. SKOS est, depuis le 18 août 2009, une recommandation du W3C.

Le développement de SKOS a impliqué des acteurs à la fois de la communauté RDF et des experts en Science de l'information. SKOS cherche à être au maximum compatible avec les standards tels ceux des thésaurus, monolingue ou multilingue¹.

Les représentations conceptuelles réalisées à l'aide de SKOS peuvent satisfaire des besoins de traitement restreints à un organisme, mais aussi, dans la perspective du Web sémantique, contribuer à la constitution d'une structure de concepts mis en commun et partagés à l'échelle du Web sous forme de ressource exploitable par les outils RDF ou autres.

2.5.3. Apache Lucene™ 4.3 :

Apache Lucene™ est une bibliothèque de moteur de recherche de haute performance complet entièrement écrit en Java. Il s'agit d'une technologie appropriée pour presque n'importe quelle application qui nécessite la recherche plein texte, en particulier multi-plate-forme. Apache Lucene est un projet open source.

2.5.4. TIKATM 1.3 :

Apache Tika™ détecte et extrait les métadonnées et le contenu de texte structuré à partir de divers documents à l'aide de l'analyseur bibliothèques existantes.

Tika est un projet de l'Apache Software Foundation, et était autrefois un sous-projet d'Apache Lucene.

2.5.5. Crawler4J:

Crawler4J est une bibliothèque Open Source ont été conçus en utilisant le langage JAVA contient des fonctions qui assurent l'exploration automatiquement de Web. Il est généralement conçu pour collecter les ressources (pages Web, images, vidéos, document, PDF ou PostScript,... etc.), afin de permettre à un moteur de recherche de les indexer.

3. ARCHITECTURE GENERALE DU SYSTEME :

Notre travail est basé sur trois approches principaux qui sont:

- ✓ Le Crawling.
- ✓ La recherche.
- ✓ Les résultats.

3.1. Le *Crawling* :

C'est la première étape de système où l'administrateur peut lancer le *crawling* avec l'initialisation et la configuration de *Crawl* pour commencer ses tâches, le *Crawl* crée des Spiders selon les caractéristiques des liens à *crawler*, le spider visite ces liens, les indexés, puis il crée un fichier résultat contient le liens d'index de site et tous les autres liens liée avec cet index, aussi il calcule la redondances des mots, et mettez les plus répétés comme des mots clés pour ce site, le spider fait une statistique pour chaque site comme le nombre total des mots, documents et tous autres fichier dans ce lien.

3.2. La recherche :

Dans cette étapes l'utilisateur écrit un mot ou phrase pour chercher et lancer la recherche, notre système crée un agent pour faire la tâche de lire la requête entrer par l'utilisateur, envoyer trois copies de cette requete: une pour la recherche de sémantique de ce mot – On fait cette opération par l'envoi d'une requête qui contient le mot à chercher et interroger l'ontologie - , et la deuxième pour obtenir les correction automatique – s'il y a une erreur dans le mot à chercher – , et la troisième pour obtenir des suggestion similaire correspond à cette recherche – Ici on utilisant l'outil SKOS -. Finalement, on obtient les mots relié à cette requête, ici on interroge la base d'index créé par le *Crawler* .

3.3. Les résultats :

C'est la dernière étapes du système, l'application affiche les résultats correspond à la requête d'utilisateur, avec la présence de la correction s'il y a une erreur de saisi, et la liste des *suggestion* .

Et voilà, comment notre application se fonctionne.

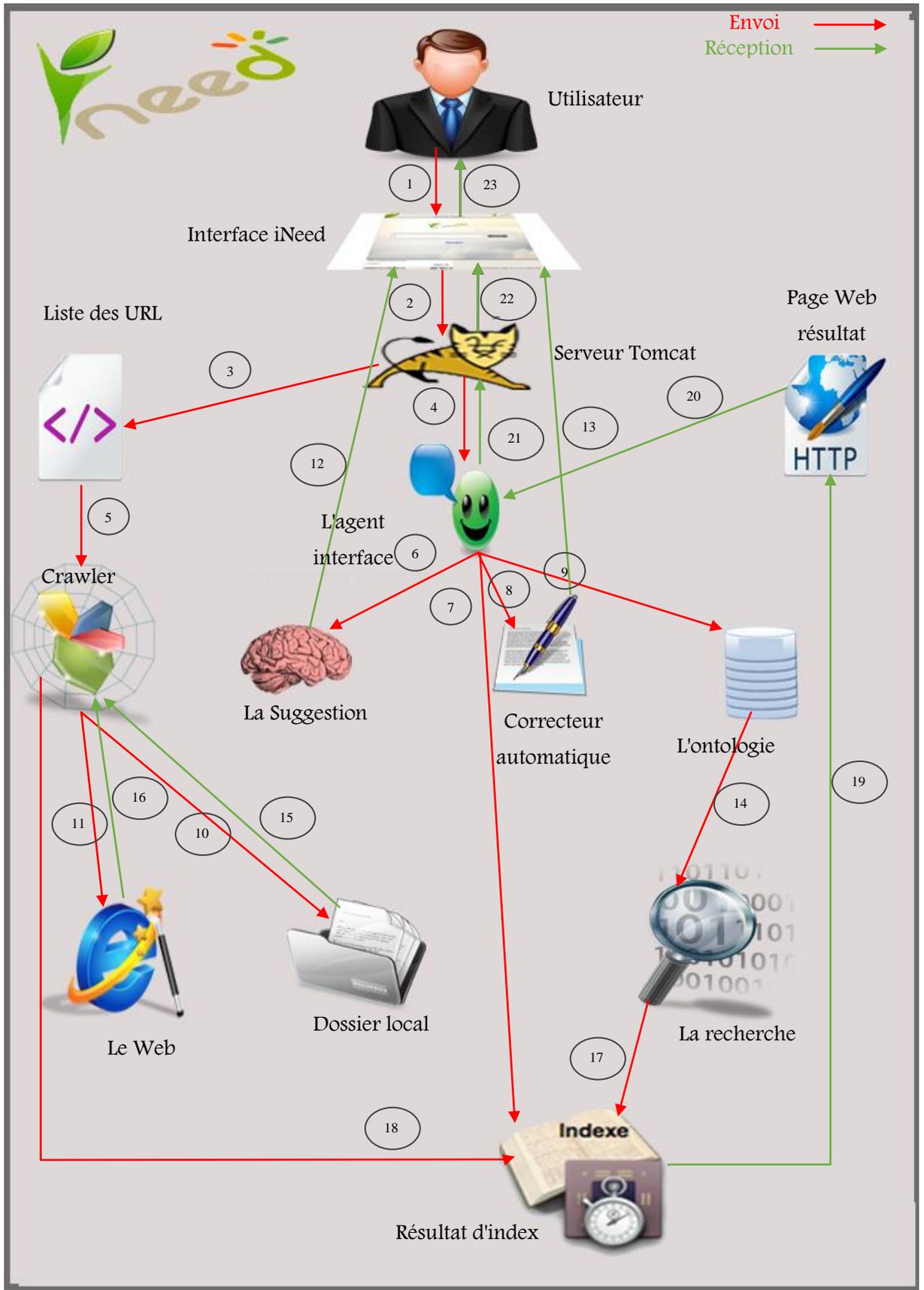


Figure V-5 : L'architecture générale de notre application "iNeed".

3.4. Explication de l'architecture :

- (1) 'utilisateur entre une requête.
- (2) Le serveur Tomcat obtient la requête.
- (3) Tomcat envoi liste d'url d'index (l'initialisation).
- (4) Tomcat envoi la requête au agent.
- (5) Le *crawler* recevoir l'ordre de démarrage.
- (6) L'agent envoi la requête au rédacteur de proposition.
- (7) L'agent envoi la requête pour vérifier les erreur de saisi s'ils trouves.
- (8) L'agent envoi la requête à l'ontologie pour obtient la sémantique correspond.
- (9) L'agent envoi la requête directement à base d'indexation.
- (10) Le *Crawler* crée des *Spider* et les envoyées au Web.
- (11) Le *Crawler* crée des *Spider* et les envoyées au dossier local de serveur.
- (12) Les suggestions sont prêtes.
- (13) Les corrections sont afficher directement s'il y a une erreur de saisi.
- (14) L'ontologie interroge la base d'indexation avec la sémantique de cette requête.
- (15) Les spiders renvoi les fichier indexer à partir de dossier serveur.
- (16) Les spiders renvoi les fichier indexer à partir le Web.
- (17) L'ontologie interroge la base d'index.
- (18) Le *crawler* enrichie la base d'index par les nouveauté de *Crawling*.
- (19) La base d'index répond par des lien du page web correspondant à la requête.
- (20) L'agent obtient la liste des liens résultat.
- (21) L'agent envoi la réponse à partir de serveur.
- (22) Le serveur envoi les résultats.
- (23) L'interface utilisateur affiche les liens résultats.

4. IMPLEMENTATION DES COMPOSANTS DU SYSTEME :

Notre application consiste à développer un système de recherche optimale pour le Web, Cela revient à construire un moteur qui procède à une recherche pour offrir une meilleure exploitation des ressources disponible sur le Web lors d'une recherche lancée par un utilisateur donné. Pour cela, on utilise les technologies et les langages décrits précédemment dans ce travail tel que le langage JAVA, le plate-forme Jade et les ontologies.

Notre outil est appelé **iNeed** synonyme anglais de "j'ai besoin" et pour simplifier la démarche de notre application, on peut la diviser en deux grandes cotés sont :

4.1. Coté administrateur :

L'administrateur ou bien le concepteur peut faire les taches suivantes :

- ✓ lancer l'exploration (Crawling) :

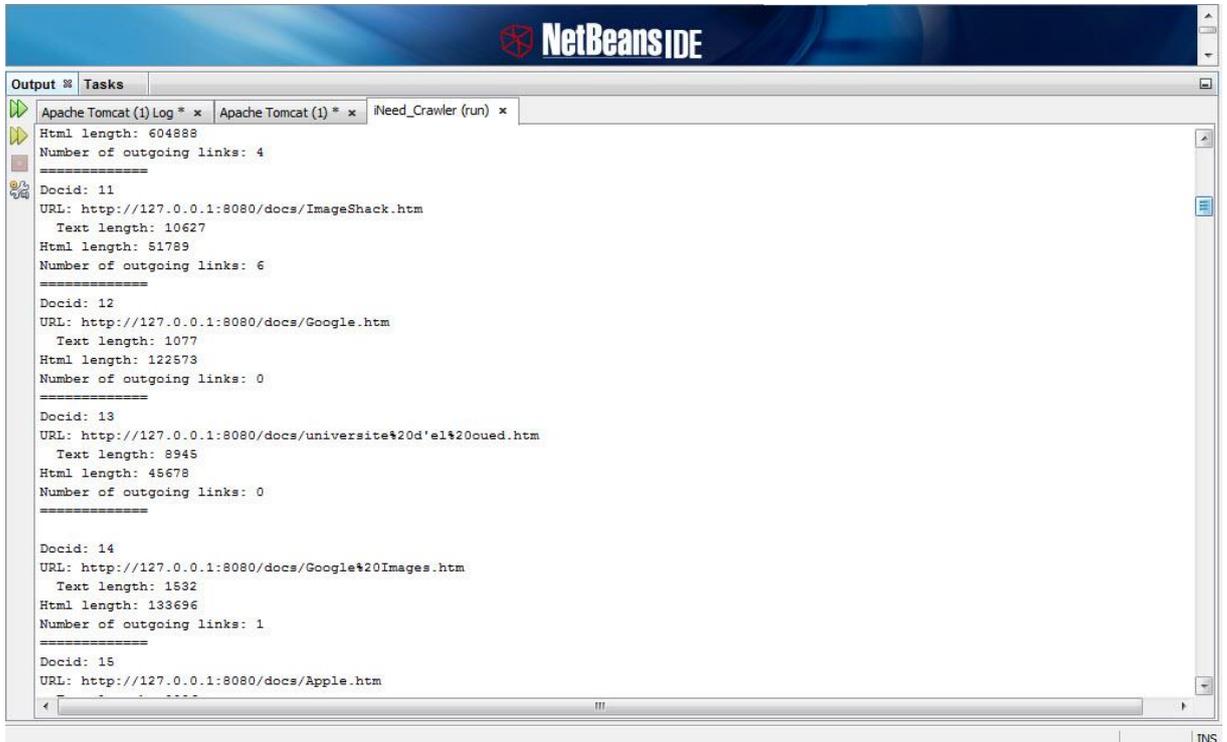


Figure V-6: La tache de crawling en cours d'exécution .

- ✓ Puis il obtient les résultats de Crawl dans un dossier local comme ce suite :

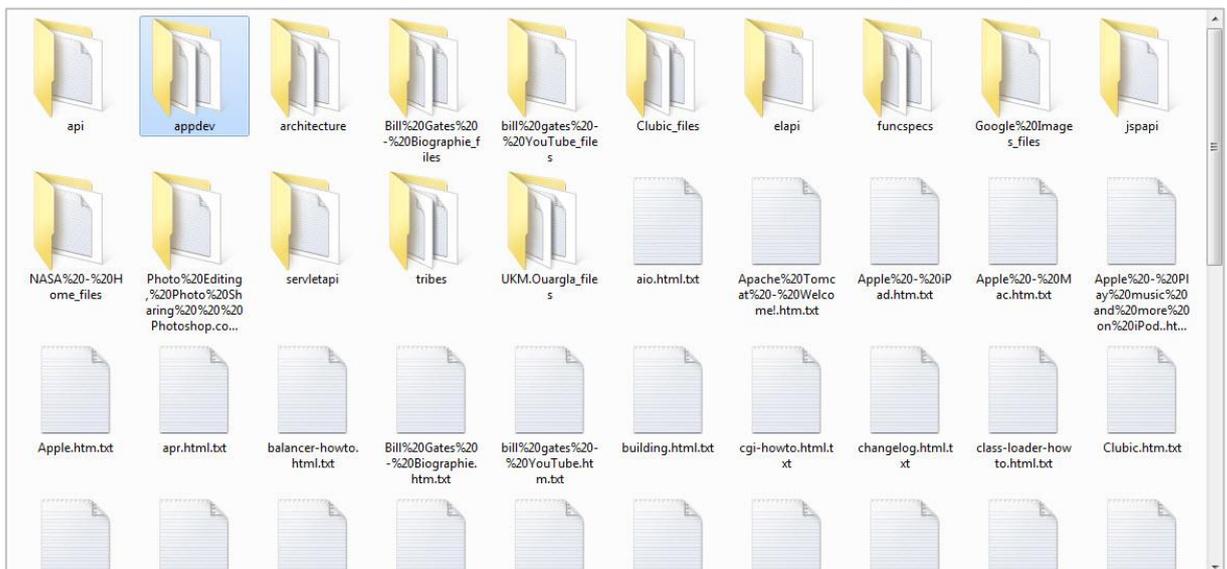


Figure V-7 : Les résultats de Crawling .

4.2.Coté utilisateur :

L'interface principale de notre application est simple, elle est constituée d'une zone de texte où l'utilisateur saisit les mots clés, et un bouton pour valider la saisie et lancer la recherche comme le montre la figure suivante :

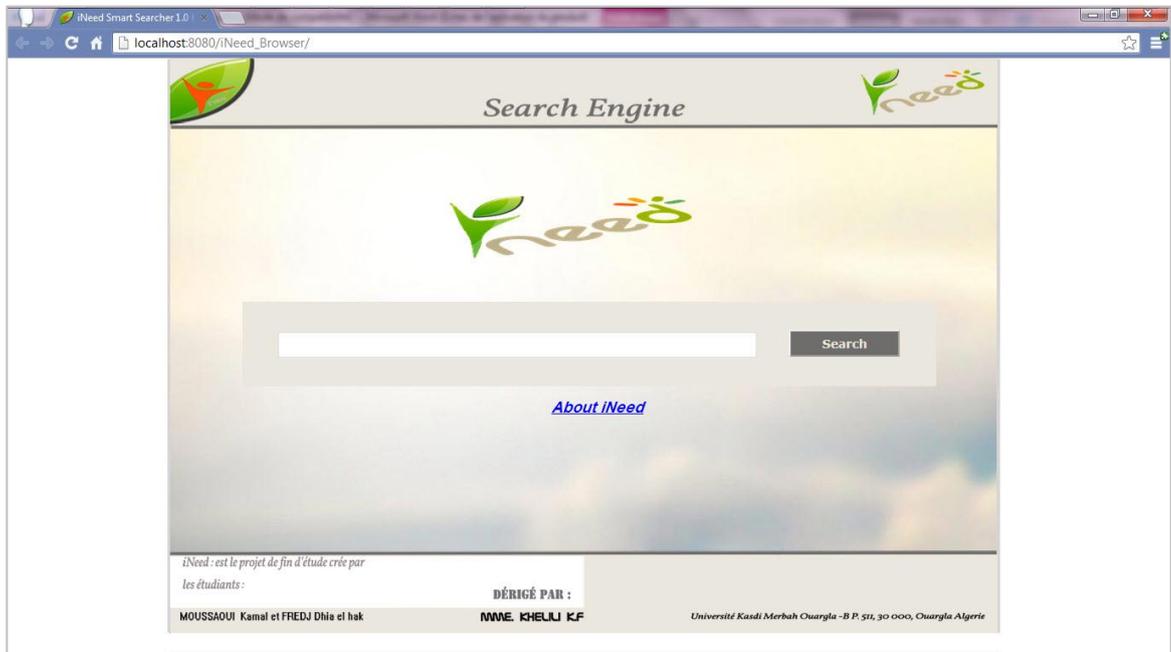


Figure V-10 :L'interface utilisateur principale d'iNeed .

Quand l'utilisateur lance une recherche, il obtient une autre page de résultat comme le montre la figure suivante:

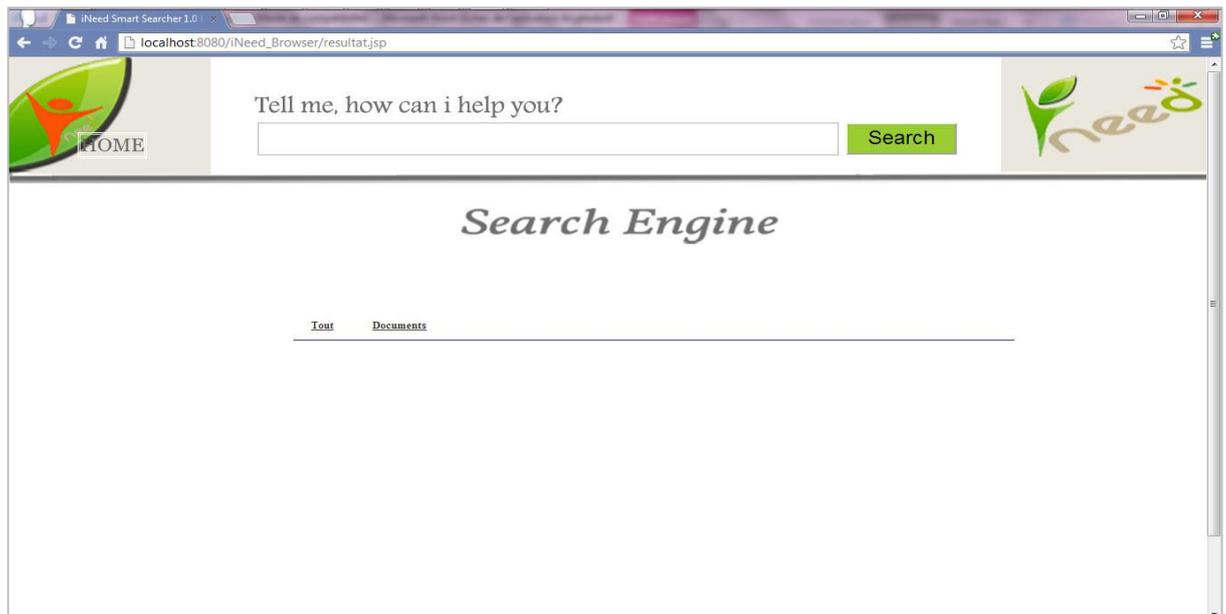


Figure V-11 :L'interface utilisateur du résultat d'iNeed.

Aussi on ajoute un lien de (A propos iNeed) pour expliquer et donner une vision générale à propos ce travail et pour aussi donner un peu de professionnalité.

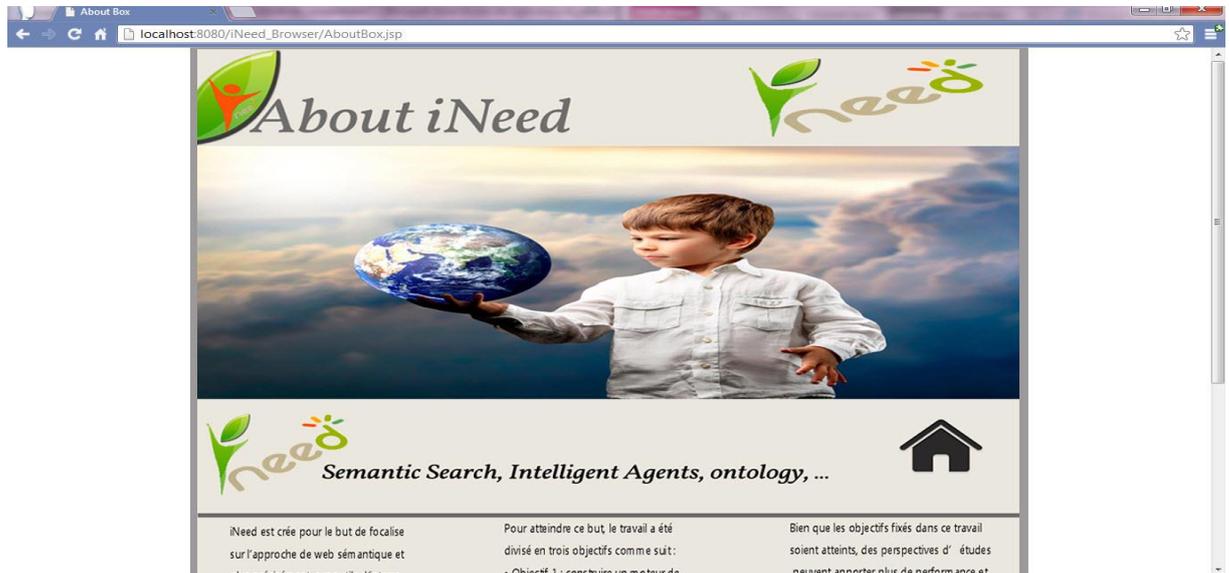


Figure V-12 :L'interface A propos d'iNeed .

5. L'INTELLIGENCE DE NOTRE OUTIL :

iNeed est un outil très simple, il consiste à lire la requête introduite par les utilisateurs, faire les recherches et puis afficher les résultats.

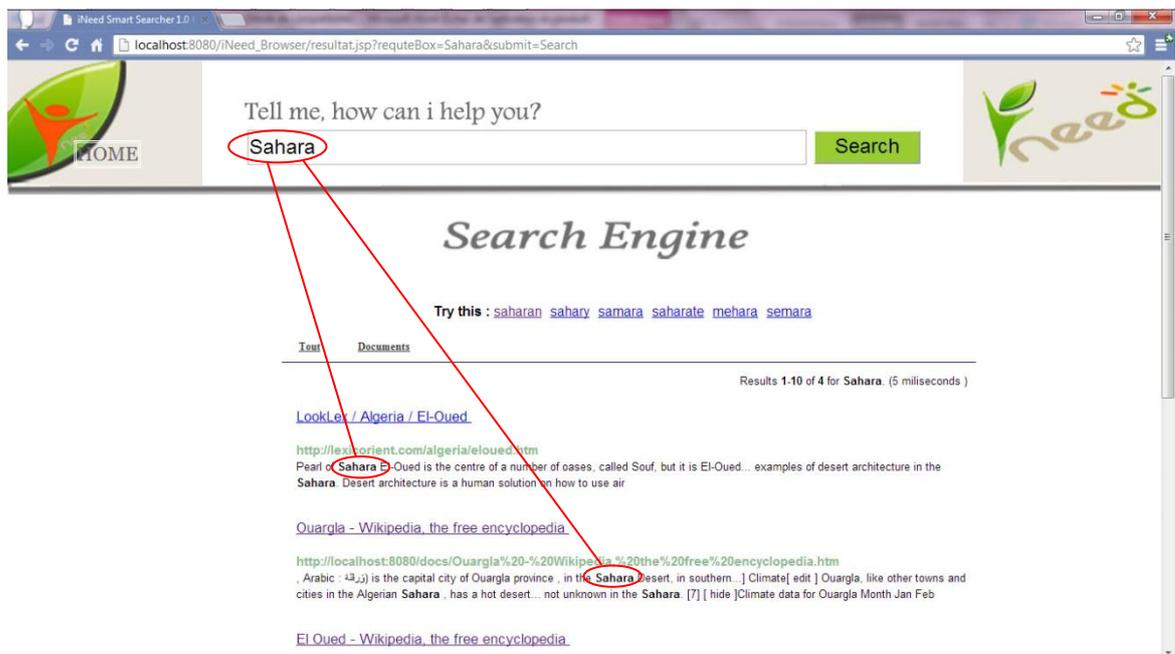


Figure V-13 :résultat d'une recherche .

Il contient plusieurs caractéristiques intelligentes telles que :

✓ La recherche sémantique :

avec l'utilisation de l'ontologie, dans cet exemple le mot recherché c'est : "site", les résultats de recherche affichent en gras les mots de même sens que le mot site tels que: Web, Pages Web, WebSite,

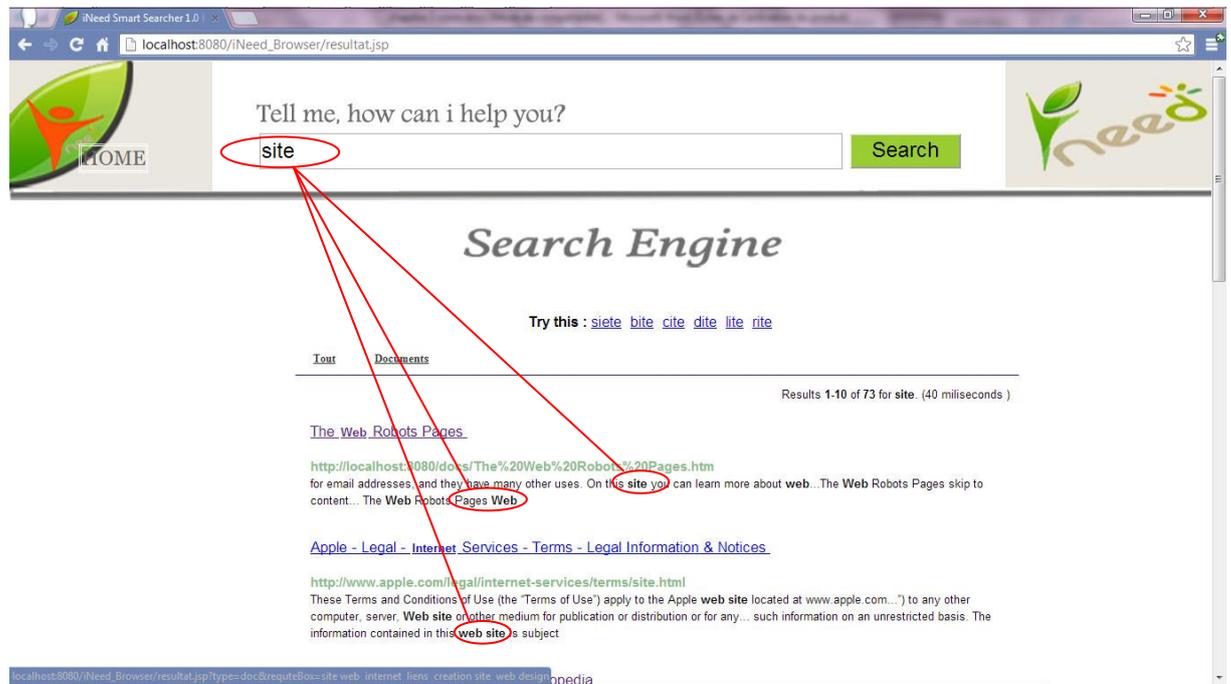


Figure V-14 : la recherche sémantique .

✓ Rédacteur des propositions (la suggestion):

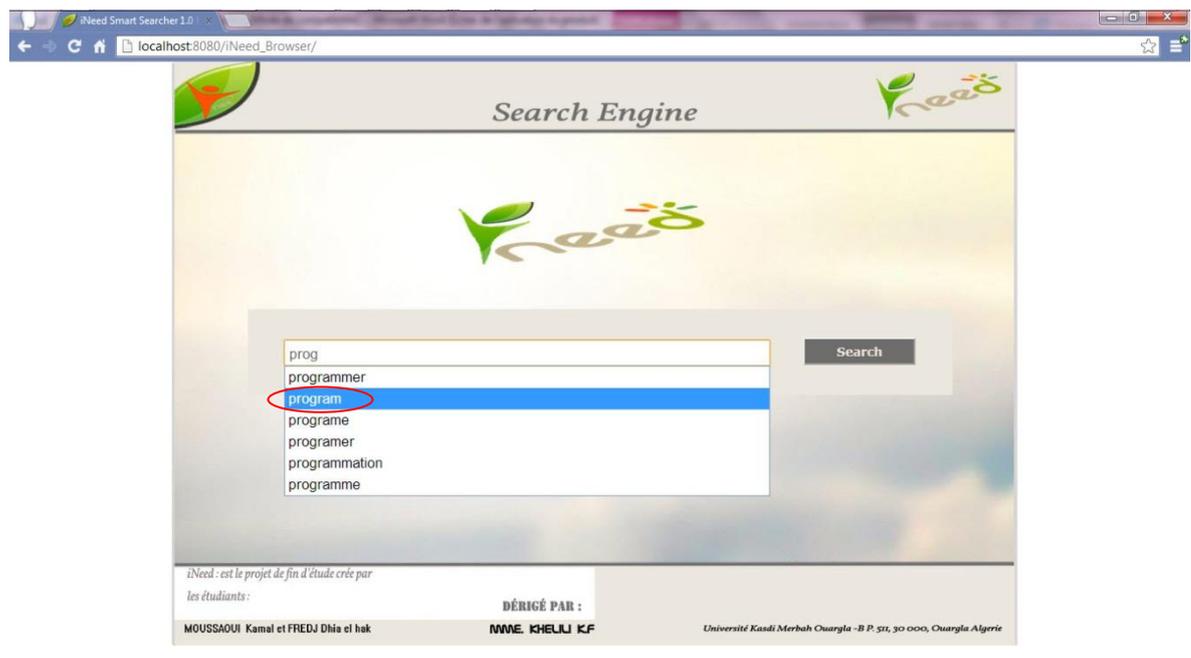


Figure V-15 : le rédacteur des propositions.

✓ Correcteur automatique :

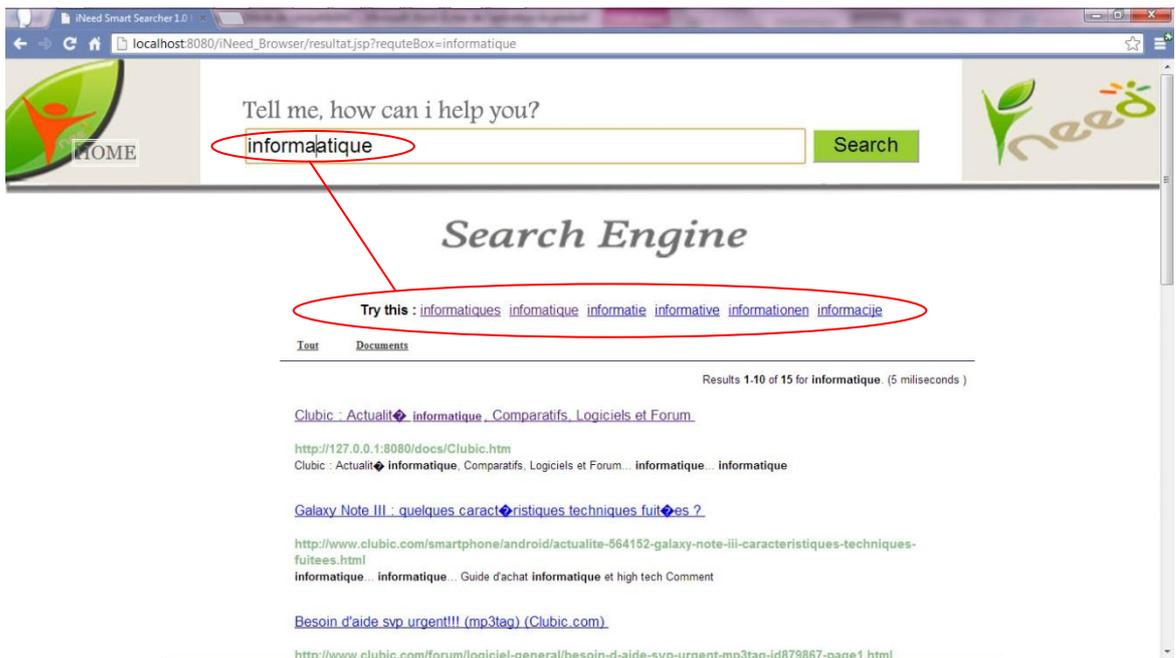


Figure V-16 : la correcteur automatique de iNeed .

✓ La recherche par types :

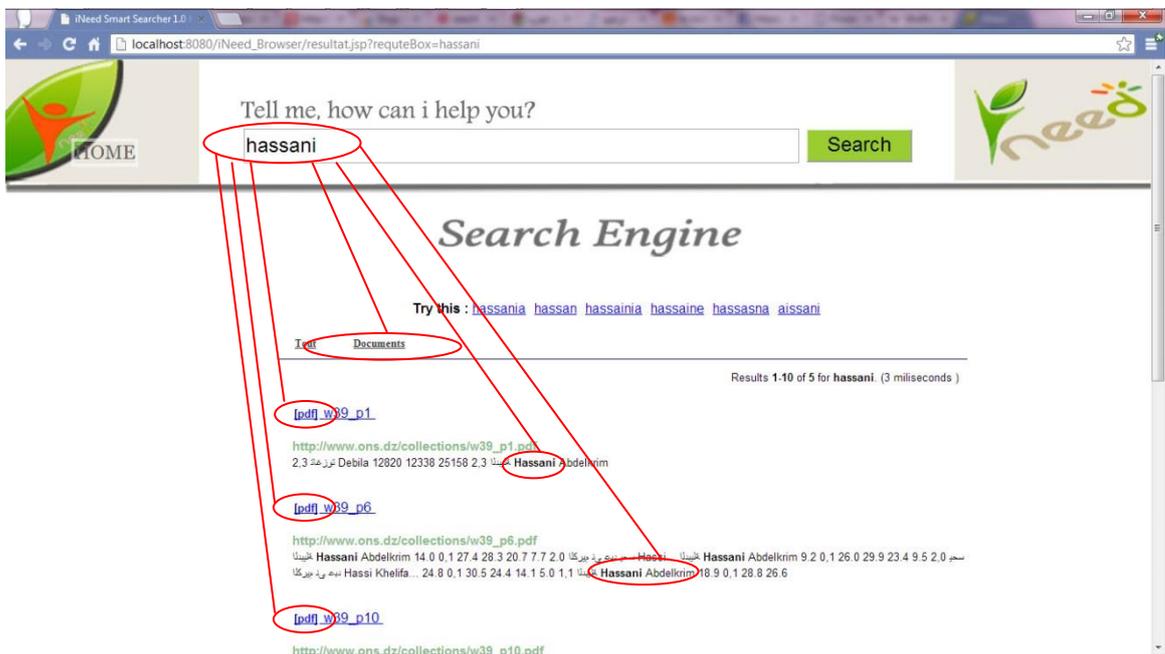


Figure V-17 : la recherche par type.

✓ Permission des téléchargements :

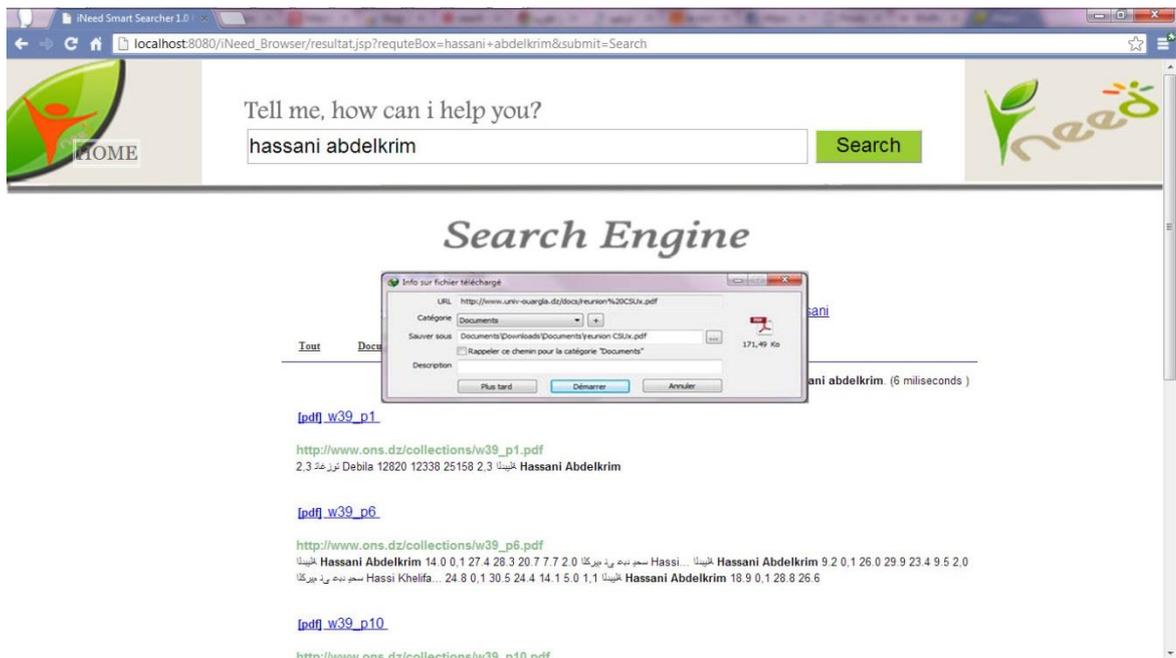


Figure V-18 :lancement d'un téléchargement.

6. CONCLUSION :

Nous avons vu - tout au long de ce chapitre – les différents outils et technologies nécessaires pour la réalisation de notre système tels que l'environnement de développement, les langages de programmations, les bibliothèques open source utilisées... etc.

Nous avons montré l'implémentation de chaque composant de notre système proposé, afin d'assurer une présentation claire et détaillée de notre outil. De plus, on a cité et expliqué les caractéristiques de notre application. Finalement, nous avons donné quelques résultats obtenus on utilisant des captures d'écran.

Conclusion Générale

Conclusion générale

Au terme de ce mémoire nous tentons de tirer des conclusions des travaux présentés et de discuter les perspectives d'amélioration de nos propositions. L'objectif principal de ce travail était d'apporter des contributions sur deux axes complémentaires: d'abord l'amélioration du processus de recherche sur le web, puis création et amélioration d'un outil de recherche qui répond aux besoins des internautes. En effet, le grand nombre de documents disponibles sur le web a soulevé l'attractivité à l'amélioration des outils de recherche d'information.

Les outils de recherche actuels (tels que Google, Yahoo et Bing) sont les plus utilisés pour parcourir le contenu du Web (BOURAMOUL, 2011). Toutefois, Ce type d'outil ne permet pas d'atteindre une grande efficacité et les résultats qu'il retourne ne correspondent pas toujours aux besoins des utilisateurs.

Pour cette raison, notre proposition utilise deux mécanismes principaux : d'une part, les agents relient aux différents acteurs autour du processus de recherche, et d'une autre part la sémantique portée par les termes de la requête et les mots des documents avec l'utilisation de l'ontologie. Le but est d'augmenter la sélectivité de notre outil de recherche d'information (*iNeed*).

Le travail présenté dans ce mémoire essaye de se bénéficier au maximum des notions de web sémantique, et plus précisément, sur la réalisation d'un outil pour l'interrogation du web. En effet, notre travail consiste à utiliser les agents avec l'intégration de la notion d'ontologie.

Pour atteindre ce but, la réalisation de l'application a été divisée en trois objectifs :

- Objectif 1 : construire un moteur de recherche.
- Objectif 2 : création d'agents.
- Objectif 3: création et intégration des ontologies.

les perspectives d'études peuvent apporter plus de performance et d'efficacité à l'outil de recherche d'information sur le web, ces perspectives sont :

- Améliorer des algorithmes plus rapides et efficaces pour l'indexation.
- améliorer le profil utilisateur.

- L'addition de la possibilité de communication entre les utilisateurs de notre système par internet.
- Internationaliser notre moteur par l'addition d'autre langue a l'interface telle que l'arabe, français.
- Ajouter la recherche dans les fichiers multimédias (Sons, Images, Vidéos)
- Traitement des langages naturels qui permet à l'utilisateur d'introduire n'importe qu'elle requête et obtenir des meilleures résultats.

Bibliographie

- Agentland. (2009). *le portail des agents intelligents : agents de veille, agents de recherche d'information sur Internet.*
- Ahmed, B. (2011). *balicours*. Récupéré sur <http://sites.google.com/site/balicours>.
- Ahmed, B. (2011). *Introduction à UML*. El-oued.
- al, T. L. (2002, May). The semantic Web. in Scientific American.
- B. Chaib-draa. (2010). *Systèmes Multiagents Cours IFT-7011*.
- Bachimont, B. (2004). Arts et sciences du numérique : Ingénierie des connaissances et critique de la raison computationnelle. Compiègne, Mémoire d'Habilitation à Diriger des Recherches, Université de Technologie de Compiègne.
- Bal Kamal. (2010). *Recherche d'information dans les documents XML: Approche par agrégation partielle des sources de pertinence.*
- Boissier, O. (2004). *Systèmes Multi-Agents*.
- Borst, W. N. (1997). Construction of engineering ontologies. *University of Twente Centre for Telematica and Information Technology*. Enschede.
- BOURAMOUL, A. (2011). *RECHERCHE D'INFORMATION* .
- Chaib-draa, B. (1999). *Agents et Systèmes Multi-Agents*.
- D. L. McGuinness, F. v. (2004, 02 10). Owl web ontology language overview. *World Wide Web Consortium, Recommendation REC-owlfeatures*.
- D. Oberle, R. V. (2004). An extensible ontology software environment. *Staab et R. Studer (Eds.), Handbook on Ontologies*. Springer Verlag.
- D.Revuz. (2004). UML Elaboration .
- Darmont J. (2000). *Programmation sous Delphi*.
- DUTIL, I. (2009). *Les agents Intelligents Sur Internet*.
- ESPINASSE, B. (2010). Introduction aux Ontologies. Marseille , Professeur à l'Université d'Aix-Marseille.
- FORESTIER, A. (2003). *Qu'est ce qu'un algorithme de moteur de recherche?*
- Fox, M. G. (1995). Methodology for the Design and Evaluation of Ontologies. *In:Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing*. Montreal, IJCAI-95.
- François Bourdon et Patrice Enjalbert, A. I. (2000). *Systèmes Multi-Agents*.

- Furst, F. (02-07-2002). *L'ingénierie ontologique*. Rapport de recherche.
- G. Van Heijst, A. S. (1997). Using explicit ontologies in kbs development. *International Journal of Human and Computer Studies*. Knowledge, Acquisition: 183–292 pages.
- Gomez Pérez A., B. V. (1999.). Overview of Knowledge Sharing and Reuse Components Ontologies and problem-Solving Methods. Dans *Proceeding of the IJCAI-99 workshop on Ontologies and problem-Solving Methods (KRR5)* (pp. 1.1-1.15). Stockholm (Suède).
- Gruber, T. (1993). A translation approach to portable ontology specification.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. Dans *Knowledge Acquisition* (pp. 5(2) :199–220).
- Gruber, T.-R. (1995). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal Human-Computer Studies*. 3(5–6): 907–928.
- Grüninger, M. U. (1996). ONTOLOGIES: Principles, Methods and Applications. *Knowledge Engineering Review*.
- Guarino, N. (1997.). Understanding, building, and using ontologies. Dans I. J. Studies.
- Guarino, N. (1998). Formal ontology in information systems. Dans *In the 1st International Conference on Formal Ontology in Information Systems (FOIS)* (pp. 3–15). Trento Italy:IOS Press.
- Guarino, N. (s.d.). Some organizing principles for a unified top-level ontology. Dans *In National Conference of the American Association on Artificial Intelligence (AAAI)* (pp. 57–63). United-States: Stanford.
- I. Horrocks, P. P.-S. (2004, May 21). SWRL: A Semantic Web Rule Language Combining OWL and RuleML. *W3C*.
- J. Ferber. (1995). *les Systèmes Multi-Agents, Vers une intelligence collective*.
- Jean-Pierre et Yves, B. e. (2001). *Introduction agents*.
- John Richard, W. (2005). *Les agents intelligents sur Internet : enjeux économiques et sociétaux*.
- Karima AMROUCHE. (2008). *Passage à l'échelle en Recherche d'Information : Méthode d'élagage pour la réduction de l'espace de recherche*.
- L. Lassila, R. R. (1999, February 22). Resource description framework (rdf) model and syntax specification. *specification. World Wide Web Consortium W3C Recommendation* .
- Lambert & Médaille & Welsh. (2006). *Internet et la recherche d'information Des outils de niveaux universitaire et spécialisé à découvrir*.
- LARGOUET, A. (2005). *La recherche d'informations sur Internet. Rapport de recherche*.
- LAWRENCE & GILES, S. &. (1998). *How big in the Web*.
- M. Ikeda, K. e. (1997). Task Ontology Makes It Easier To Use Authoring Tools. Dans *Proc. of IJCAI-97* (pp. 342-347).
- M.Hemam. (2005). *Un processus de developpement d'onntologie dans le cadre du web sémantique. - Oum El Bouaghi-: Centre Universitaire Larbi Ben M'hidi -Oum El Bouaghi-Institut des sciences exactes*.

- McGuinness, N. F. (2001, March). *Ontology Development 101: A Guide to CreatingStanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880.*
- N. Guarino, A. G. (1995). Understanding top-level ontological distinctions. The International Joint Conference on Artificial Intelligence (IJCAI), In Workshop on Basic Ontological Issues in Knowledge Sharing.
- Nadia BEN SEGHIR. (2009). *Une approche système multi-agent basée agent mobile dans un système réparti.*
- Napoli, A. (1997). Une introduction aux logiques de description. INRIA, Technical report.
- NetCraft. (2013). <http://news.netcraft.com/archives/category/web-server-survey/>.
- PICAROUGNE, F. (2004). *Recherche d'information sur Internet par algorithmes évolutionnaires.*
- Porter, M. (2006). *The Porter Stemming Algorithm.*
- R. Mizoguchi, M. I. (1997). Towards ontology engineering. Dans *International Conference on Intelligent Systems* (pp. 259–266). Singapore: Pacific Asian Conference on Expert systems.
- R. Neches, R. F. (1991). Enabling technology for knowledge sharing. *AI Magazine.*
- R.ZAIANE, O. (1999). *From resource discovery to knowledge discovery on the Inernet.*
- Roussey C. (2001). *Une méthode d'indexation sémantique adaptée aux corpus multilingues.*
- S. Staab, A. M. (2000). Axioms are objects too: Ontology engineering beyond the modeling of concepts and relations, Research report 399. Karlsruhe, Institute AIFB.
- T. Berners-Lee, J. H. (2001). The Semantic Web. *Scientific American Magazine*, 34-43.
- WorldWideWebSize. (2013). <http://www.worldwidewebsite.com/>.
- Yves, D. e.-P. (2001). *Introduction agents.*
- Riad LEKHCHINE,2008/2009.Mémoire Thème : Construction d'une ontologie pour le domaine de la sécurité : Application aux agents mobiles.
- ASKANE Younes,EL-OUCHI Younes,EL MESSBAHI Salima,EL JADIDI ALAOUI Sadik.rapport l'onologie. Université Abdelmalek Essadi 2009/2010.
- Melle Samia BOUARROUDJ,2009-2010, Mémoire: Magister.ecole Doctorale en Informatique de l'Est Pole ANNABA. Raisonement sur une ontologieenrichir par des regles SWRL pour la recherche sémantique d'images annotées, Université 20 Août 1955, Skikda.
- Baazaoui Zghal, H., Aufaure, MA. and Ben Mustapha, N. (2007). Extraction of Ontologies from Web Pages: conceptual modeling and tourism, Journal of internet Technologies Volume 8 No. 4 (2007), octobre 2007, ISSN 1607-9264.

- Baziz, M., (2004). Towards a Semantic Representation of Documents by Ontology- Document Mapping. The Eleventh International Conference on Artificial Intelligence (AIMSA 2004).
- Miller, G.A. (1995). WordNet: A Lexical Database for English. Communications of the ACM, 11, 39-41.
- Voorhees, E. M., Gupta, N. K. et Johnson-Laird, B. (1994). The collection fusion problem. Proceedings of the 3rd Text REtrieval Conference (TREC-3). p. 95-104.

Webographie

(site1)<http://news.netcraft.com/archives/category/web-server-survey/>

(site2)<http://www.worldwidewebsite.com/>

(site3)<http://www.themediatrend.com/wordpress/2008/11/20/les-moteurs-de-recherche-semantique-un-pas-dans-le-web-30/>

(site4)<http://www.licence-referencement.fr/2012/03/28/recherche-semantique-referencement-google/>

Résumé

Aujourd'hui, le web est la plus large ressource le plus utilisé pour la recherche, et aussi la publication des informations, mais d'autre part, l'augmentation des données accessibles via le réseau pose des problèmes tels que la saturation du réseau et donc la recherche dans le web est devenu très lente, et que des millions de surfeurs internautes ne parviennent pas à satisfaire la totalité de leurs besoins en information.

Donc, il est indépasseable de trouver des nouvelles solutions pour améliorer la recherche d'information dans le Web, les moteurs de recherche intelligents apparaissent dans ce contexte comme une solution prometteuse permettant la limitation de l'impact de ces problèmes.

Dans ce travail, on proposera une architecture à base d'agents intelligents avec l'utilisation de l'ontologie qui servira de base pour développer notre moteur de recherche.

Mots clés : Outil de recherche d'information, Recherche d'information ,Web sémantique, Agent, ontologie.

ملخص

اليوم، شبكة الإنترنت هي المورد الأكبر و الأكثر استخداما للعثور على المعلومات، وكذلك نشر المعلومات، ولكن من ناحية أخرى، فإن ازدياد البيانات ينطوي على مشاكل مثل ازدحام الشبكة، وكذلك البحث في الشبكة يصبح بطيئا جداً. وأن الملايين من متصفحى الإنترنت تفشل في تلبية جميع احتياجاتهم من المعلومات. ولذلك، فمن الضروري إيجاد حلول جديدة لمعالجة المعلومات. محركات البحث تعتبر في هذا السياق حلا و اعدا للحد من آثار هذه المشاكل. في هذا العمل، نقترح هندسة تعتمد على الوكلاء مع استخدام الأنطولوجيا كأساس لتطوير محرك بحثنا.

كلمات مفتاحية : أدوات البحث عن المعلومات، البحث عن المعلومات، ويب دلالي، وكيل، أنطولوجيا.

Abstract

Today, the web is the largest and most used resource for finding information, and also the publication of information, but on the other hand, the increase in accessible data networks involves problems such as congestion network and also research in the web become very slowly, and millions of the net users surfers fail to satisfy their full informational needs.

Therefore, new solutions become available to manipulate the information from the resource. the intelligent search engines appear in this context as a promising solution to limiting the impact of these problems.

In this work, we propose an architecture based on intelligent agents with the use of ontology as a basis for developing our search engine

Keywords : Search information tools, Information Retrieval, Semantic Web, Agent, ontology.